CHAPTER 4

WMIC の管理

このマニュアルでは、Cisco Wireless Mobile Interface（WMIC; ワイヤレス モバイル インターフェイス カード）を管理する方法について説明します。

システム名およびプロンプトの設定

WMIC に、識別するためのシステム名を設定します。大なり記号 [>] が付加されます。**prompt** コマンドをグローバル コンフィギュレーション モードで使用して手動でプロンプトを設定していないかぎり、システム名が変更されると、プロンプトも更新されます。

（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 Configuration Fundamentals Command Reference*』および『*Cisco IOS IP and IP Routing Command Reference for Release 12.1*』を参照してください。

システム名の設定

手動でシステム名を設定するには、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	**hostname** *name*	手動でシステム名を設定します。 名前は ARPANET ホスト名の規則に従う必要があります。この規則ではホスト名は文字で始まり、文字または数字で終わります。その間には文字、数字、またはハイフンのみを使用できます。名前には 63 文字まで使用できます。
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。

設定されたシステム名は、システム プロンプトにも使用されます。

デフォルトのホスト名に戻すには、**no 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。

DNS の管理

Domain Name System（DNS; ドメイン ネーム システム）プロトコルは、ホスト名を IP アドレスに対応付ける場合に使用できる分散型データベース DNS を制御します。WMIC に DNS を設定すると、**ping**、**telnet**、**connect** などのすべての IP コマンドを使用する場合や、関連する Telnet サポート操作時に、IP アドレスの代わりにホスト名を使用できます。

IP によって定義される階層型の命名方式では、デバイスを場所またはドメインで識別できます。ドメイン名の区切り文字には、ピリオド（.）を使用します。たとえば、シスコシステムズは *com* というドメイン名で識別される商業組織なので、ドメイン名は *cisco.com* です。

ドメイン名を追跡するために、IP ではドメイン ネーム サーバという概念が定義されています。ドメイン ネーム サーバでは、IP アドレスに対応する名前がキャッシュ（またはデータベース）に保管されています。ドメイン名を IP アドレスに対応付けるには、ホスト名を識別し、ネットワーク上に存在するネーム サーバを指定し、DNS をイネーブルにします。

DNS のデフォルト設定

表 4-1 に、DNS のデフォルト設定を示します。

表 4-1　DNS のデフォルト設定

機能	デフォルトの設定
DNS イネーブル ステート	ディセーブル
DNS デフォルト ドメイン名	設定なし
DNS サーバ	ネーム サーバのアドレスの設定なし

DNS の設定

DNS を使用するように WMIC 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	**ip domain-name** *name*	未修飾のホスト名（ドット付き 10 進表記ドメイン名のない名前）を完成させるためにソフトウェアが使用するデフォルトのドメイン名を定義します。 ドメイン名と未修飾ホスト名を区切る最初のピリオドは入力しないでください。 起動時にドメイン名は設定されていませんが、BOOTP または Dynamic Host Configuration Protocol（DHCP）サーバからコンフィギュレーションが取得されている場合は、BOOTP または DHCP サーバによってデフォルトのドメイン名が設定されることがあります（DHCP サーバにこの情報が設定されている場合）。
ステップ 3	**ip name-server** *server-address1* [*server-address2 ... server-address6*]	名前およびアドレスの解決に使用する 1 台以上のネーム サーバのアドレスを指定します。 最大 6 つのネーム サーバを指定できます。各サーバ アドレスはスペースで区切ります。最初に指定されたサーバが、プライマリ サーバです。WMIC は、まずプライマリ サーバに DNS クエリーを送信します。そのクエリーに失敗した場合は、バックアップ サーバにクエリーを送信します。

	コマンド	目的
ステップ 4	**ip domain-lookup**	（任意）WMIC 上で、DNS に基づくホスト名からアドレスへの変換をイネーブルにします。この機能は、デフォルトでイネーブルです。 ご使用のネットワーク デバイスを、名前の割り当てを制御できないネットワーク内のデバイスと接続する必要がある場合は、グローバルなインターネット命名方式（DNS）を使用することにより、デバイスを一意に識別するデバイス名を動的に割り当てることができます。
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。

WMIC の IP アドレスをホスト名として使用する場合は、IP アドレスが使用され、DNS クエリーは発生しません。ピリオド（.）を含まないホスト名を設定すると、ピリオドと、それに続くデフォルトのドメイン名がホスト名に追加され、そのあとで DNS クエリーが行われ、名前が IP アドレスに対応付けられます。

デフォルトのドメイン名は、**ip domain-name** グローバル コンフィギュレーション コマンドによって設定される値です。ホスト名にピリオド（.）が含まれている場合、Cisco IOS ソフトウェアは、ホスト名にデフォルトのドメイン名を付加しないで、IP アドレスを検索します。

ドメイン名を削除するには、**no ip domain-name** *name* コマンドをグローバル コンフィギュレーション モードで使用します。ネーム サーバのアドレスを削除するには、**no ip name-server** *server-address* コマンドをグローバル コンフィギュレーション モードで使用します。WMIC の DNS をディセーブルにするには、**no ip domain-lookup** コマンドをグローバル コンフィギュレーション モードで使用します。

DNS 設定の表示

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

バナーの作成

MoTD（Message-of-The-Day）およびログイン バナーを設定できます。MOTD バナーはログイン時にすべての接続端末に表示されます。すべてのネットワーク ユーザに関連したメッセージ（システム シャットダウン予告など）を送信する場合に便利です。

ログイン バナーも、すべての接続端末に表示されます。表示されるのは、MOTD バナーのあと、ログイン プロンプトが表示される前です。

（注）　ここで使用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 Configuration Fundamentals Command Reference for Release 12.2*』を参照してください。

バナーのデフォルト設定

MOTD およびログイン バナーは設定されていません。

MOTD ログイン バナーの設定

ユーザが WMIC にログインしたときに画面に表示される 1 行または複数行のメッセージ バナーを作成できます。

MOTD ログイン バナーを設定するには、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	**banner motd** *c message c*	MOTD メッセージを指定します。 *c* には、ポンド記号（#）などの任意の区切り文字を入力して、**Return** キーを押します。区切り文字はバナー テキストの始まりと終わりを表します。終わりの区切り文字のあとの文字列は廃棄されます。 *message* には、255 文字までのバナー メッセージを入力します。区切り文字はメッセージ内で使用できません。
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 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。

MOTD バナーを削除するには、**no banner motd** 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、ポンド（#）を開始および終了の区切り文字として使用して、WMIC の MOTD バナーを設定する例を示します。

```
bridge(config)# banner motd #
This is a secure site. Only authorized users are allowed.
For access, contact technical support.
#
bridge(config)#
```

次に、上記設定によって表示されるバナーを示します。

```
Unix> telnet 172.2.5.4
Trying 172.2.5.4...
Connected to 172.2.5.4.
Escape character is '^]'.

This is a secure site. Only authorized users are allowed.
For access, contact technical support.

User Access Verification

Password:
```

ログイン バナーの設定

すべての接続端末に表示されるログイン バナーを設定できます。このバナーが表示されるのは、MOTD バナーのあと、ログイン プロンプトが表示される前です。

ログイン バナーを設定するには、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	**banner login** *c message c*	ログイン メッセージを指定します。 *c* には、ポンド記号（#）などの任意の区切り文字を入力して、**Return** キーを押します。区切り文字はバナー テキストの始まりと終わりを表します。終わりの区切り文字のあとの文字列は廃棄されます。 *message* には、255 文字までのバナー メッセージを入力します。区切り文字はメッセージ内で使用できません。
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 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。

ログイン バナーを削除するには、**no banner login** 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、ドル記号（$）を開始および終了の区切り文字として使用して、WMIC のログイン バナーを設定する例を示します。

```
bridge(config)# banner login $
Access for authorized users only. Please enter your username and password.
$
bridge(config)#
```

イネーブル EXEC コマンドへのアクセスの保護

ネットワーク内の端末へのアクセスを制御する簡単な方法は、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。権限レベルは、ネットワーク デバイスへのログイン後にユーザが使用できるかコマンドを定義します。

（注）　ここで使用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 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。

デフォルトのパスワードおよび権限レベルの設定

表 4-2 に、デフォルトのパスワードおよび権限レベルの設定を示します。

表 4-2　デフォルトのパスワードおよび権限レベル

機能	デフォルトの設定
ユーザ名およびパスワード	デフォルトのユーザ名は *Cisco* 、デフォルトのパスワードは *Cisco* です。
イネーブル パスワードおよび権限レベル	デフォルトのパスワードは *Cisco* です。デフォルトの権限レベルは 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 ファイル内で暗号化されています。
イネーブル シークレット パスワードおよび権限レベル	デフォルトのイネーブル パスワードは *Cisco* です。デフォルトの権限レベルは 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	デフォルトのパスワードは *Cisco* です。パスワードは、コンフィギュレーション ファイル内で暗号化されています。

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。

（注）　**no enable password** グローバル コンフィギュレーション コマンドを使用するとイネーブル パスワードが削除されます。このコマンドを使用する場合は、十分注意してください。イネーブル パスワードを削除すると、EXEC モードが開始されなくなります。

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	**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 モードにアクセスするためのパスワードを新規に定義するか、または既存のパスワードを変更します。 デフォルトのパスワードは *Cisco* 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。ストリングの先頭には数字を指定できません。ストリングには大文字と小文字の区別があり、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（?）をストリングに含めるには、パスワードを作成するときに、疑問符の前にキーの組み合わせ Ctrl-V を入力します。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。 **1.** **abc** を入力します。 **2.** **Crtl-V** を入力します。 **3.** **?123** を入力します。 イネーブル パスワードを入力するよう求められた場合は、疑問符の前に Ctrl-V を入力する必要はありません。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。 WMIC のコンフィギュレーション ファイルではイネーブル パスワードは暗号化されず、読み取ることができます。

