スイッチベースの認証の設定

この章では、Catalyst 3750 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。特に明記しないかぎり、*スイッチ*という用語はスタンドアロン スイッチおよびスイッチ スタックを意味します。

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.9-2）
- イネーブル EXEC コマンドへのアクセスの保護（p.9-3）
- TACACS+ によるスイッチ アクセスの制御（p.9-11）
- RADIUS によるスイッチ アクセスの制御（p.9-19）
- Kerberos によるスイッチ アクセスの制御（p.9-33）
- スイッチのローカル認証および許可の設定（p.9-38）
- SSH のためのスイッチの設定（p.9-39）
- スイッチの SSL HTTP の設定（p.9-44）
- SCP のためのスイッチ設定（p.9-51）

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は、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.9-3）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.9-8）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.9-11）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、次の設定について説明します。

- デフォルトのパスワードおよび権限レベル設定（p.9-3）
- 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.9-3）
- 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.9-4）
- パスワード回復のディセーブル化（p.9-6）
- 端末回線に対する Telnet パスワードの設定（p.9-7）
- ユーザ名とパスワードのペアの設定（p.9-8）
- 複数の権限レベルの設定（p.9-8）

デフォルトのパスワードおよび権限レベル設定

表 9-1 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 9-1　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	**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	**enable password** *password*	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。 *password* 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 Ctrl-v を入力すれば使用できます。たとえば、パスワード abc?123 を作成するときは、次のようにします。 **abc** を入力します。 **Ctrl-v** を入力します。 **?123** 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 Ctrl-v 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 abc?123 と入力できます。
ステップ 3	**end**	イネーブル EXEC モードに戻ります。
ステップ 4	**show running-config**	設定を確認します。
ステップ 5	**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。 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 *l1u2c3k4y5* 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 l1u2c3k4y5
```

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、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特に設定する場合には、**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ので、**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	説明
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	**enable password** [**level** *level*] {*password* | *encryption-type encrypted-password*} または **enable secret** [**level** *level*] {*password* | *encryption-type encrypted-password*}	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。 • （任意）*level* 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）。 • *password* 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。 • （任意）*encryption-type* 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。 **（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できません。
ステップ 3	**service password-encryption**	（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	**end**	イネーブル EXEC モードに戻ります。
ステップ 5	**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。

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.9-8）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。

権限レベル 2 に対して暗号化パスワード *1FaD0$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、スイッチに物理的に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と、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の一部をディセーブルにすると、スイッチ パスワードへのアクセスを防止できます。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 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 コピーを保存しておいてください。スイッチ上には、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 VTP 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.43-4）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	**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	**no service password-recovery**	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 ローダおよび Cisco IOS イメージがアクセスできる領域に保存されます。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	**end**	イネーブル EXEC モードに戻ります。
ステップ 4	**show version**	コマンド出力の最後の数行を見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、スイッチの電源再投入後に、ブート ローダ 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、継続して使用できるようにデフォルト設定を作成します。またセットアップ プログラムは、スイッチでパスワードを介した Telnet へのアクセス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	**enable password** *password*	イネーブル EXEC モードを開始します。
ステップ 3	**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 4	**line vty 0 15**	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	**password** *password*	1 つまたは複数の回線の Telnet パスワードを入力します。 *password* 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	**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。 コマンド **line vty 0 15** の下にパスワードが表示されます。
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。

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し、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	**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	**username** *name* [**privilege** *level*] {**password** *encryption-type password*}	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。 • *name* 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。 • （任意）*level* 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。 • *encryption-type* 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。 • *password* 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、**username** コマンドの最後のオプションとして指定します。
ステップ 3	**line console 0** または **line vty 0 15**	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	**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。

複数の権限レベルの設定

Cisco IOS 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。

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