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ず、レベル 15 のアクセス権が与えられます（従来のイネーブル EXEC モード アクセス）。

```
bridge(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、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** コマンドを使用できます。両方のコマンドは、暗号化パスワードを設定するという同じ働きをします。ユーザがイネーブル 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場合には、このパスワードを入力する必要があります。

enable secret コマンドで使用されている暗号化アルゴリズムの方が効率的なため、このコマンドを使用することを推奨します。

enable secret コマンドを設定した場合、このコマンドは **enable password** コマンドよりも優先されます。同時に 2 つのコマンドを有効にできません。

イネーブルの暗号化を設定し、シークレット パスワードの暗号化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	**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}	イネーブル EXEC モードにアクセスするためのパスワードを新規に定義するか、または既存のパスワードを変更します。
	または	または
	enable secret [**level** *level*] {*password* | *encryption-type encrypted*-password}	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。ストリングの先頭には数字を指定できません。ストリングには大文字と小文字の区別があり、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、暗号化パスワード（異なる WMIC コンフィギュレーションからコピーした暗号化パスワード）を指定する必要があります。 **（注）** 暗号化タイプを指定したにもかかわらず、クリア テキスト パスワードを入力した場合は、イネーブル EXEC モードを再開できません。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。 暗号化を行うと、コンフィギュレーション ファイル内でパスワードを読み取り不可能にできます。
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。

特定の権限レベルに対応する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、その権限レベルでアクセスする必要のあるユーザにのみパスワードを通知します。さまざまなレベルでアクセスできるコマンドを指定するには、グローバル コンフィギュレーション モードで **privilege level** 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.4-10）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*] コマンドをグローバル コンフィギュレーション モードで使用します。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** コマンドをグローバル コンフィギュレーション モードで使用します。

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

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

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、WMIC にローカルに保存できます。これら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。認証されたユーザは、WMIC にアクセスできます。権限レベルが定義されている場合は、ユーザ名とパスワードの各ペアに特定の権限レベル（および対応する権利および権限）を割り当てることもできます。

ログイン ユーザ名とパスワードを必要とする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。

	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	**username** *name* [**privilege** *level*] {**password** *encryption-type password*}	各ユーザのユーザ名、権限レベル、および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、アクセス後にユーザに設定する権限レベルを指定します。範囲は 0 ～ 15 です。レベル 15 は、イネーブル EXEC モードのアクセスを設定します。レベル 1 は、ユーザ EXEC モードのアクセスを設定します。 • *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 WMIC 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。パスワードには 1 ～ 25 文字を指定します（埋め込みスペースを含めることができます）。パスワードは **username** コマンドの最後のオプションとして指定する必要があります。
ステップ 3	**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いて行われます。
ステップ 4	**end**	イネーブル EXEC モードに戻ります。

	コマンド	目的
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。

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* コマンドをグローバル コンフィギュレーション モードで使用します。

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** コマンドをライン コンフィギュレーション モードで使用します。

（注）　少なくとも 1 つのユーザ名を設定する必要があります。また、WMIC に対する Telnet セッションを開始するには、ローカル ログインを設定する必要があります。唯一のユーザ名にユーザ名を入力しない場合は、WMIC を開始できません。

複数の権限レベルの設定

Cisco IOS ソフトウェアには、デフォルトで、ユーザ EXEC とイネーブル EXEC の 2 つのパスワード セキュリティ モードがあります。各モードには、最大 16 個の階層レベルからなるコマンドを設定できます。複数のパスワードを設定することにより、ユーザ グループ別に特定のコマンドへのアクセスを許可できます。

たとえば、**clear line** コマンドへのアクセスを多くのユーザに許可する場合は、このコマンドにレベル 2 のセキュリティを割り当てて、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** コマンドへアクセスできるユーザ数を少なくする場合は、このコマンドにレベル 3 のセキュリティを割り当てて、そのパスワードをより小さいユーザ グループに配布することもできます。

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	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、アクセスを制限するコマンドを指定します。

	コマンド	目的
ステップ 3	**enable password level** *level password*	権限レベルに対応するイネーブル 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。ストリングの先頭には数字を指定できません。ストリングには大文字と小文字の区別があり、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。

コマンドを特定の権限レベルに設定すると、構文がそのコマンドのサブセットとな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 route** 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、自動的に権限レベル 15 に設定されます。

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

次に、**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。

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

権限レベルへのログインおよび終了

指定された権限レベルにログインしたり、指定された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	目的
ステップ 1	**enable** *level*	指定された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*	指定された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。

無線 LAN の保護

ネットワークに対する不正アクセスを禁止するためのセキュリティを設定します。WMIC はワイヤレス デバイスであるため、建物の物理的な境界を越えて通信できます。次のような高度なセキュリティ機能を適用できます。

- ビーコンでブロードキャストされない一意の Service Set Identifier（SSID、「Service Set Identifier」を参照）
- Wired Equivalent Privacy（WEP）および WEP 機能（「暗号スイートおよび WEP」を参照）
- ダイナミック WEP 認証（認証タイプを参照）

VLAN の使用方法

無線 LAN 上の VLAN に SSID を割り当てます。無線 LAN で VLAN を使用しない場合は、SSID に割り当てることができるセキュリティ オプションは制限されます。これは、暗号化設定と認証タイプが対応付けられているためです。VLAN を使用しない場合は、インターフェイスに暗号化設定（WEP および暗号）が適用されます。1 つのインターフェイスに複数の暗号化設定を使用できません。

たとえば、VLAN がディセーブルの状態でスタティック WEP を使用する SSID を作成した場合は、Wi-Fi Protected Access（WPA）認証を使用する SSID を別途作成できません。使用される暗号化設定が異なるためです。SSID のセキュリティ設定が他の SSID と一致しない場合は、SSID を 1 つまたは複数削除して、不一致が生じないようにしてください。

Express Security のタイプ

表 4-3 に、SSID に割り当てることができる 4 つのセキュリティ タイプを示します。

表 4-3　SSID に割り当てることができるセキュリティ タイプ

セキュリティ タイプ	説明	イネーブル化されたセキュリティ機能
セキュリティなし	セキュリティが一番小さいオプションです。このオプションは、パブリック スペースで SSID を使用する場合に限って使用します。ネットワークへのアクセスを制限する VLAN に割り当てます。	なし
スタティック WEP キー	セキュリティなしよりもセキュリティが高いオプションです。ただし、スタティック WEP キーは攻撃に対して脆弱です。このオプションを設定する場合は、MAC アドレスに基づいてアクセス ポイントとのアソシエーションを制限してください。ネットワークに RADIUS サーバが配置されていない場合は、アクセス ポイントをローカル認証サーバとして使用することを検討してください。	必須の Web 暗号化、鍵管理なし、およびオープン認証。ルート アクセス ポイント モードでは、クライアント デバイスがこの SSID を使用して対応付けを行う場合、アクセス ポイント キーと一致する WEP キーを使用する必要があります。

表 4-3　SSID に割り当てることができるセキュリティ タイプ（続き）

セキュリティ タイプ	説明	イネーブル化されたセキュリティ機能
Extensible Authentication Protocol（EAP; 拡張認証プロトコル）認証	802.1x 認証（LEAP、PEAP、EAP-TLS、EAP-TTLS、EAP-GTC、およびその他の 802.1X/EAP に基づく製品）がイネーブルになります。ネットワークの認証サーバ（サーバ認証ポート 1645）に IP アドレスおよび共有シークレットを入力する必要があります。802.1x 認証ではダイナミック暗号鍵が提供されるため、WEP キーが必要ありません。	必須の 802.1x 認証。ルート アクセス ポイント モードでは、クライアント デバイスがこの SSID を使用して対応付けを行う場合、802.1x 認証を実行する必要があります。 EAP-FAST を使用して認証するように無線クライアントが設定されている場合は、EAP を使用したオープン認証も設定してください。EAP を使用したオープン認証を設定しないと、次の GUI 警告メッセージが表示されます。 WARNING: Network EAP is used for LEAP authentication only.If radio clients are configured to authenticate using EAP-FAST, Open Authentication with EAP should also be configured. Command-Line Interface（CLI; コマンドライン インターフェイス）を使用している場合は、次の警告メッセージが表示されます。 SSID CONFIG WARNING: [SSID]: If radio clients are using EAP-FAST, AUTH OPEN with EAP should also be configured.
WPA	WPA を使用すると、認証サーバのサービスによって認証されたユーザがデータベースに無線アクセスできるようになり、WEP で使用されるアルゴリズムよりも強力なアルゴリズムによってユーザの IP トラフィックが暗号化されます。EAP 認証と同様に、ネットワークの認証サーバ（サーバ認証ポート 1645）に IP アドレスおよび共有シークレットが必要です。 この設定では、暗号化暗号、Temporal Key Integrity Protocol（TKIP）、オープン認証と EAP、ネットワーク EAP 認証、鍵管理（WPA 必須）、RADIUS サーバ認証ポート 1645 を使用します。	必須の WPA 認証。この SSID を使用して対応付けを行うクライアント デバイスは、WPA 対応でなければなりません。 EAP-FAST を使用して認証するように無線クライアントが設定されている場合は、EAP を使用したオープン認証も設定してください。EAP を使用したオープン認証を設定しないと、次の GUI 警告メッセージが表示されます。 WARNING: Network EAP is used for LEAP authentication only.If radio clients are configured to authenticate using EAP-FAST, Open Authentication with EAP should also be configured. CLI を使用している場合は、次の警告メッセージが表示されます。 SSID CONFIG WARNING: [SSID]: If radio clients are using EAP-FAST, AUTH OPEN with EAP should also be configured.

セキュリティの設定例

ここで示す設定例は、次のとおりです。

セキュリティ SSID なしの例

no_security_ssid という名前の SSID を作成し、この SSID をビーコンに追加し、VLAN 10 に割り当てて、VLAN 10 をネイティブ VLAN として選択する設定例の一部を示します（2.4-GHz（802.11b/g）WMIC に適用する場合）。

```
Dot11 ssid no_security-ssid
    vlan 10
    authentication open
    guest-mode
interface Dot11Radio0
 no ip address
 no ip route-cache
!
ssid no_security-ssid
!
!
speed basic-1.0 basic-2.0 basic-5.5 6.0 9.0 basic-11.0 12.0 18.0 24.0 36.0 48.0 54.0
 rts threshold 4000
 station-role root
 infrastructure-client
 bridge-group 1
!
interface Dot11Radio0.10
 encapsulation dot1Q 10
 no ip route-cache
 bridge-group 10
 bridge-group 10 spanning-disabled
!
interface FastEthernet0
 no ip address
 no ip route-cache
 duplex auto
 speed auto
 bridge-group 1
!
interface FastEthernet0.10
      encapsulation dot1Q 10
      no ip address
      no ip route-cache
 duplex auto
 speed auto
 bridge-group 1
```

4.9 GHz WMIC に適用する場合：

```
hostname root
!
username Cisco password 7 02250D480809
ip subnet-zero
!
no aaa new-model
```

```
!
bridge irb
!
interface Dot11Radio0
 no ip address
 no ip route-cache
 !
 ssid test
    authentication open
    infrastructure-ssid
 !
 spacing 5 channel 4942
 speed basic-1.5 2.25 basic-3.0 4.5 basic-6.0 9.0 12.0 13.5
 power local 10
 station-role root
 infrastructure-client
 bridge-group 1
 bridge-group 1 spanning-disabled
!
interface FastEthernet0
 no ip address
 no ip route-cache
 duplex auto
 speed auto
 bridge-group 1
 bridge-group 1 spanning-disabled
!
interface BVI1
 ip address 192.1.1.2 255.255.255.0
 no ip route-cache
!
ip http server
no ip http secure-server
ip http help-path http://www.cisco.com/warp/public/779/smbiz/prodconfig/help/eag
ip radius source-interface BVI1
logging snmp-trap emergencies
logging snmp-trap alerts
logging snmp-trap critical
logging snmp-trap errors
logging snmp-trap warnings
bridge 1 route ip
!
!
!
line con 0
 exec-timeout 0 0
 transport preferred all
 transport output all
line vty 0 4
 login local
 transport preferred all
 transport input all
 transport output all
line vty 5 15
 login
 transport preferred all
 transport input all
 transport output all
!
end
```

スタティック WEP セキュリティの例

static_wep_ssid という名前の SSID を作成し、この SSID をビーコンから除外し、VLAN 20 に割り当てて、3 をキー スロットとして選択し、128 ビット鍵を入力する設定例の一部を示します。