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	**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*	コマンドの権限レベルを設定します。 • *mode* 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 **configure** を、EXEC モードの場合は **exec** 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 **interface** を、ライン コンフィギュレーション モードの場合は **line** を、それぞれ入力します。 • *level* 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、**enable** パスワードによって許可されるアクセス レベルです。 • *command* には、アクセスを制限したいコマンドを指定します。
ステップ 3	**enable password level** *level password*	権限レベルのイネーブル パスワードを指定します。 • *level* 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。 • *password* 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 4	**end**	イネーブル EXEC モードに戻ります。
ステップ 5	**show running-config** または **show privilege**	設定を確認します。 **show running-config** 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。**show previlege** コマンドは、権限レベルの設定を表示します。
ステップ 6	**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。

コマンドをある権限レベルに設定すると、そのコマンドと同じ構文をサブセットとして持つコマンドもすべて同じ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	**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	**line vty** *line*	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	**privilege level** *level*	回線のデフォルト権限レベルを変更します。 *level* 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、**enable** 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	**end**	イネーブル EXEC モードに戻ります。
ステップ 5	**show running-config** または **show privilege**	設定を確認します。 **show running-config** 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。**show previlege** コマンドは、権限レベルの設定を表示します。
ステップ 6	**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。

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 コマンドで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れば、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	**enable** *level*	特定の権限レベルにログインします。 *level* に指定できる範囲は 0 ～ 15 です。
ステップ 2	**disable** *level*	特定の権限レベルを終了します。 *level* 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、Terminal Access Controller Access Control System Plus（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、詳細なアカウンティング情報を収集し、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、アカウンティング）を介して機能します。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*』Release 12.2 を参照してください。

ここでは、次の設定について説明します。

- TACACS+ の概要（p.9-11）
- TACACS+ の動作（p.9-13）
- TACACS+ の設定（p.9-13）
- TACACS+ 設定の表示（p.9-18）

TACACS+ の概要

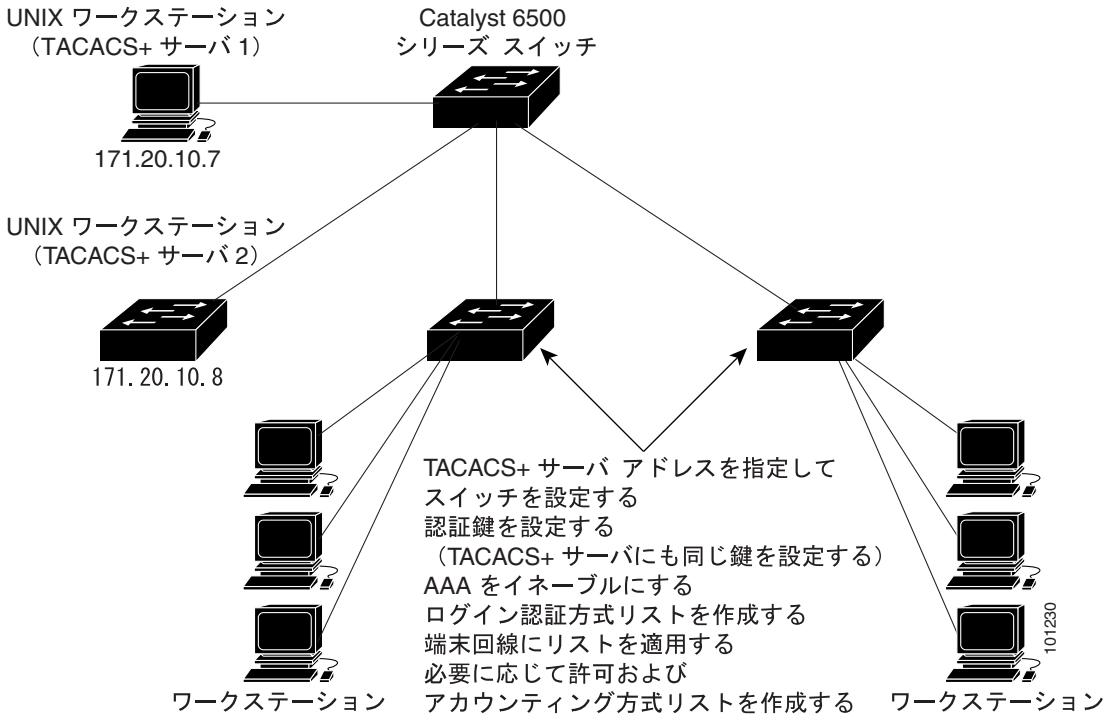
TACACS+ は、スイッチにアクセスしようとするユーザの検証を集中的に行うセキュリティ アプリケーションです。TACACS+ サービスは、通常 UNIX または Windows NT ワークステーション上で稼働する TACACS+ デーモンのデータベースで管理されます。スイッチに TACACS+ 機能を設定するには、TACACS+ サーバにアクセスして設定する必要があります。

（注）　スイッチ スタックと TACACS+ サーバとの間では冗長接続を確立することを推奨します。これは、接続されているスタック メンバーがスイッチ スタックから削除された場合でも、TACACS+ サーバがアクセス可能なまま維持されるようにする上で役立ちます。

TACACS+ は、個別のモジュール型認証、許可、およびアカウンティング機能を備えています。TACACS+ により、単一のアクセス制御サーバ（TACACS+ デーモン）が AAA の各サービスを個別に行うことができます。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式を提供することです。スイッチは、他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 サーバにすることができます。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 9-1 を参照）。

図 9-1　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、社会保険番号など数種類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することができます。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［ポイントツーポイント プロトコル］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+ を使用するスイッチに簡易 ASCII ログインを試行し、認証を要求すると、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するのに十分な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、またはデーモンとスイッチ間のネットワーク接続時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、そのユーザおよびユーザがアクセスできるサービスの 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ティ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、次の設定について説明します。

- TACACS+ のデフォルト設定（p.9-14）
- TACACS+ サーバ ホストの特定と認証鍵の設定（p.9-14）
- TACACS+ ログイン認証の設定（p.9-15）
- 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.9-17）
- TACACS+ アカウンティングの開始（p.9-18）

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。

セキュリティ失効の防止のため、ネットワーク管理アプリケーションで TACACS+ を設定することはできません。TACACS+ をイネーブルに設定した場合、CLI でスイッチにアクセスしたユーザが認証されます。

（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 ホストの特定と認証鍵の設定

スイッチを単一サーバまたは AAA サーバ グループを使用するように設定して、既存のサーバ ホストをグループ化して認証用ホストとして使用することができます。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	説明
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	**tacacs-server host** *hostname* [**port** *integer*] [**timeout** *integer*] [**key** *string*]	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。 • *hostname* には、ホストの名前または IP アドレスを指定します。 • （任意）**port** *integer* 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。 • （任意）**timeout** *integer* 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。 • （任意）**key** *string* 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復号化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。
ステップ 3	**aaa new-model**	AAA をイネーブルにします。
ステップ 4	**aaa group server tacacs+** *group-name*	（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	**server** *ip-address*	（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。

	コマンド	説明
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 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定義してから、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、自動的にすべてのポートに適用されます。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	説明
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	説明