```
 encryption vlan 20 key 3 size 128bit 7 4E78330C1A841439656A9323F25A transmit-key
 encryption vlan 20 mode wep mandatory
 !
Dot11 ssid static_wep_ssid
    vlan 20
    authentication open
interface Dot11Radio0
 no ip address
 no ip route-cache
 !
ssid static_wep_ssid
 !
speed basic-1.0 basic-2.0 basic-5.5 6.0 9.0 basic-11.0 12.0 18.0 24.0 36.0 48.0 54.0
 rts threshold 4000
 station-role root
 infrastructure-client
 bridge-group 1
!
interface Dot11Radio0.20
 encapsulation dot1Q 20
 no ip route-cache
 bridge-group 20
 bridge-group 20 spanning-disabled
!
interface FastEthernet0
 no ip address
 no ip route-cache
 duplex auto
 speed auto
 bridge-group 1
!
interface FastEthernet0.20
 encapsulation dot1Q 20
 no ip route-cache
 bridge-group 20
 bridge-group 20 spanning-disabled
```

EAP 認証セキュリティの例

eap_ssid という名前の SSID を作成し、この SSID をビーコンから除外し、VLAN 30 に割り当てる設定例の一部を示します。

```
encryption vlan 30 mode wep mandatory
 !
Dot11 ssid eap_ssid
    vlan 30
    authentication open eap eap_methods
    authentication network-eap eap_methods
interface Dot11Radio0
 no ip address
 no ip route-cache
 !
ssid eap_ssid
 !
 speed basic-1.0 basic-2.0 basic-5.5 basic-11.0
 rts threshold 2312
 station-role root
```

```
 bridge-group 1
 bridge-group 1 subscriber-loop-control
 bridge-group 1 block-unknown-source
 no bridge-group 1 source-learning
 no bridge-group 1 unicast-flooding
 bridge-group 1 spanning-disabled
!
interface Dot11Radio0.30
 encapsulation dot1Q 30
 no ip route-cache
 bridge-group 30
 bridge-group 30 subscriber-loop-control
 bridge-group 30 block-unknown-source
 no bridge-group 30 source-learning
 no bridge-group 30 unicast-flooding
 bridge-group 30 spanning-disabled
!
interface FastEthernet0
 mtu 1500
 no ip address
 ip mtu 1564
 no ip route-cache
 duplex auto
 speed auto
 bridge-group 1
 no bridge-group 1 source-learning
 bridge-group 1 spanning-disabled
!
interface FastEthernet0.30
 mtu 1500
 encapsulation dot1Q 30
 no ip route-cache
 bridge-group 30
 no bridge-group 30 source-learning
 bridge-group 30 spanning-disabled
!
```

WPA セキュリティの例

wpa_ssid という名前の SSID を作成し、この SSID をビーコンから除外し、VLAN 40 に割り当てる設定例の一部を示します。

```
aaa new-model
!
aaa group server radius rad_eap
 server 10.91.104.92 auth-port 1645 acct-port 1646
!
aaa group server radius rad_mac
!
aaa group server radius rad_acct
!
aaa group server radius rad_admin
!
aaa group server tacacs+ tac_admin
!
aaa group server radius rad_pmip
!
aaa group server radius dummy
!
aaa authentication login eap_methods group rad_eap
aaa authentication login mac_methods local
aaa authorization exec default local
aaa authorization ipmobile default group rad_pmip
```

```
aaa accounting network acct_methods start-stop group rad_acct
aaa session-id common
!
!
bridge irb
!
!
interface Dot11Radio0
 no ip address
 no ip route-cache
 !
 encryption vlan 40 mode ciphers tkip
!
 ssid wpa_ssid
    vlan 40
    authentication open eap eap_methods
    authentication network-eap eap_methods
    authentication key-management wpa
!
 concatenation
 speed basic-1.0 basic-2.0 basic-5.5 6.0 9.0 basic-11.0 12.0 18.0 24.0 36.0 48 54.0
 rts threshold 4000
 station-role root
 infrastructure-client
 bridge-group 1
!
interface Dot11Radio0.40
 encapsulation dot1Q 40
 no ip route-cache
 bridge-group 40
!
interface FastEthernet0
 no ip address
 no ip route-cache
 duplex auto
 speed auto
 bridge-group 1
!
interface FastEthernet0.40
 encapsulation dot1Q 40
 no ip route-cache
 bridge-group 40
!
ip http server
ip http help-path http://www.cisco.com/warp/public/779/smbiz/prodconfig/help/eag
/122-15.JA/1100
ip radius source-interface BVI1
radius-server attribute 32 include-in-access-req format%h
radius-server host 10.91.104.92 auth-port 1645 acct-port 1646 key 7 135445415F59
radius-server authorization permit missing Service-Type
radius-server vsa send accounting
bridge 1 route ip
!
line con 0
line vty 5 15
!
end
```

RADIUS の設定およびイネーブル化

ここでは、Remote Authentication Dial-In User Service（RADIUS）を設定してイネーブルにする方法について説明します。

RADIUS の概要

RADIUS は、不正アクセスからネットワークを保護する分散型クライアント/サーバ システムです。RADIUS クライアントは、シスコのサポート対象デバイス上で稼動し、中央の RADIUS サーバに認証要求を送ります。中央の RADIUS サーバには、全ユーザの認証情報とネットワーク サービスへのアクセス情報が保管されています。RADIUS ホストは通常、シスコ、Livingston、Merit、Microsoft、または他のソフトウェア プロバイダーの RADIUS サーバ ソフトウェアが稼動しているマルチユーザ 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、次のようなネットワーク環境で使用します。

- マルチベンダー アクセス サーバで構成され、各サーバが RADIUS をサポートしているネットワーク。たとえば、複数のベンダーのアクセス サーバで、単一の 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。マルチベンダー アクセス サーバで構成される IP ベース ネットワークにおいて、ダイヤルイン ユーザは、Kerberos セキュリティ システムと連動するようにカスタマイズされた RADIUS サーバを使用して認証されます。
- アプリケーションが RADIUS プロトコルをサポートする、ターンキー方式のネットワーク セキュリティ環境。*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など。たとえば、Enigma のセキュリティ カードと RADIUS を組み合わせて使用し、ユーザを検証してネットワーク リソースへのアクセス権を与えている例があります。
- すでに RADIUS を使用しているネットワーク。RADIUS クライアントが組み込まれたシスコのブリッジをネットワークに追加できます。
- リソース アカウンティングが必要なネットワーク。RADIUS の認証または許可機能とは別個に RADIUS アカウンティング機能を使用できます。RADIUS アカウンティング機能を使用すると、サービスの開始時と終了時にデータを送信し、セッションで使用されたリソース量（時間、パケット、バイトなど）を示すことができます。インターネット サービス プロバイダーは、セキュリティおよび請求に関する特殊なニーズを満たすために、フリーウェア バージョンの RADIUS アクセス制御およびアカウンティング ソフトウェアを使用する場合もあります。

RADIUS は、次のようなネットワーク条件には適しません。

- 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 Packet Assembler Disassembler（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 は双方向では認証を行いません。非シスコ装置で認証が必要な場合、RADIUS を使用すると、ある装置から非シスコ装置に対して認証を実行できます。
- さまざまなサービスを使用するネットワーク。RADIUS は一般に、1 つのサービス モデルにユーザを束縛します。

RADIUS の動作

RADIUS サーバによってアクセスが制御されるブリッジに対して、非ルート ブリッジが認証を試みた場合、ネットワークに対する認証は図 4-1 の手順で行われます。

図 4-1　EAP 認証のシーケンス

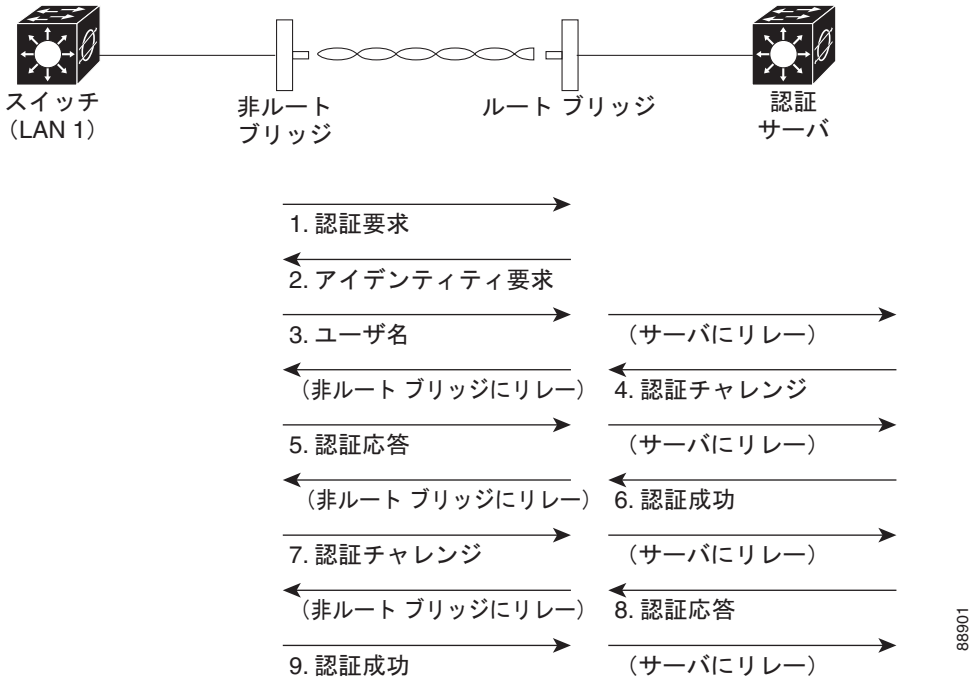


図 4-1 では、有線 LAN の非ルート ブリッジおよび RADIUS サーバは 802.1x および EAP を使用して、ルート デバイスによる相互認証を実行します。RADIUS サーバは非ルート ブリッジに認証チャレンジを送信します。非ルート ブリッジはユーザが指定したパスワードを一方向に暗号化して、チャレンジへの応答を生成し、RADIUS サーバに送信します。RADIUS サーバは、ユーザ データベースの情報を使用して独自の応答を作成し、非ルート ブリッジからの応答と比較します。RADIUS サーバが非ルート ブリッジを認証すると、プロセスが逆方向で繰り返され、非ルート ブリッジは RADIUS サーバを認証します。

相互認証が完了すると、RADIUS サーバおよび非ルート ブリッジは非ルート ブリッジに対して一意の WEP キーを判別し、非ルート ブリッジに適切なネットワーク アクセス レベルを設定します。これにより、配線されたスイッチド セグメントのセキュリティ レベルが、デスクトップとほぼ同じレベルになります。非ルート ブリッジはこの鍵をロードし、ログオン セッションに使用する準備を行います。

ログオン セッション中に、RADIUS サーバは *セッション キー*と呼ばれる WEP キーを暗号化し、有線 LAN を介してルート デバイスに送信します。ルート デバイスはセッション キーを使用してブロードキャスト キーを暗号化し、暗号化されたブロードキャスト キーを非ルート ブリッジに送信します。非ルート ブリッジはセッション キーを使用して、ブロードキャスト キーの暗号を解除します。非ルート ブリッジおよびルート デバイスは WEP をアクティブにし、セッションの残りの期間中のすべての通信に対して、セッションおよびブロードキャスト WEP キーを使用します。

EAP 認証には複数のタイプがありますが、ルート デバイスの動作はどのタイプでも同様です。ルート デバイスは非ルート ブリッジから送られた認証メッセージを RADIUS サーバにリレーし、RADIUS サーバから送信されたものを非ルート ブリッジにリレーします。RADIUS サーバを使用する認証の設定手順については、「認証タイプ」を参照してください。

RADIUS による WMIC アクセスの制御

ここでは、RADIUS を使用して WMIC への管理者アクセスを制御する方法を示します。

RADIUS を使用すると、アカウンティングの詳細を取得したり、認証および許可プロセスの柔軟な管理制御を実現できます。RADIUS は Authentication, Authorization, and Accounting（AAA; 認証、許可、アカウンティング）によって機能が拡張されており、RADIUS をイネーブルにするには AAA コマンドを使用する必要があります。RADIUS および AAA は、デフォルトでディセーブルです。

最低限、RADIUS サーバ ソフトウェアを実行するホスト（1 つまたは複数）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。

方式リストでは、非ルート ブリッジの認証、許可、またはアカウント維持に使用する方式とその順番を定義します。使用するセキュリティ プロトコルを 1 つまたは複数指定するときに方式リストが使用されるので、最初の方式が失敗してもバックアップ システムが確保されます。ソフトウェアは、最初に指定された方式を使用して非ルート ブリッジの認証、許可、またはアカウント維持を行います。その方式が失敗した場合、ソフトウェアはリストの 2 番めの方式を選択します。リスト内の方式と対話が成立するまで、または方式リストが終わるまで、このプロセスが繰り返されます。

RADIUS 機能を設定する前に、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、RADIUS の設定について説明します。

- RADIUS サーバ ホストの指定
- RADIUS ログイン認証の設定
- AAA サーバ グループの定義
- ユーザ イネーブル アクセスおよびネットワーク サービスに対する RADIUS 許可の設定
- RADIUS アカウンティングの開始
- すべての RADIUS サーバで共通の値の設定
- ベンダー固有の RADIUS 属性を使用するためのブリッジの設定
- ベンダー独自仕様の RADIUS サーバ通信に対応できるようにするためのブリッジの設定
- RADIUS 設定の表示

（注）　ここで使用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 Release 12.2*』を参照してください。

RADIUS サーバ ホストの指定

アクセス ポイントと RADIUS サーバ間の通信には、複数の要素が必要です。

- 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
- 鍵文字列
- タイムアウト期間
- 再送信の値

RADIUS セキュリティ サーバは、ホスト名または IP アドレス、ホスト名と特定の User Datagram Protocol（UDP; ユーザ データグラム プロトコル）ポート番号、または IP アドレスと特定の UDP ポート番号によって指定します。IP アドレスと UDP ポート番号を組み合わせて一意の識別子を作成することにより、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。一意の識別子により、同じ IP アドレスのサーバ上にある複数の UDP ポートに RADIUS 要求を送信できます。

同じサービス（たとえば、アカウンティング）に対して、同じ RADIUS サーバ上に 2 種類のホスト エントリを設定した場合、2 番めに設定されたホスト エントリが最初のエントリに対するフェールオーバー バックアップの役割を果たします。この場合、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かった場合、ブリッジは同じ装置上でアカウンティング サービス用に設定されている 2 番めのホスト エントリを試行します （RADIUS ホスト エントリは、設定された順に試行されます）。

RADIUS サーバおよびブリッジは、共有する秘密鍵の文字列を使用してパスワードを暗号化し、応答をやり取りします。AAA セキュリティ コマンドを使用できるように RADIUS を設定するには、RADIUS サーバ デーモンが動作するホストとともに、ブリッジと共有する秘密鍵の文字列を指定する必要があります。

タイムアウト、再送信、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設定することも、サーバ単位で設定することも、または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。ブリッジと通信するすべての RADIUS サーバに対してグローバルに設定値を適用する場合は、**radius-server timeout**、**radius-server retransmit**、**radius-server key** という 3 種類の一意のグローバル コンフィギュレーション コマンドを使用します。特定の RADIUS サーバにこれらの値を適用する場合は、**radius-server host** コマンドをグローバル コンフィギュレーション モードで使用します。

（注）　ブリッジ上でグローバル機能とサーバ単位の機能を両方とも設定した場合（timeout、retransmission、および key コマンド）、サーバ単位の timer、retransmission、および key value コマンドによってグローバルな timer、retransmission、および key value コマンドが上書きされます。すべての RADIUS サーバ上でこれらの値を設定する手順については、「すべての RADIUS サーバで共通の値の設定」（P.4-29）を参照してください。

AAA サーバ グループを使用して認証用に既存のサーバ ホストをグループ分けするように、ブリッジを設定できます。詳細については、「AAA サーバ グループの定義」（P.4-25）を参照してください。

サーバ単位の 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。

	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。
ステップ 3	**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、認証要求に対応する 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に対応する UDP 宛先ポートを指定します。 • （任意）**timeout** *seconds* には、ブリッジが再送信を行うまでに RADIUS サーバの応答を待機するタイム インターバルを指定します。範囲は 1 ～ 1000 です。この設定によって、**radius-server timeout** コマンドを使用して設定されたインターバルは上書きされます。**radius-server host** コマンドでタイムアウトを設定していない場合は、**radius-server timeout** コマンドで設定されたタイム インターバルが使用されます。 • （任意）**retransmit** *retries* には、サーバが応答しなかった場合、または応答が遅れた場合に、RADIUS 要求をサーバに送信する回数を指定します。範囲は、1 ～ 1000 です。**radius-server host** コマンドで再送信値を設定していない場合は、**radius-server retransmit** コマンドで設定された値が使用されます。 • （任意）**key** *string* には、ブリッジと RADIUS サーバ上で動作している RADIUS デーモン間で使用される、認証および暗号鍵を指定します。 **（注）** key は、RADIUS サーバで使用される暗号キーと一致する必要のあるテキスト文字列です。鍵は必ず、**radius-server host** コマンドの最終項目として設定します。先行スペースは無視されますが、鍵の中および末尾のスペースは使用されます。鍵の中にスペースが含まれる場合、引用符が鍵の一部である場合を除き、鍵を引用符で囲まないでください。 1 つの IP アドレスと対応付けられた複数のホスト エントリを認識するようにブリッジを設定する場合は、必要な回数だけ次のコマンドを入力し、各 UDP ポートが重複しないようにします。ブリッジ ソフトウェアは指定された順に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、および暗号鍵の値を設定します。
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。

指定した RADIUS サーバを削除するには、**no radius-server host** *hostname* | *ip-address* コマンドをグローバル コンフィギュレーション モードで使用します。

認証に使用する RADIUS サーバとアカウンティングに使用する RADIUS サーバを 1 つずつ設定する例を示します。

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

RADIUS サーバとして *host1* を設定し、認証とアカウンティングの両方にデフォルト ポートを使用する例を示します。

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

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、そのリストを各種インターフェイスに適用します。認証方式リストでは、実行すべき認証のタイプと、実行する順序を定義します。このリストを個々のインターフェイスに適用しないと、定義した認証方式はいずれも実行されません。唯一の例外は、デフォルトの方式リストです（このリストは *default* という名前です）。デフォルトの方式リストは、名前付きの方式リストを明示的に定義したインターフェイスを除くすべてのインターフェイスに、自動的に適用されます。

方式リストは、ユーザを認証するために使用する認証方式およびその順序を表すリストです。認証に使用するセキュリティ プロトコルを 1 つまたは複数指定することによって、最初の方式が失敗したときの認証のバックアップ システムを確保します。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。その方式に失敗した場合、ソフトウェアは方式リスト内の次の認証方式を選択します。リスト内の認証方式とコミュニケーションが成立するまで、または定義されているすべての方式を使い果たすまで、このプロセスが繰り返されます。このサイクルのいずれかの時点で認証に失敗した場合（セキュリティ サーバまたはローカル ユーザ名データベースの応答でユーザ アクセスが拒否された場合）、認証プロセスは中止され、その他の認証方式が試みら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。
ステップ 3	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]	ログイン認証の方式リストを作成します。 • **login authentication** コマンドで名前付きリストが指定されていない場合に使用する、デフォルトのリストを作成するには、**default** キーワードに続いて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、すべてのインターフェイスに自動的に適用されます。リスト名の詳細については、次のリンクを参照してください。 http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsaaa/scfathen.htm#xtocid2 • *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。定義済みの他の認証方式が使用されるのは、前の方式が失敗した場合ではなく、エラーが戻った場合に限られます。 次のいずれかの方式を選択します。 • **line**：ライン パスワードを使用して認証します。この認証方式を使用するには、先にライン パスワードを定義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使用して認証します。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* コマンドをグローバル コンフィギュレーション モードで使用します。 • **radius**：RADIUS 認証を使用します。この認証方式を使用するには、事前に RADIUS サーバを設定しておく必要があります。詳細については、「RADIUS サーバ ホストの指定」を参照してください。

	コマンド	目的
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、認証リストを適用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、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	**radius-server attribute 32 include-in-access-req format %h**	NAS_ID 属性でシステム名を送信して認証を行うように、デバイスを設定します。
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** コマンドをグローバル コンフィギュレーション モードで使用します。AAA 認証をディセーブルにするには、**no aaa authentication login** {**default** | *list-name*} *method1* [*method2...*] コマンドをグローバル コンフィギュレーション モードで使用します。ログインに関する RADIUS 認証をディセーブルにするか、またはデフォルト値に戻すには、**no login authentication** {**default** | *list-name*} コマンドをライン コンフィギュレーション モードで使用します。

AAA サーバ グループの定義

AAA サーバ グループを使用するように WMIC を設定すると、既存のサーバ ホストをグループ化して認証できます。設定済みのサーバ ホストの一部を選択して、それらを特定のサービスに使用します。サーバ グループを使用する場合は、グローバルなサーバ ホスト リスト（選択したサーバ ホストの IP アドレスのリスト）も使用します。

サーバ グループには、同じサーバのホスト エントリを複数含めることもできます。そのためには各エントリの ID（IP アドレスと UDP ポート番号の組み合わせ）を一意に設定し、各ポートを特定の AAA サービスを提供する RADIUS ホストとして個別に定義できるようにする必要があります。同じ RADIUS サーバ上の異なる 2 つのホスト エントリに同じサービス（アカウンティングなど）を設定した場合、2 番めに設定したホスト エントリは、最初に設定したホスト エントリのフェールオーバー バックアップとして機能します。

定義したグループ 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。サーバを IP アドレスで識別したり、複数のホスト インスタンスまたはエントリを識別したりするには、オプションの **auth-port** および **acct-port** キーワードを使用します。

AAA サーバ グループを定義し、そのグループに特定の RADIUS サーバを対応付けるには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	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。
ステップ 3	**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、認証要求に対応する 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に対応する UDP 宛先ポートを指定します。 • （任意）**timeout** *seconds* には、ブリッジが再送信を行うまでに RADIUS サーバの応答を待機するタイム インターバルを指定します。範囲は 1 ～ 1000 です。この設定によって、**radius-server timeout** コマンドを使用して設定されたインターバルは上書きされます。**radius-server host** コマンドでタイムアウトを設定していない場合は、**radius-server timeout** コマンドで設定されたタイム インターバルが使用されます。 • （任意）**retransmit** *retries* には、サーバが応答しなかった場合、または応答が遅れた場合に、RADIUS 要求をサーバに送信する回数を指定します。範囲は、1 ～ 1000 です。**radius-server host** コマンドで再送信値を設定していない場合は、**radius-server retransmit** コマンドで設定された値が使用されます。 • （任意）**key** *string* には、ブリッジと RADIUS サーバ上で動作している RADIUS デーモン間で使用される、認証および暗号鍵を指定します。 **（注）** key は、RADIUS サーバで使用される暗号キーと一致する必要のあるテキスト文字列です。鍵は必ず、**radius-server host** コマンドの最終項目として設定します。先行スペースは無視されますが、鍵の中および末尾のスペースは使用されます。鍵の中にスペースが含まれる場合、引用符が鍵の一部である場合を除き、鍵を引用符で囲まないでください。 1 つの IP アドレスと対応付けられた複数のホスト エントリを認識するようにブリッジを設定する場合は、必要な回数だけ次のコマンドを入力し、各 UDP ポートが重複しないようにします。ブリッジ ソフトウェアは指定された順に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、および暗号鍵の値を設定します。
ステップ 4	**aaa group server radius** *group-name*	グループ名を使用して AAA サーバグループを定義します。 このコマンドを使用して、ブリッジをサーバ グループ コンフィギュレーション モードにします。
ステップ 5	**server** *ip-address*	特定の RADIUS サーバを定義済みのサーバ グループに関連付けます。AAA サーバ グループに含まれる RADIUS サーバごとにこの手順を繰り返します。 **（注）** グループの各サーバは、ステップ 2 で事前に定義しておく必要があります。
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。
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.4-24）を参照してください。

指定した RADIUS サーバを削除するには、**no radius-server host** *hostname* | *ip-address* コマンドをグローバル コンフィギュレーション モードで使用します。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* コマンドをグローバル コンフィギュレーション モードで使用します。RADIUS サーバの IP アドレスを削除するには、**no server** *ip-address* コマンドをサーバ グループ コンフィギュレーション モードで使用します。

次の例では、2 種類の RADIUS サーバ グループ（*group1* および *group2*）を認識するように、ブリッジを設定しています。group1 では、同じ RADIUS サーバ上の異なる 2 つのホスト エントリを、同じサービス用に設定しています。2 番めのホスト エントリは、最初のエントリのフェールオーバー バックアップとして機能します。