ステップ 3	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]	ログイン認証方式リストを作成します。 • **login authentication** 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、**default** 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。 • *list-name* 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。 • *method1...* 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。 次のいずれかの方式を選択します。 • **enable** — 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、**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。 • **group tacacs+** — 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、「TACACS+ サーバホストの特定と認証鍵の設定」(p.9-14) を参照してください。 • **line** — 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。**password** *password* ライン コンフィギュレーション コマンドを使用します。 • **local** — 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。**username** *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。 • **local-case** — 大文字と小文字が区別されるローカル ユーザ名データベースを認証に使用します。**username** *name* **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。 • **none** — ログインに認証を使用しません。
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*}	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。 • **default** を指定する場合は、**aaa authentication login** コマンドで作成したデフォルトのリストを使用します。 • *list-name* には、**aaa authentication login** コマンドで作成したリストを指定します。
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 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、認証されたユーザの場合、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	**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 authorization network tacacs+**	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	**aaa authorization exec tacacs+**	ユーザにイネーブル EXEC アクセスがある場合に、ユーザ TACACS+ 許可をスイッチに設定します。 **exec** キーワードを指定すると、ユーザ プロファイル情報（**autocommand** 情報など）が返されることがあります。
ステップ 4	**end**	イネーブル EXEC モードに戻ります。
ステップ 5	**show running-config**	設定を確認します。
ステップ 6	**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。

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アトリビュートと値（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	**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 accounting network start-stop tacacs+**	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	**aaa accounting exec start-stop tacacs+**	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	**end**	イネーブル EXEC モードに戻ります。
ステップ 5	**show running-config**	設定を確認します。
ステップ 6	**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。

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、詳細なアカウンティング情報を収集し、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*』Release 12.2 を参照してください。

ここでは、次の設定について説明します。

- RADIUS の概要（p.9-19）
- RADIUS の動作（p.9-21）
- RADIUS の設定（p.9-21）
- RADIUS 設定の表示（p.9-32）

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製ルータおよびスイッチ上で稼働し、中央 RADUIS サーバに認証要求を送信します。中央 RADUIS 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。RADUIS 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。

（注）　スイッチ スタックと RADIUS サーバとの間では冗長接続を確立することを推奨します。これは、接続されているスタック メンバーがスイッチ スタックから削除された場合でも、RADIUS サーバがアクセス可能なまま維持されるように保証する上で役立ちます。

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。

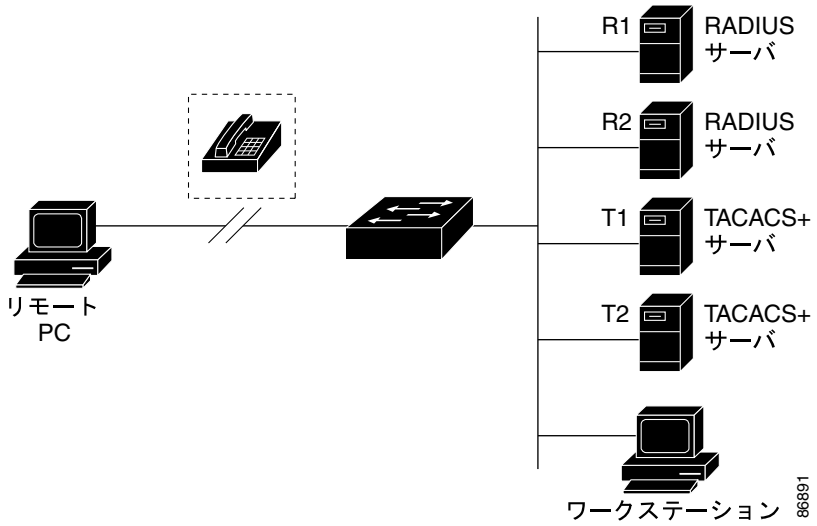
- それぞれが RADIUS をサポートする、マルチベンダー アクセス サーバによるネットワーク。たとえば、複数のベンダーのアクセス サーバが、1 つの RADIUS サーバベース セキュリティ データベースを使用します。複数のベンダーのアクセス サーバからなる IP ベースのネットワークでは、ダイヤルイン ユーザは RADUIS サーバを通じて認証されます。RADUIS サーバは、Kerberos セキュリティ システムで動作するようにカスタマイズされています。
- アプリケーションが RADIUS プロトコルをサポートするターンキー ネットワーク セキュリティ環境。たとえば、*スマート カード*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 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 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 9-2（p.9-20）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 10 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。RADUIS 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; NetBIOS フレーム制御プロトコル）、NetWare Asynchronous Services Interface（NASI;NetWare 非同期サポート インターフェイス）、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す。
- 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 9-2　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ティングの方式リストも定義できます。

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、次の設定について説明します。

- RADIUS のデフォルト設定（p.9-22）
- RADIUS サーバ ホストの特定（p.9-22）（必須）
- RADIUS ログイン認証の設定（p.9-24）（必須）
- AAA サーバ グループの定義（p.9-26）（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.9-28）（任意）
- RADIUS アカウンティングの開始（p.9-29）（任意）
- すべての RADIUS サーバに対する設定（p.9-30）（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.9-30）（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.9-32）（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で 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
- アカウンティング宛先ポート
- 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ティング サービスを提供できない場合、スイッチは、「%RADIUS-4-RADIUS_DEAD」メッセージを出したあとで、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.9-30）を参照してください。

AAA サーバ グループを使用して、既存のサーバ ホストを認証用としてグループ化するよう、スイッチを設定することができます。詳細については、「AAA サーバ グループの定義」（p.9-26）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	**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	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]	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。 • （任意）**auth-port** *port-number* には、認証要求の UDP 宛先ポートを指定します。 • （任意）**acct-port** *port-number* には、アカウンティング要求の UDP 宛先ポートを指定します。 • （任意）**timeout** *seconds* 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 です。この設定は、**radius-server timeout** 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。**radius-server host** コマンドでタイムアウトを設定しない場合は、**radius-server timeout** コマンドの設定が使用されます。 • （任意）**retransmit** *retries* 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。**radius-server host** コマンドで再送信の値を設定しない場合は、**radius-server retransmit** 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。 • （任意）**key** *string* 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鍵を指定します。 **（注）** 鍵は、RADIUS サーバ上で使用する暗号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 **radius-server host** 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。 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、および暗号鍵の値を設定します。
ステップ 3	**end**	イネーブル EXEC モードに戻ります。