```
bridge(config)# aaa new-model
bridge(config)# radius-server host 172.20.0.1 auth-port 1000 acct-port 1001
bridge(config)# radius-server host 172.10.0.1 auth-port 1645 acct-port 1646
bridge(config)# aaa group server radius group1
bridge(config-sg-radius)# server 172.20.0.1 auth-port 1000 acct-port 1001
bridge(config-sg-radius)# exit
bridge(config)# aaa group server radius group2
bridge(config-sg-radius)# server 172.20.0.1 auth-port 2000 acct-port 2001
bridge(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 サービスに対する RADIUS 許可の設定

AAA 許可は、ユーザが使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場合、WMIC はユーザのプロファイルから取得した情報を使用して、ユーザのセッションを設定します。ユーザのプロファイルは、ローカル ユーザ データベースまたはセキュリティ サーバ上にあります。ユーザ プロファイルの情報によって許可されている場合に限り、ユーザは要求したサービスにアクセスできます。

イネーブル EXEC モードで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するには、グローバル コンフィギュレーション モードの **aaa authorization** コマンドを **radius** キーワードを指定して使用します。

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、RADIUS を使用してイネーブル EXEC アクセスを許可します。
- 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る場合でも、CLI を使用してログインした認証済みユーザには、許可が省略されます。

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	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 authorization network radius**	ネットワーク関連のあらゆるサービス要求に関して、ユーザの RADIUS 許可を実行するようにブリッジを設定します。
ステップ 3	**aaa authorization exec radius**	ユーザのイネーブル EXEC アクセス権を判別するために、ユーザの RADIUS 許可を実行するようにブリッジを設定します。 **exec** キーワードを指定すると、ユーザ プロファイル情報（**autocommand** 情報など）が戻ることがあります。
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* コマンドをグローバル コンフィギュレーション モードで使用します。

RADIUS アカウンティングの開始

AAA アカウンティング機能は、ユーザがアクセスしているサービスおよびユーザが消費しているネットワーク リソース量を追跡します。AAA アカウンティングがイネーブルの場合、ブリッジはアカウンティング レコードの形で、RADIUS サーバにユーザ アクティビティを報告します。各アカウンティング レコードは、アカウンティング属性と値（AV）のペアが含まれ、セキュリティ サーバ上で保管されます。このデータを分析し、ネットワーク管理、クライアントに対する課金、または監査目的で使用できます。

各 Cisco IOS イネーブル レベルおよびネットワーク サービスに対して 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	**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あらゆるサービス要求に関して、RADIUS アカウンティングをイネーブルにします。
ステップ 3	**ip radius source-interface bvi1**	アカウンティング レコードの NAS_IP_ADDRESS 属性で Bridge-Group Virtual Interface（BVI; ブリッジ グループ仮想インターフェイス）IP アドレスを送信するようにブリッジを設定します。
ステップ 4	**aaa accounting update periodic** *minutes*	アカウンティングの更新間隔（分）を入力します。
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。

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...* コマンドをグローバル コンフィギュレーション モードで使用します。

すべての RADIUS サーバで共通の値の設定

ブリッジとすべての RADIUS サーバ間のグローバルな通信の値を設定するには、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	**radius-server key** *string*	ブリッジとすべての RADIUS サーバ間で使用する、共有秘密鍵の文字列を指定します。 **（注）** key は、RADIUS サーバで使用される暗号キーと一致する必要のあるテキスト文字列です。先行スペースは無視されますが、鍵の中および末尾のスペースは使用されます。鍵の中にスペースが含まれる場合、引用符が鍵の一部である場合を除き、鍵を引用符で囲まないでください。
ステップ 3	**radius-server retransmit** *retries*	ブリッジが送信を中止するまでに、各 RADIUS 要求をサーバに送信する回数を指定します。デフォルトは 3、範囲は 1 ～ 1000 です。
ステップ 4	**radius-server timeout** *seconds*	ブリッジが RADIUS 要求を再送信するまでに、要求に対する応答を待機する秒数を指定します。デフォルトは 5、範囲は 1 ～ 1000 です。
ステップ 5	**radius-server deadtime** *minutes*	このコマンドを使用すると、認証要求に応答しなかった RADIUS サーバは、Cisco IOS ソフトウェアがダウンしていると判断するので、要求のタイムアウトを待たなくても、次のコンフィギュレーション サーバを試行できます。ダウンしていると判断された RADIUS サーバは、指定した分数だけその後の要求では除外されます。 **（注）** 複数の RADIUS サーバを用意する場合は、最適なパフォーマンスが得られるように RADIUS サーバのデッドタイムを設定する必要があります。
ステップ 6	**radius-server attribute 32 include-in-access-req format %h**	NAS_ID 属性でシステム名を送信して認証を行うように、デバイスを設定します。
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。

retransmit、timeout、および deadtime をデフォルト値に戻すには、各コマンドの **no** 形式を使用します。

ベンダー固有の RADIUS 属性を使用するためのブリッジの設定

Internet Engineering Task Force（IETF）のドラフト規格では、ベンダー固有の属性（attribute 26）を使用することによってブリッジと RADIUS サーバ間でベンダー固有情報を受け渡す方式が指定されています。ベンダー固有の属性（VSA）によって、ベンダー各社は汎用には向かない独自の拡張属性をサポートできます。シスコの RADIUS は、仕様の推奨フォーマットを使用することによって、ベンダー固有オプションを 1 つサポートします。シスコのベンダー ID は 9 です。サポートするオプションはベンダー タイプ 1 で、名前は *cisco-avpair* です。値は次のフォーマットの文字列です。

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

protocol は、特定の認証タイプに対応するシスコ プロトコル属性の値です。*attribute* および *value* は、シスコの TACACS+ 仕様で定義されている、適切な AV ペアです。*sep* は必須属性を表す = およびオプションの属性を表すアスタリスク（*）です。これにより、TACACS+ 認証で利用できるフルセットの機能を RADIUS にも使用できます。

たとえば、次の AV ペアの場合、IP 認証時（Point-to-Point Protocol IP Control Protocol（PPP IPCP; ポイントツーポイント プロトコル/IP 制御プロトコル）によるアドレス割り当て時）に、シスコの *Multiple Named IP Address Pools*（複数の名前付き IP アドレス プール）機能がアクティブになります。

```
cisco-avpair= ”ip:addr-pool=first“
```

次に、ブリッジからイネーブル EXEC コマンドに即時アクセスできる状態でユーザをログインさせる例を示します。

```
cisco-avpair= ”shell:priv-lvl=15“
```

ベンダーごとに、それぞれ一意のベンダー ID、オプション、および対応 VSA があります。ベンダー ID および 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ブリッジを設定するには、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	**radius-server vsa send** [**accounting** | **authentication**]	RADIUS IETF attribute 26 で定義された VSA をブリッジが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 VSA セットをアカウンティング属性に限定します。 • （任意）**authentication** キーワードを使用して、認識される VSA セットを認証属性に限定します。 キーワードを指定しないでこのコマンドを入力した場合は、アカウンティングと認証の VSA が両方とも使用されます。
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 属性の全リストまたは VSA 26 の詳細については、『*Cisco IOS Security Configuration Guide for Release 12.2*』の付録「RADIUS Attributes」を参照してください。

ベンダー独自仕様の RADIUS サーバ通信に対応できるようにするためのブリッジの設定

RADIUS に関する IETF のドラフト規格で、ブリッジと RADIUS サーバ間でベンダー独自仕様の情報を受け渡す方式が指定されていますが、一部のベンダーは、それぞれ独自の方法で RADIUS 属性セットを拡張しています。Cisco IOS ソフトウェアは、ベンダー独自仕様 RADIUS 属性のサブセットをサポートします。

前述のとおり、（ベンダー独自仕様であるか、IETF のドラフトに準拠しているかにかかわらず）RADIUS を設定するには、RADIUS サーバ デーモンを実行するホストとともに、ブリッジと共有する秘密鍵の文字列を指定する必要があります。RADIUS ホストと秘密鍵の文字列を指定するには、**radius-server** コマンドをグローバル コンフィギュレーション モードで使用します。

ベンダー独自仕様の RADIUS サーバ ホストおよび共有する秘密鍵文字列を指定するには、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	**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、そのホストでベンダー独自仕様の RADIUS を使用することを指定します。
ステップ 3	**radius-server key** *string*	ブリッジとベンダー独自仕様の RADIUS サーバ間で使用する、共有秘密鍵の文字列を指定します。ブリッジと RADIUS サーバはこの文字列を使用し、パスワードを暗号化して応答を交換します。 **（注）** key は、RADIUS サーバで使用される暗号キーと一致する必要のあるテキスト文字列です。先行スペースは無視されますが、鍵の中および末尾のスペースは使用されます。鍵の中にスペースが含まれる場合、引用符が鍵の一部である場合を除き、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。

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

ベンダー独自仕様の RADIUS ホストを指定し、ブリッジとサーバ間で *rad124* という秘密鍵を指定する例を示します。

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

RADIUS 設定の表示

RADIUS 設定を表示するには、**show running-config** コマンドをイネーブル EXEC モードで使用します。

TACACS+ による WMIC アクセスの制御

ここでは、Terminal Access Controller Access Control System Plus（TACACS+）を使用して WMIC への管理者アクセスを制御する方法を示します。

TACACS+ を使用すると、アカウンティングの詳細を取得したり、認証および許可プロセスの柔軟な管理制御を実現できます。TACACS+ は AAA によって機能が拡張されており、TACACS+ をイネーブルにするには AAA コマンドを使用する必要があります。

（注）　ここで使用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 Release 12.2*』を参照してください。

TACACS+ の概要

TACACS+ は、ブリッジにアクセスしようとするユーザを一元的に検証するセキュリティ アプリケーションです。TACACS+ は RADIUS と異なり、ルート デバイスに対応付けられた非ルート ブリッジの認証は行いません。

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動作している TACACS+ デーモンのデータベースで維持されます。WMIC 上で TACACS+ 機能を設定する前に、TACACS+ サーバにアクセスして設定する必要があります。

TACACS+ は、独立したモジュラ型の認証、許可、およびアカウンティング機能を提供します。TACACS+ を使用すると、1 つのアクセス制御サーバ（TACACS+ デーモン）で認証、許可、またはアカウンティングの各サービスを別々に提供できます。各サービスを専用のデータベースに結合することによって、デーモンの能力に応じて、そのサーバ上またはネットワーク上で利用できる他のサービスを活用できます。

TACACS+ は AAA セキュリティ サービスを介して管理され、次のサービスを提供できます。

- 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、さらにメッセージング サポートによって、管理者の認証を全面的に制御します。

  認証機能は、管理者に対する対話を続けます（たとえば、ユーザ名とパスワードの入力後、自宅の住所、母親の旧姓、サービス タイプ、社会保険番号といったいくつかの質問をユーザに出します）。TACACS+ 認証サービスは、管理者の画面にメッセージを送信することもできます。たとえば、パスワードの有効期限に関する会社の方針を理由に、パスワードの変更が必要であることをメッセージで管理者に通知できます。

- 許可：管理者のセッションが終了するまで、自動コマンド、アクセス制御、セッションの長さ、プロトコル サポートをはじめ、管理者の能力をきめ細かく制御します。管理者が TACACS+ 許可機能で実行できるコマンドに制限を加えることもできます。

- アカウンティング：請求、監査、TACACS+ デーモンへのレポートに使用する情報を収集して送信します。ネットワーク管理者はアカウンティング機能を使用することによって、セキュリティ監査目的で管理者の行動を追跡したり、ユーザに請求するための情報を提供したりできます。アカウンティング レコードには管理者のアイデンティティ、開始時刻、終了時刻、実行したコマンド（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 WMIC と TACACS+ デーモン間で認証を行います。WMIC と TACACS+ デーモン間のすべてのプロトコル交換が暗号化されるので、機密性が保証されます。

WMIC 上で TACACS+ を使用するには、TACACS+ デーモン ソフトウェアが動作しているシステムが必要です。

TACACS+ の動作

管理者が TACACS+ を使用し、WMIC に対して認証することによって単純な ASCII ログインを試行した場合、次のプロセスが発生します。

1. 接続の確立後、WMIC は TACACS+ デーモンにアクセスしてユーザ名プロンプトを取得し、そのプロンプトが管理者に表示されます。管理者がユーザ名を入力します。その後、WMIC が TACACS+ デーモンにアクセスしてパスワード プロンプトを取得します。WMIC が管理者にパスワード プロンプトを表示します。管理者はパスワードを入力し、そのパスワードが TACACS+ デーモンに送信されます。

   TACACS+ は、デーモンが管理者を認証できるだけの情報を得るまで、デーモンと管理者間で対話を続けさせます。デーモンはユーザ名とパスワードのコンビネーションを入力するように要求しますが、ユーザの母親の旧姓など、他の項目を含めることもできます。

2. WMIC は最終的に、TACACS+ デーモンから次の応答のいずれか 1 つを得ます。
   - ACCEPT：管理者は認証されました。サービスを開始できます。WMIC が許可を要求するように設定されている場合は、この時点で許可のプロセスが開始されます。
   - REJECT：管理者は認証されていません。管理者は TACACS+ デーモンに応じて、アクセスが拒否されることもあれば、ログイン シーケンスのやり直しが求められることもあります。
   - ERROR：認証中にデーモンで、またはデーモンと WMIC 間のネットワーク接続でエラーが発生しました。ERROR 応答を受信した WMIC は通常、別の方式で管理者の認証を試みます。
   - CONTINUE：管理者は、他の認証情報を入力するように求められます。

   WMIC 上で許可機能がイネーブルになっている場合は、認証後、管理者は許可を試みます。管理者は TACACS+ 認証を正常に完了しないかぎり、TACACS+ 許可には進めません。

3. TACACS+ 許可が必要な場合、TACACS+ デーモンに再びアクセスし、デーモンが ACCEPT または REJECT の許可応答を戻します。ACCEPT の応答が戻った場合、応答には属性の形で、その管理者の EXEC または NETWORK セッションを管理し、管理者がアクセスできるサービスを決定付けるデータが含まれています。データは次のとおりです。
   - Telnet、rlogin、またはイネーブル EXEC サービス
   - 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、管理者のタイムアウトなどの接続パラメータ

TACACS+ のデフォルト設定

TACACS+ および AAA は、デフォルトではディセーブルです。

セキュリティの失効を防止するため、ネットワーク管理アプリケーションを使用して TACACS+ を設定できません。TACACS+ をイネーブルに設定した場合は、CLI を通じて WMIC にアクセスする管理者を認証できます。

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、そのリストを各種インターフェイスに適用します。認証方式リストでは、実行すべき認証のタイプと、実行する順序を定義します。このリストを個々のインターフェイスに適用しないと、定義した認証方式はいずれも実行されません。唯一の例外は、デフォルトの方式リストです（このリストは *default* という名前です）。

デフォルトの方式リストは、名前付きの方式リストを明示的に定義したインターフェイスを除くすべてのインターフェイスに、自動的に適用されます。定義された方式リストは、デフォルトの方式リストよりも優先します。

方式リストは、ユーザを認証するために使用する認証方式およびその順序を表すリストです。認証に使用するセキュリティ プロトコルを 1 つまたは複数指定することによって、最初の方式が失敗したときの認証のバックアップ システムを確保できます。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。その方式に失敗した場合、ソフトウェアは方式リスト内の次の認証方式を選択します。リスト内の認証方式とコミュニケーションが成立するまで、または定義されているすべての方式を使い果たすまで、このプロセスが繰り返されます。認証に失敗した場合（セキュリティ サーバまたはローカル ユーザ名データベースの応答でユーザ アクセスが拒否された場合）、認証プロセスは中止され、その他の認証方式が試みられることはありません。

TACACS+ サーバ ホストの指定および認証鍵の設定

単一サーバまたは AAA サーバ グループを使用して認証用に既存のサーバ ホストをグループ分けするように、WMIC を設定できます。サーバをグループに分けると、設定されているサーバ 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 グループはグローバル サーバ ホスト リストと組み合わせて使用し、選択されたサーバ ホストの IP アドレスのリストを指定します。

TACACS+ サーバを維持している IP ホスト（1 つまたは複数）を指定し、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	**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+ サーバを維持している IP ホスト（1 つまたは複数）を指定します。このコマンドを繰り返し入力して、優先ホストのリストを作成します。ソフトウェアは指定された順に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。範囲は 1 ～ 65535 です。 • （任意）**timeout** *integer* には、タイムアウトしてエラーを宣言するまでに、WMIC がデーモンの応答を待機する時間を秒数で指定します。デフォルト値は 5 です。範囲は、1 ～ 1000 です。 • （任意）**key** *string* には、WMIC と TACACS+ デーモン間の全トラフィックを暗号化/復号化するための暗号鍵を指定します。暗号化を正しく行うには、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）グループ名を使用して AAA サーバ グループを定義します。 このコマンドによって、WMIC をサーバ グループ サブコンフィギュレーション モードにします。

	コマンド	目的
ステップ 5	**server** *ip-address*	（任意）特定の TACACS+ サーバを定義したサーバ グループに対応付けます。AAA サーバ グループに含まれる TACACS+ サーバごとにこの手順を繰り返します。 グループの各サーバは、ステップ 2 で事前に定義しておく必要があります。
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 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。

指定した TACACS+ サーバを削除するには、**no tacacs-server host** *hostname* コマンドをグローバル コンフィギュレーション モードで使用します。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 tacacs+** *group-name* グローバル コンフィギュレーション モードを使用します。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 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、そのリストを各種インターフェイスに適用します。認証方式リストでは、実行すべき認証のタイプと、実行する順序を定義します。この方式を個々のインターフェイスに適用しないと、定義した認証方式はいずれも実行できません。唯一の例外は、デフォルトの方式リストです（このリストは *default* という名前です）。デフォルトの方式リストは、名前付きの方式リストを明示的に定義したインターフェイスを除くすべてのインターフェイスに、自動的に適用されます。定義された方式リストは、デフォルトの方式リストよりも優先します。