ステップ 4	**show running-config**	設定を確認します。
ステップ 5	**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 サーバをアカウンティ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ティ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。これらの設定には、スイッチの IP アドレス、およびサーバとスイッチで共用するキー ストリングが含まれます。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定義してから、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、自動的にすべてのポートに適用されます。

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。この手順は必須です。

	コマンド	説明
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	説明
ステップ 3	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]	ログイン認証方式リストを作成します。 • **login authentication** 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、**default** 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。 • *list-name* 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。 • *method1...* 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。 次のいずれかの方式を選択します。 – **enable** — 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、**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。 – **group radius** — 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、「RADIUS サーバ ホストの特定」（p.9-22）を参照してください。 – **line** — 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。**password** *password* ライン コンフィギュレーション コマンドを使用します。 – **local** — 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。**username** *name* **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。 – **local-case** — 大文字と小文字が区別されるローカル ユーザ名データベースを認証に使用します。**username** *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。 – **none** — ログインに認証を使用しません。
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*}	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。 • **default** を指定する場合は、**aaa authentication login** コマンドで作成したデフォルトのリストを使用します。 • *list-name* には、**aaa authentication login** コマンドで作成したリストを指定します。
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 エントリを同じサービス（たとえば、アカウンティング）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	**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	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]	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。 • （任意）**auth-port** *port-number* には、認証要求の UDP 宛先ポートを指定します。 • （任意）**acct-port** *port-number* には、アカウンティング要求の UDP 宛先ポートを指定します。 • （任意）**timeout** *seconds* 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 です。この設定は、**radius-server timeout** 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。**radius-server host** コマンドでタイムアウトを設定しない場合は、**radius-server timeout** コマンドの設定が使用されます。 • （任意）**retransmit** *retries* 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。**radius-server host** コマンドで再送信の値を設定しない場合は、**radius-server retransmit** 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。 • （任意）**key** *string* 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鍵を指定します。 **（注）** 鍵は、RADIUS サーバ上で使用する暗号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 **radius-server host** 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。 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、および暗号鍵の値を設定します。
ステップ 3	**aaa new-model**	AAA をイネーブルにします。
ステップ 4	**aaa group server radius** *group-name*	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 コンフィギュレーション モードになります。
ステップ 5	**server** *ip-address*	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。 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	**end**	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	**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。
ステップ 9		RADIUS ログイン認証をイネーブルにします。「RADIUS ログイン認証の設定」（p.9-24）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** キーワードを指定して使用すると、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。

	コマンド	説明
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 authorization network radius**	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	**aaa authorization exec radius**	ユーザにイネーブル EXEC アクセスがある場合に、ユーザ RADIUS 許可をスイッチに設定します。 **exec** キーワードを指定すると、ユーザ プロファイル情報（**autocommand** 情報など）が返されることがあります。
ステップ 4	**end**	イネーブル EXEC モードに戻ります。
ステップ 5	**show running-config**	設定を確認します。
ステップ 6	**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。

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アトリビュートと値（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	**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 accounting network start-stop radius**	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	**aaa accounting exec start-stop radius**	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します。
ステップ 4	**end**	イネーブル EXEC モードに戻ります。
ステップ 5	**show running-config**	設定を確認します。
ステップ 6	**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。

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の設定を行うには、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	**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	**radius-server key** *string*	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鍵と照合する必要のあるテキスト 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	**radius-server retransmit** *retries*	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	**radius-server timeout** *seconds*	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	**radius-server deadtime** *minutes*	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	**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。

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アトリビュートと-値（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 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: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	**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	**radius-server vsa send** [**accounting** | **authentication**]	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。 • （任意）**accounting** 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ティング アトリビュートに限定します。 • （任意）**authentication** 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。 キーワードなしでこのコマンドを入力すると、アカウンティングおよび認証の両方のベンダー固有アトリビュートが使用されます。
ステップ 3	**end**	イネーブル EXEC モードに戻ります。
ステップ 4	**show running-config**	設定を確認します。
ステップ 5	**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。

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	**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	**radius-server host** {*hostname* | *ip-address*} **non-standard**	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	**radius-server key** *string*	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。 **（注）** 鍵は、RADIUS サーバ上で使用する暗号鍵と照合する必要のあるテキスト 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 4	**end**	イネーブル EXEC モードに戻ります。
ステップ 5	**show running-config**	設定を確認します。
ステップ 6	**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。

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介してネットワーク 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）バージョンをスイッチにインストールしておく必要があります。

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には、許可を取得する必要があります。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、以下について説明します。

- Kerberos の概要（p.9-33）
- Kerberos の動作（p.9-35）
- Kerberos の設定（p.9-37）

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/

（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、『*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

（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 3750 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 はマサチューセッツ工科大学（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; データ暗号化規格）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）といいます。

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 サーバとして設定可能で、Kerberos プロトコルを用いてユーザを認証できる Catalyst 3750 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
- rsh（リモート シェル プロトコル）

表 9-2 に、一般的な Kerberos 関連用語とその定義を示します。

表 9-2　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（TGT[1] 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	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、*user@REALM* という形式です（たとえば、smith@EXAMPLE.COM）。Kerberos インスタンスのある Kerberos プリンシパルは、*user/instance@REALM* という形式です（たとえば、smith/admin@EXAMPLE.COM）。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の小文字でなければなりません。 **（注）** Kerberos レルム名はすべて大文字でなければなりません。
KDC[2] 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 9-2　Kerberos の用語（続き）

用語	定義
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*。
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 サーバにクエリーを送信して、他のネットワーク サービスの認証を得ます。
KEYTAB[3]	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 を使って暗号化されたサービス証明書を復号して認証します。Kerberos 5 以前のバージョンでは、KEYTAB は SRVTAB[4] といいます。
プリンシパル	Kerberos ID ともいい、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 とも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 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。

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

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 プロトコルを用いてリモート ユーザを認証できる Catalyst 3750 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 サーバとしての Catalyst 3750 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.9-36）
2. KDC からの TGT の取得（p.9-36）
3. ネットワーク サービスに対する認証の取得（p.9-36）

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第一のセキュリティ 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 の復号を試行します。
   - 復号に成功したら、ユーザはスイッチに対して認証を得ます。
   - 復号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第二のセキュリティ 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/fsecsp/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第三のセキュリティ レイヤについて説明します。TGT を持つユーザは、ここで Kerberos レルム内のネットワーク 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/fsecsp/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 レルム内のホストと 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しなければなりません。これを実行するために、互いを識別する必要があります。KDC 上の Kerberos データベースにホストのエントリを追加し、Kerberos レルム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、次のガイドラインに従ってください。

- 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 レルム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 プロトコルを用いてユーザを認証できる Catalyst 3750 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

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ティング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	**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。
ステップ 3	**aaa authentication login default local**	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。**default** 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	**aaa authorization exec local**	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します。
ステップ 5	**aaa authorization network local**	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	**username** *name* [**privilege** *level*] {**password** *encryption-type password*}	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確立します。 ユーザごとにこのコマンド入力を繰り返します。 • *name* 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。 • （任意）*level* 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。 • *encryption-type* 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。 • *password* 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、**username** コマンドの最後のオプションとして指定します。
ステップ 7	**end**	イネーブル EXEC モードに戻ります。
ステップ 8	**show running-config**	設定を確認します。
ステップ 9	**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。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には、許可を取得する必要があります。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、以下について説明します。

- SSH の概要（p.9-39）
- SSH の設定（p.9-40）
- SSH の設定およびステータスの表示（p.9-43）

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