方式リストは、ユーザを認証するために使用する認証方式およびその順序を表すリストです。認証に使用するセキュリティ プロトコルを 1 つまたは複数指定することによって、最初の方式が失敗したときの認証のバックアップ システムを確保できます。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。その方式に失敗した場合、ソフトウェアは方式リスト内の次の認証方式を選択します。リスト内の認証方式とコミュニケーションが成立するまで、または定義されているすべての方式を使い果たすまで、このプロセスが繰り返されます。このサイクルのいずれかの時点で認証に失敗した場合（セキュリティ サーバまたはローカル ユーザ名データベースの応答で管理者アクセスが拒否された場合）、認証プロセスは中止され、その他の認証方式が試みら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。
ステップ 3	**aaa authentication login** {**default** | *list-name*} ***method1*** [***method2...***]	ログイン認証の方式リストを作成します。 • **login authentication** コマンドで名前付きリストが指定されていない場合に使用する、デフォルトのリストを作成するには、**default** キーワードに続いて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、すべてのインターフェイスに自動的に適用されます。 • *list-name* には、作成するリストの名前とな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。他の認証方式が使用されるのは、前の方式が失敗した場合ではなく、エラーが戻った場合に限られます。 次のいずれかの方式を選択します。 • **local**：ローカル ユーザ名データベースを使用して認証します。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。**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。 • **tacacs+**：TACACS+認証を使用します。この認証方式を使用するには、事前に TACACS+ サーバを設定しておく必要があります。
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、認証リストを適用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、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** コマンドをグローバル コンフィギュレーション モードで使用します。AAA 認証をディセーブルにするには、**no aaa authentication login** {**default** | *list-name*} *method1* [*method2...*] コマンドをグローバル コンフィギュレーション モードで使用します。ログインに関する TACACS+ 認証をディセーブルにするか、またはデフォルト値に戻すには、**no login authentication** {**default** | *list-name*} コマンドをライン コンフィギュレーション モードで使用します。

ユーザ イネーブル アクセスおよびネットワーク サービスに対する TACACS+ 許可の設定

AAA 許可は、ユーザが使用できるサービスを制限します。AAA 許可機能がイネーブルの場合、WMIC はユーザのプロファイルから取り出した情報を使用して、ユーザのセッションを設定します。ユーザのプロファイルは、ローカル ユーザ データベースまたはセキュリティ サーバ上にあります。ユーザ プロファイルの情報によって許可されている場合に限り、ユーザは要求したサービスにアクセスできます。

イネーブル EXEC モードで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するには、**aaa authorization** グローバル コンフィギュレーション コマンドに **tacacs+** キーワードを指定して使用します。

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 に対するアクセス許可に TACACS+ を使用します。
- TACACS+ を使用して認証を行わなかった場合は、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る場合でも、CLI を使用してログインした認証済みユーザには、許可が省略されます。

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	**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してユーザ TACACS+ 許可を行うように、WMIC を設定します。
ステップ 3	**aaa authorization exec tacacs+**	ユーザに対するイネーブル EXEC アクセスの可否をユーザ TACACS+ 許可によって判別するように、WMIC 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（**autocommand** 情報など）が戻ることがあります。
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* コマンドをグローバル コンフィギュレーション モードで使用します。

TACACS+ アカウンティングの開始

AAA アカウンティング機能は、管理者がアクセスしているサービスおよび消費しているネットワーク リソース量を追跡します。AAA アカウンティングがイネーブルの場合、WMIC はアカウンティング レコードの形で、TACACS+ セキュリティ サーバに管理者のアクティビティを報告します。各アカウンティング レコードは、アカウンティング属性と値（AV）のペアが含まれ、セキュリティ サーバ上で保管されます。このデータを分析し、ネットワーク管理、クライアントに対する課金、または監査目的で使用できます。

各 Cisco IOS イネーブル レベルおよびネットワーク サービスに対して 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	**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あらゆるサービス要求に関して、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+ アカウンティング機能から、イネーブル EXEC プロセスの開始時にアカウンティング レコード開始通知、終了時にレコード停止通知を送信できるようにします。
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...* コマンドをグローバル コンフィギュレーション モードで使用します。

TACACS+ 設定の表示

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

WMIC のローカル認証および許可の設定

ローカル モードで AAA を実行するように WMIC を設定すると、サーバがなくても動作するように AAA を設定できます。この場合は、WMIC が認証および許可を処理します。この設定ではアカウンティングを使用できません。

ローカル AAA 用に WMIC 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。
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** キーワードを指定すると、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	**aaa authorization exec local**	ユーザに EXEC シェルの実行が許可されているかどうかを判別するためにローカル データベースを調べるように、ユーザ AAA 許可を設定します。
ステップ 5	**aaa authorization network local**	すべてのネットワーク関連サービス要求に対して、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設定します。 ユーザごとにこのコマンドを繰り返します。 • *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、アクセス後にユーザに設定する権限レベルを指定します。範囲は 0 ～ 15 です。レベル 15 は、イネーブル EXEC モードのアクセスを設定します。レベル 0 は、ユーザ EXEC モードのアクセスを設定します。 • *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 WMIC 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。パスワードには 1 ～ 25 文字を指定します（埋め込みスペースを含めることができます）。パスワードは **username** コマンドの最後のオプションとして指定する必要があります。
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** コマンドをグローバル コンフィギュレーション モードで使用します。許可をディセーブルにするには、**no aaa authorization** {**network** | **exec**} *method1* コマンドをグローバル コンフィギュレーション モードで使用します。

WMIC の SSH の設定

ここでは、Secure Shell（SSH; セキュア シェル）機能を設定する方法について説明します。

（注）　この章で使用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 Release 12.2』の「*Secure Shell Commands*」を参照してください。

SSH の概要

SSH は、レイヤ 2 またはレイヤ 3 デバイスに安全なリモート接続を提供するプロトコルです。SSH には、バージョン 1 とバージョン 2 の 2 つのバージョンがあります。Cisco IOS リリース 12.3(8)JK では、SSH バージョン 1 のみをサポートします。

SSH は、リモート接続に関するセキュリティを Telnet よりも高めることができます。デバイスの認証時に、SSH は強力な暗号化を行います。SSH 機能には SSH サーバと SSH 統合クライアントがあります。クライアントは、次のユーザ認証方式をサポートします。

- RADIUS（詳細については、「RADIUS による WMIC アクセスの制御」（P.4-21）を参照してください）
- ローカル認証および許可（詳細については、「WMIC のローカル認証および許可の設定」（P.4-39）を参照してください）

SSH の詳細については、『Cisco IOS Security Configuration Guide for Release 12.2』の「*Configuring Secure Shell*」を参照してください。

（注）　Cisco IOS リリース 12.3(8)JK の SSH 機能は、IP Security（IPSec）をサポートしていません。

SSH の設定

SSH を設定する前に、暗号化ソフトウェア イメージを Cisco.com からダウンロードしてください。SSH の設定と SSH 設定の表示については、『Cisco IOS Security Configuration Guide for Release 12.2』の「*Configuring Secure Shell*」を参照してください。

Aironet 拡張機能の管理

WMIC は Cisco Aironet 802.11 拡張機能を使用して、シスコ クライアント デバイスの機能を検出したり、対応するクライアント デバイス間で特定の相互作用を必要とする機能をサポートしたりします。Aironet 拡張機能は、ルート アクセス ポイント モードの場合のみ、非アクティブにできます。ワークグループ ブリッジ、ルート デバイス、非ルート ブリッジはシスコ固有のモードであるため、これらのモードでは常に Aironet 拡張機能が使用されます。

次の機能をサポートするには、Aironet 拡張機能をイネーブルにする必要があります。

- 負荷分散：WMIC は Aironet 拡張機能を使用して、最適なネットワーク接続を提供するアクセス ポイントにクライアント デバイスを転送します。基準となるのは、ユーザ数、ビット エラー レート、信号強度などの係数です。
- Message Integrity Check（MIC）：MIC は、暗号化パケットに対する攻撃（*ビットフリップ攻撃*）を防ぐ追加の WEP セキュリティ機能です。WMIC と対応するクライアント デバイスの両方に MIC を実装すると、パケットの変更を防止するための数バイトのデータが各パケットに追加されます。
- Temporal Key Integrity Protocol（TKIP）：TKIP（別名 *WEP キー ハッシュ*）は追加の WEP セキュリティ機能です。この機能を使用すると、侵入者が暗号化パケット内の *Initialization Vector*（IV）という非暗号化セグメントを使用して WEP キーを計算する WEP 攻撃を防ぐことができます。
- 対応するクライアント デバイスの電力レベルの制限：クライアント デバイスが WMIC に対応付けられている場合、WMIC はこのクライアントに許可されている最大電力レベル設定を送信します。

Aironet 拡張機能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	**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	**interface dot11radio 0**	無線インターフェイスに対応するインターフェイス 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	**station-role root ap-only**	ステーションの役割を指定します。root を指定すると、アクセス ポイント モードがイネーブルになります。
ステップ 4	**no dot11 extension aironet**	拡張機能をディセーブルにする **extension aironet** コマンドを開始します。
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	**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。

無線を Aironet 拡張機能が必要な役割に変更すると、Aironet 拡張機能は自動的にイネーブルになります。

```
wmic1(config)#int dot 0
wmic1(config-if)#station-role root
Selected role requires Cisco Aironet Extension enabled.
Enabled Cisco Aironet Extension.
```

無線を適切な役割に設定しないで Aironet 拡張機能を変更しようとすると、次のエラー メッセージが表示されます。

```
wmic1(config-if)#
wmic1(config-if)#no dot11 extension aironet
Aironet Extension is always enabled in Bridge or WGB mode.
```

システム日時の管理

WMIC のシステム日時を管理するには、Network Time Protocol（NTP）を使用して自動的に行うか、または WMIC にシステム日時を設定して手動で行います。

（注）　ここで使用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 Configuration Fundamentals Command Reference for Release 12.2*』を参照してください。

システム クロックの概要

時刻サービスの中核となるのはシステム クロックです。このクロックはシステムが起動した瞬間に動作を開始し、日時を常時監視します。

システム クロックは、次の方法で設定できます。

- Network Time Protocol
- 手動設定

システム クロックは、Coordinated Universal Time（UTC; 協定世界時）（別名 GMT [グリニッジ標準時]）に基づいてシステム内部の時刻を決定します。現地のタイム ゾーンおよび夏時間に関する情報を設定することにより、現地のタイム ゾーンに応じて時刻を正確に表示できます。

システム クロックは、時刻が*信頼できる*かどうか（つまり、信頼できると見なされるタイム ソースによって時刻が設定されているか）を常時監視します。システム クロックが信頼できない場合、時刻は表示目的でのみ利用され、再配布されません。設定の詳細については、「日時の手動設定」（P.4-44）を参照してください。

NTP の概要

NTP は、ネットワーク上のデバイス間で時刻を同期するために設計されています。NTP は RFC 1305 に文書化されています。

NTP ネットワークは通常、タイム サーバに接続されたラジオ クロックまたは原子時計などの信頼できるタイム ソースから時刻を取得します。NTP が取得した時刻は、ネットワークに配信されます。NTP はきわめて効率的で、1 分間に 1 パケットを使用するだけで、2 つのデバイスを 1 ミリ秒以内に同期化できます。

NTP は、*ストラタム（階層）*という概念を使用して、信頼できるタイム ソースとデバイス間の NTP ホップ数を示します。ストラタム 1 タイム サーバには、ラジオ クロックまたは原子時計が直接接続されており、ストラタム 2 タイム サーバは、NTP を使用してストラタム 1 タイム サーバから時刻を取得します（以降のストラタムも同様です）。

NTP が稼動するデバイスは、タイム ソースとして、NTP を使用して通信するストラタム番号が最少のデバイスを自動的に選択します。この手法により、NTP スピーカの自動編成型ツリーが効率的に構築されます。