（注）　この章で使用される構文および使用方法の詳細については、現リリースのコマンド リファレンスおよび次の URL から 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。

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.9-40）
- 制限事項（p.9-40）

（注）　IP ベース イメージ（以前の標準マルチレイヤ イメージ [SMI]）ソフトウェアまたは IP サービス イメージ（以前の拡張マルチレイヤ イメージ [EMI]）ソフトウェアの暗号化バージョンを実行するスタック マスターで障害が発生し、ソフトウェアの非暗号化バージョンを実行するスイッチに置き換わった場合、スイッチ スタックへの SSH 接続は失われることがあります。IP ベース イメージ ソフトウェアまたは IP サービス イメージ ソフトウェアの暗号化バージョンを実行するスイッチをスタック マスターにすることを推奨します。スタック マスターが IP ベース イメージ ソフトウェアまたは IP サービス イメージ ソフトウェアの非暗号化バージョンを実行している場合、暗号化機能は使用できません。

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。

スイッチは、SSHv1 サーバまたは 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、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、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。

また SSH も、次のユーザ認証方式をサポートします。

- TACACS+（詳細については、「TACACS+ によるスイッチ アクセスの制御」[p.9-11] 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.9-19] 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.9-38] 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.9-40）
- スイッチでの SSH 実行の設定（p.9-41）（必須）
- SSH サーバの設定（p.9-42）（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。
- SSH サーバがスタック マスター上で稼働中に、スタック マスターに障害が生じた場合、新しいスタック マスターは直前のスタック マスターによって生成された RSA 鍵を使用します。

- **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.9-41）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、この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.9-38）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	**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	**hostname** *hostname*	スイッチにホスト名を設定します。
ステップ 3	**ip domain-name** *domain_name*	スイッチにホスト ドメインを設定します。
ステップ 4	**crypto key generate rsa**	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	**end**	イネーブル EXEC モードに戻ります。
ステップ 6	**show ip ssh** または **show ssh**	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	**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。

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	**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	**ip ssh version** [**1** | **2**]	(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。 • **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。 • **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。 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。
ステップ 3	**ip ssh** {**timeout** *seconds* | **authentication-retries** *number*}	SSH 制御パラメータを設定します。 • 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。 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 4 に対してセッション 0)。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。 • 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。 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。
ステップ 4	**end**	イネーブル EXEC モードに戻ります。
ステップ 5	**show ip ssh** または **show ssh**	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 サーバ接続のステータスを表示します。
ステップ 6	**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。

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 の設定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 9-3 に示されるイネーブル EXEC コマンドの 1 つまたは複数を使用します。

表 9-3　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
show ip ssh	SSH サーバのバージョンおよび設定情報を表示します。
show ssh	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/ios122/122cgcr/fsecur_r/fothercr/srfssh.htm.

スイッチの SSL HTTP の設定

ここでは、Secure Socket Layer（SSL）バージョン 3.0 の HTTP 1.1 サーバおよびクライアントに対するサポートを設定する方法について説明します。SSL は HTTP クライアント認証のほかに、サーバ認証、暗号化、およびメッセージ保全を行い、セキュア HTTP 通信を実現します。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には、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、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、以下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.9-44）
- セキュア HTTP サーバおよびセキュア HTTP クライアントの設定（p.9-46）
- セキュア HTTP サーバおよびセキュア HTTP クライアント ステータスの表示（p.9-50）

この章で使用されるコマンドの設定例および構文と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」機能解説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/ftsslsht.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 サーバの入出データは暗号化されてからインターネットに送信されます。SSL 暗号化機能付き HTTP は、Web ブラウザからのスイッチの設定などの機能を実現するセキュア接続を提供します。シスコによるセキュア HTTP サーバおよびセキュア HTTP クライアントの実装機能は、アプリケーションレイヤの暗号化による SSL バージョン 3.0 の実装機能を使用します。HTTP over SSL は、省略して HTTPS といわれます。セキュア接続の URL は、http:// ではなく https:// で始まります。

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）上の HTTPS 要求を受信して、この要求を HTTP 1.1 Web サーバに渡すことです。HTTP 1.1 サーバは要求を処理して、HTTP セキュア サーバに応答（ページ）を戻します。入れ替わりに、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 User Agent サービスに対する Cisco IOS アプリケーション要求に応答して、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戻すことです。

CA の信頼点

Certificate Authority（CA; 認証局）は証明書要求を管理して、加入しているネットワーク デバイスに証明書を発行する管理局です。このサービスにより、加入しているデバイスは中央集中型のセキュリティ鍵および証明書管理が提供されます。特定の CA サーバを、*信頼点*と呼びます。

接続が試行されると、HTTPS サーバが指定の CA 信頼点から取得した認証済み X.509v3 証明書を発行することにより、クライアントにセキュア接続を提供します。クライアント（通常、Web ブラウザ）は次に、証明書を認証できる公開鍵を所有します。

セキュア HTTP 接続には、CA 信頼点を設定することを強く推奨します。HTTPS サーバが稼働するデバイスに CA 信頼点が設定されていなければ、サーバは自動認証して、必要とされる RSA 鍵ペアを生成します。自己認証（自己署名）証明書では十分なセキュリティが提供されないため、接続ク

ライアントは証明書が自己認証されたものであるという通知を生成し、ユーザがこの接続を許可または拒否するよう選択できます。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）に便利です。

CA 信頼点を設定していない場合にセキュア 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）用の一時的または永続的に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およびドメイン名が設定されていない場合は、一時的に自己署名証明書が生成されます。スイッチが再起動する場合、一時的に自己署名証明書は失われ、新たに一時的に自己署名された新しい証明書が割り当てられます。
- スイッチにホスト名およびドメイン名が設定されている場合、永続的に自己署名証明書が生成されます。この証明書は、スイッチの再起動後またはセキュア HTTP サーバをディセーブルにしてもアクティブのままで、セキュア HTTP 接続を再度イネーブルにした場合もそのままです。

自己署名証明書が生成された場合、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。次に、自己署名証明書を表示するコマンド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)
```

この自己署名証明書を削除するには、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 コマンドを入力します。セキュア HTTP サーバをあとで再度イネーブルにすると、新たに自己署名証明書が生成されます。

（注） *TP self-signed* のあとの値は、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 サーバがクライアントからの X.509v3 証明書を要求できるようにします。クライアントの認証は、サーバの自己認証よりも高いセキュリティを実現します。

CA 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 サーバに接続する際、クライアント Web ブラウザは、サポート対象の CipherSuite のリストを提供します。クライアントおよびサーバは、リスト上の両方のサポート対象 CipherSuite から使用するのに最適な暗号化アルゴリズムをネゴシエートします。たとえば、Netscape Communicator 4.76 は RSA Public Key Cryptography、MD2、MD5、RC2-CBC、RC4、DES-CBC、および DES-EDE3-CBC を使用して U.S. セキュリティをサポートします。

可能な限り最適な暗号化を実現するためには、Microsoft Internet Explorer Version 5.5（またはそれ以降）または Netscape Communicator Version 4.76（またはそれ以降）などの 128 ビットの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 ビットの暗号化を行わないため、他の CipherSuite と同程度のセキュリティは提供されません。

CipherSuite の安全性および複雑性が高度になると、処理時間が多少長く必要になります。次に、スイッチがサポートする CipherSuite を定義して、ルータ処理ロード（速度）が速いものから遅いものへランク付けしたリストを示します。

1. SSL_RSA_WITH_DES_CBC_SHA — メッセージの暗号化には DES-CBC、メッセージ ダイジェストには SHA を使用する RSA 鍵交換（RSA Public Key Cryptography）
2. SSL_RSA_WITH_RC4_128_MD5 — メッセージ ダイジェストに RC4 128 ビット暗号化および MD5 を使用する RSA 鍵交換
3. SSL_RSA_WITH_RC4_128_SHA — メッセージ ダイジェストに RC4 128 ビット暗号化および SHA を使用する RSA 鍵交換
4. SSL_RSA_WITH_3DES_EDE_CBC_SHA — メッセージの暗号化には 3DES および DES-EDE3-CBC、メッセージ ダイジェストには SHA を使用する RSA 鍵交換

RSA（指定された暗号化とダイジェスト アルゴリズム コンビネーションの連携）は、鍵生成および SSL 接続上の認証の両方に使用されます。この使用方法は、CA 信頼点が設定されているかどうかには依存しません。

セキュア HTTP サーバおよびセキュア HTTP クライアントの設定

ここでは、次の設定について説明します。

- SSL のデフォルト設定（p.9-46）
- SSL 設定時の注意事項（p.9-47）
- CA 信頼点の設定（p.9-47）
- セキュア HTTP サーバの設定（p.9-48）
- セキュア HTTP クライアントの設定（p.9-49）

SSL のデフォルト設定

標準の HTTP サーバはイネーブルに設定されています。

SSL はイネーブルです。

CA 信頼点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 をスイッチ クラスタで使用する場合、SSL セッションはクラスタ コマンダで終了します。クラスタ メンバー スイッチは、標準 HTTP を実行する必要があります。

CA 信頼点を設定する前に、システム クロックが設定されているか確認する必要があります。クロックが設定されていない場合、証明書は不正な日時のため拒否されます。

Catalyst 3750 スイッチ スタックでは、SSL セッションはスタック マスターで終了します。

CA 信頼点の設定

セキュア HTTP 接続には、公式の CA 信頼点を設定することを強く推奨します。CA 信頼点は、自己署名証明書より安全です。

CA 信頼点を設定するには、イネーブル EXEC モードで次の手順を行います。

	コマンド	説明
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	**hostname** *hostname*	スイッチのホスト名（事前にホスト名を設定していない場合にのみ必要）を指定します。ホスト名は、セキュリティ鍵および証明書用に必要です。
ステップ 3	**ip domain-name** *domain-name*	スイッチの IP ドメイン名（事前に IP ドメイン名を設定していない場合にのみ必要）を指定します。ドメイン名は、セキュリティ鍵および証明書用に必要です。
ステップ 4	**crypto key generate rsa**	（任意）RSA 鍵ペアを生成します。スイッチの証明書を取得するには、RSA 鍵ペアが必要です。RSA 鍵ペアは、自動的に生成されます。必要な場合、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	**crypto ca trustpoint** *name*	CA 信頼点のローカル コンフィギュレーション名を指定して、CA 信頼点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	**enrollment url** *url*	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	**enrollment http-proxy** *host-name port-number*	（任意）HTTP プロキシ サーバを経由して 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	**crl query** *url*	ピアの証明書が失効していないことを確認するために、スイッチが Certificate Revocation List（CRL; 証明書失効リスト）を要求するよう設定します。
ステップ 9	**primary**	（任意）CA 要求のためのプライマリ（デフォルト）信頼点として使用する信頼点を指定します。
ステップ 10	**exit**	CA 信頼点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	**crypto ca authentication** *name*	CA の公開鍵を取得して、CA を認証します。ステップ 5 と同じ名前を使用します。
ステップ 12	**crypto ca enroll** *name*	指定された CA 信頼点から証明書を取得します。このコマンドにより、各 RSA 鍵ペアのサイン済み証明書が要求されます。
ステップ 13	**end**	イネーブル EXEC モードに戻ります。
ステップ 14	**show crypto ca trustpoints**	設定を確認します。
ステップ 15	**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。

すべての ID 情報および CA に対応付けられた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、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 信頼点を設定する必要があります。CA 信頼点が設定されていなければ、初めてセキュア HTTP サーバを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 HTTP サーバおよびセキュア HTTP サーバの両方に適用するオプション（パス、適用するアクセス リスト、最大接続数、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	**show ip http server status**	（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どうか判別するために、HTTP サーバのステータスを表示します。出力中の次のどちらかの行が表示されます。
		`HTTP secure server capability: Present`
		または
		`HTTP secure server capability: Not present`

ステップ 2	**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 3	**ip http secure-server**	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。デフォルトでは、HTTPS サーバはイネーブルに設定されています。
ステップ 4	**ip http secure-port** *port-number*	（任意）HTTPS サーバで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、443 です。有効なオプションは 443、または 1025 ～ 65535 の任意の数です。
ステップ 5	**ip http secure-ciphersuite** {[**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される 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は、サーバおよびクライアントが、両方のサポート対象 CipherSuite をネゴシエーションするように許可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	**ip http secure-client-auth**	（任意）HTTP サーバが、接続プロセス中に認証用にクライアントからの X.509v3 証明書を要求するように設定します。デフォルトでは、クライアントがサーバからの証明書を要求しますが、サーバはこのクライアントを認証しません。
ステップ 7	**ip http secure-trustpoint** *name*	CA 信頼点を指定して、X.509v3 セキュリティ証明書を取得し、クライアントの証明書接続を認証します。 **（注）** このコマンドの使用は、前述の手順に従って CA 信頼点が設定済みであることを仮定しています。
ステップ 8	**ip http path** *path-name*	（任意）HTML ファイル用のベースの HTTP パスを設定します。パスは、ローカル システム（通常、システム フラッシュ メモリに存在する）上の HTTP サーバ ファイルの場所を指定します。
ステップ 9	**ip http access-class** *access-list-number*	（任意）HTTP サーバへのアクセスを許可するのに使用するアクセス リストを指定します。
ステップ 10	**ip http max-connections** *value*	（任意）HTTP サーバに許可された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、デフォルトは 5 です。

	コマンド	説明
ステップ 11	**ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value*	（任意）定義された環境で、HTTP サーバへの接続をオープンにしておくことができる時間を指定します。 • **idle —** データを受信していないか、応答データを送信できない最大時間。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。 • **life —** 接続を確立してからの最大時間。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。 • **requests —** 持続接続で処理される最大要求数。最大値は 86400 です。デフォルト値は 1 です。
ステップ 12	**end**	イネーブル EXEC モードに戻ります。
ステップ 13	**show ip http server secure status**	HTTP セキュア サーバのステータスを表示して、設定を確認します。
ステップ 14	**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。

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。デフォルトの設定に戻すには、**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。クライアント認証に関する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。この 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は、URL に続いてポート番号を指定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。セキュア HTTP クライアントの認証には、CA が必要です。この手順では、事前にスイッチ上に CA 信頼点が設定されていると仮定しています。CA 信頼点が設定されていなくて、リモート HTTPS サーバでクライアント認証が必要な場合、セキュア HTTP クライアントへの接続は失敗します。

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	説明
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	**ip http client secure-trustpoint** *name*	（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、使用される CA 信頼点を指定します。このコマンドの使用は、前述の手順に従って CA 信頼点が設定済みであることを仮定しています。クライアント認証が必要でない場合、またはプライマリ信頼点が設定されている場合には、このコマンドはオプションになります。

	コマンド	説明
ステップ 3	**ip http client secure-ciphersuite** {[**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される 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は、サーバおよびクライアントが、両方のサポート対象の CipherSuite をネゴシエートするように許可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	**end**	イネーブル EXEC モードに戻ります。
ステップ 5	**show ip http client secure status**	HTTP セキュア サーバのステータスを表示して、設定を確認します。
ステップ 6	**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。

クライアントの信頼点設定を削除するには、**no ip http client secure-trustpoint** *name* を使用します。クライアントに事前設定された 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 サーバおよびセキュア HTTP 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、次のイネーブル EXEC コマンドを使用します（表 9-4 を参照）。

表 9-4　SSH 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
show ip http client secure status	HTTP セキュア クライアントの設定を表示します。
show ip http server secure status	HTTP セキュア サーバの設定を表示します。
show running-config	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。

SCP のためのスイッチ設定

Secure Copy Protocol（SCP）機能では、スイッチ設定またはスイッチ イメージ ファイルをセキュアな認証方法でコピーすることができます。SCP は、SSH に依存します。SSH は、Berkeley r-tool をセキュアにしたアプリケーションおよびプロトコルです。

SSH を動作させるには、RSA の公開鍵と秘密鍵のペアがスイッチで必要です。セキュアな転送のために SSH に依存している SCP も、同様です。

SSH は AAA 認証にも依存しており、SCP も AAA 認証に依存しているので、正しい設定が必要になります。

- SCP をイネーブルにする前に、SSH、認証、許可をスイッチで正しく設定する必要があります。
- SCP はセキュアな転送のために SSH に依存するので、ルータでは RSA 鍵ペアが必要です。

（注）　SCP を使用する場合は、copy コマンドにパスワードを入力することはできません。プロンプトが表示されてから、パスワードを入力する必要があります。

セキュア コピーについて

セキュア コピー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 Berkeley r-tools スイートに由来する RCP に似ていますが、SCP のセキュリティは SSH に依存します。また、SCP では AAA 認証が設定されていなければならないため、ルータはユーザが適切な権限レベルを持っているかどうかを判断することができます。

適切な権限を持つユーザは SCP を使用して、Cisco IOS File System（IFS）とスイッチ間で **copy** コマンドにより、ファイルをコピーすることができます。許可された管理者は、ワークステーションからコピーすることもできます。

SCP の設定方法および確認方法については、次の URL にある『*Cisco IOS New Features*』Cisco IOS Release 12.2 の「Secure Copy Protocol」の章を参照してください。

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t2/ftscp.htm