NTP では、複数のデバイスから報告される時刻を比較して、ストラタムの番号が小さくても、時刻が他のデバイスと大幅に異なるデバイスとは同期化しません。

NTP が稼動するデバイス間の通信（*アソシエーション*）は、通常スタティックに設定されます。各デバイスには、アソシエーションを形成する必要があるすべてのデバイスの IP アドレスが与えられます。アソシエーションを形成するデバイス ペア間で NTP メッセージを交換することによって、正確な時刻の維持が可能になります。ただし、LAN 環境では、代わりに IP ブロードキャスト メッセージを使用するように NTP を設定できます。この代替手段では、単にブロードキャスト メッセージを送受信するように各デバイスを設定すればよいので、設定の複雑さが緩和されます。ただし、この場合は、情報の流れは一方向に限定されます。

デバイス上で維持される時刻は、重要なリソースです。NTP のセキュリティ機能を使用して、不正確な時刻が誤って、あるいは意図的に設定されることがないようにしてください。アクセスリストベースの制約方式と、暗号化認証メカニズムの 2 つのメカニズムが使用できます。

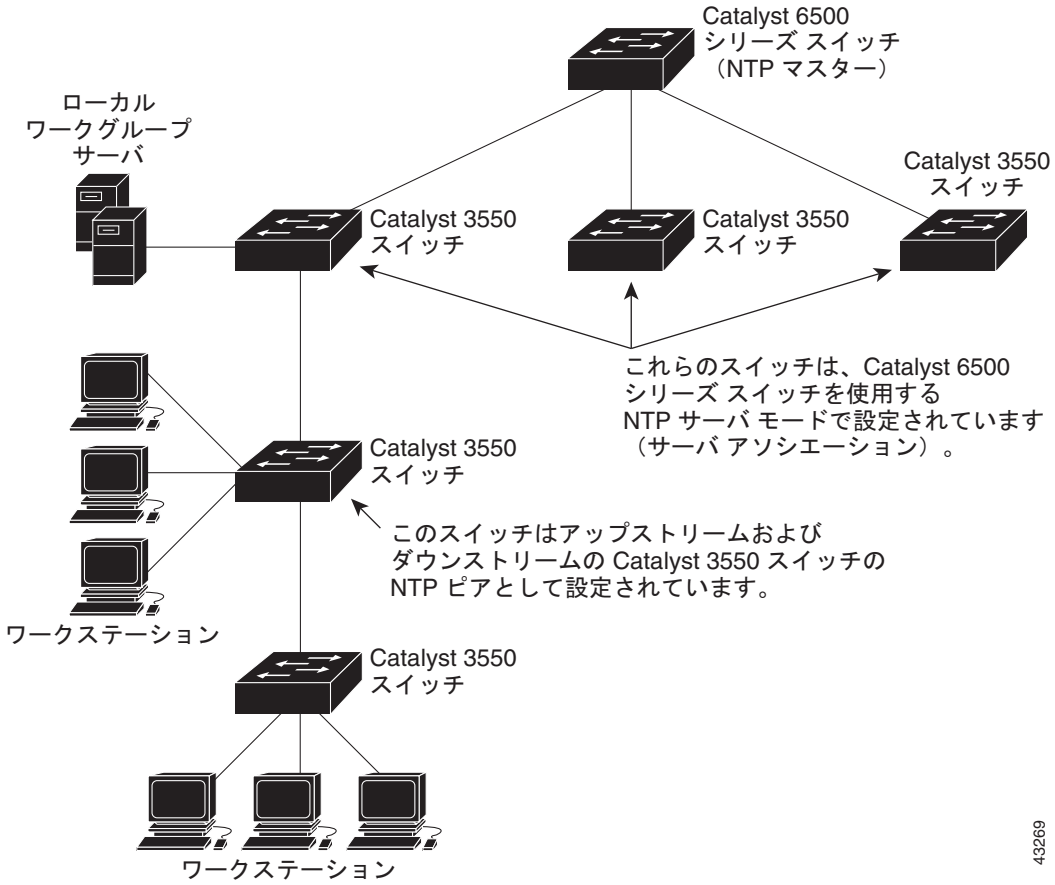
シスコの NTP はストラタム 1 サービスをサポートしていないので、ラジオ クロックまたは原子時計に接続できません。ネットワークの時刻サービスは、IP インターネットで利用可能なパブリック NTP サーバから取得することを推奨します。図 4-2 に、NTP を使用する一般的なネットワーク例を示します。

ネットワークがインターネットから切り離されている場合、シスコの NTP によって、実際には、他の方法で時刻が決定されているにもかかわらず、デバイスが NTP を使用して同期しているかのように動作させることができます。その他のデバイスは、NTP によりこのデバイスと同期化されます。

複数のタイム ソースがある場合、NTP は常により信頼できると見なされます。NTP の時刻は、他の方法による時刻よりも優先します。

一部のメーカーでは、ホスト システムに NTP ソフトウェアを組み入れています。また、UNIX システム用のバージョンやその派生ソフトウェアも一般に入手できます。NTP ソフトウェアを使用すると、ホスト システムの時刻も同期化できます。

図 4-2　　一般的な NTP ネットワーク構成



日時の手動設定

他のタイム ソースを使用できない場合は、システムの再起動後に日時を手動で設定できます。時刻は、次にシステムを再起動するまで正確です。手動設定は最後の手段としてのみ使用することを推奨します。WMIC を同期化できる外部ソースがある場合は、システム クロックを手動で設定する必要はありません。

システム クロックの設定

ネットワーク上に、時刻サービスを提供する NTP サーバなどの外部ソースがある場合は、手動でシステム クロックを設定する必要がありません。

システム クロックを設定するには、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	**clock set** *hh*:*mm*:*ss day month year* または **clock set** *hh*:*mm*:*ss month day year*	次のいずれかのフォーマットを使用して、手動でシステム クロックを設定します。 • *hh*:*mm*:*ss* には、時刻を時間（24 時間形式）、分、秒で指定します。指定された時刻は、設定されたタイム ゾーンに基づきます。 • *day* には、当月の日付で日を指定します。 • *month* には、月を名前で指定します。 • *year* には、年を指定します（短縮不可）。
ステップ 2	**show running-config**	設定を確認します。
ステップ 3	**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。

次に、システム クロックを手動で 2001 年の 7 月 23 日午後 1 時 32 分に設定する例を示します。

```
bridge# clock set 13:32:00 23 July 2001
```

日時設定の表示

日時設定を表示するには、**show clock** [**detail**] コマンドをイネーブル EXEC モードで使用します。

システム クロックには、時刻が信頼できる（正確であると確信できる）かどうかを示す *authoritative* フラグがあります。システム クロックがタイミング ソース（NTP など）によって設定されている場合は、このフラグが設定されます。時刻が信頼できない場合、このフラグは表示目的でのみ使用されます。クロックが信頼でき、*authoritative* フラグが設定された状態でないと、ピアの時刻が無効でも、ピアはクロックと同期化できません。

show clock の表示の前にある記号には、次の意味があります。

- *：時刻は信頼できません。
- （空白）：時刻は信頼できます。
- .：時刻は信頼できますが、NTP は同期していません。

タイム ゾーンの設定

手動でタイム ゾーンを設定するには、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	**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	**clock timezone** *zone hours-offset* [***minutes-offset***]	タイム ゾーンを設定します。 デバイスは内部時刻を UTC で管理しています。したがって、表示目的の場合、および手動で時刻を設定した場合以外は、このコマンドを使用しないでください。 • *zone* には、標準時間が有効な場合に表示されるタイムゾーンの名前を入力します。デフォルトは UTC です。 • *hours-offset* には、UTC からの時差（時間）を入力します。 • （任意）*minutes-offset* には、UTC からの時差（分）を入力します。
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。

clock timezone コマンドの *minutes-offset* 変数は、現地のタイム ゾーンと UTC との時差が分単位であるような領域で利用できます。たとえば、カナダ大西洋沿岸のある区域のタイム ゾーン（AST [大西洋標準時]）は UTC-3.5 です。3 は 3 時間、.5 は 50% を意味します。この場合、必要なコマンドは **clock timezone AST -3 30** です。

時刻を UTC に設定するには、**no clock timezone** コマンドをグローバル コンフィギュレーション モードで使用します。

夏時間の設定

毎年特定の曜日に夏時間が開始および終了する地域に夏時間を設定するには、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	**clock summer-time** *zone* **recurring** [*week day month hh*:*mm week day month hh*:*mm* [*offset*]]	毎年指定された日に開始および終了する夏時間を設定します。 夏時間はデフォルトでディセーブルです。パラメータなしで **clock summer-time** *zone* **recurring** を指定すると、夏時間の規則はデフォルトの米国規則に設定されます。 • *zone* には、夏時間が有効な場合に表示されるタイム ゾーンの名前（たとえば PDT）を入力します。 • （任意）*week* には、月の何番めの週かを指定します（1 ～ 5、または **last**）。 • （任意）*day* には、曜日を指定します（Sunday、Monday など）。 • （任意）*month* には、月を指定します（January、February など）。 • （任意）*hh*:*mm* には、時刻を時間（24 時間形式）と分で指定します。 • （任意）*offset* には、夏時間の間、追加する分の数を指定します。デフォルトは 60 です。
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。

clock summer-time コマンドの最初の部分は夏時間の開始時期を、2 番めの部分は終了時期を指定します。すべての時刻は、現地のタイム ゾーンを基準にしています。開始時間は標準時を基準にしています。終了時間は夏時間を基準にしています。開始月が終了月よりあとの場合は、南半球であると想定されます。

次に、夏時間が 4 月の第 1 日曜の午前 2 時に始まり、10 月の最終日曜の午前 2 時に終わるように指定する例を示します。

```
bridge(config)# clock summer-time PDT recurring 1 Sunday April 2:00 last Sunday October
2:00
```

NTP の設定

WMIC はハードウェアサポート クロックを備えておらず、外部 NTP ソースが使用できない場合は、ピアの同期化に使用される NTP マスター クロックとして機能できません。これらのデバイスは、カレンダーに対するハードウェア サポートも備えていません。そのため、**ntp update-calendar** および **ntp master** コマンドが使用できません。

NTP のデフォルト設定

表 4-4 に、NTP のデフォルト設定を示します。

表 4-4　NTP のデフォルト設定

機能	デフォルトの設定
NTP 認証	ディセーブル。認証鍵は指定されていません。
NTP ピアまたはサーバ アソシエーション	設定なし。
NTP ブロードキャスト サービス	ディセーブル。どのインターフェイスも NTP ブロードキャスト パケットを送受信しません。
NTP アクセス制限	アクセス制御は指定されていません。
NTP パケット送信元 IP アドレス	送信元アドレスは、発信インターフェイスによって決定されます。

NTP はデフォルトでディセーブルです。

NTP 認証の設定

ここに記載された手順は、NTP サーバの管理者と共同で行う必要があります。この手順で設定する情報は、時刻を同期化するために WMIC で使用される NTP サーバに対応している必要があります。

他のデバイスとのアソシエーション（正確な時刻を維持するための NTP 稼動デバイス間の通信）を認証するには、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	**ntp authenticate**	デフォルトでディセーブルに設定されている NTP 認証機能をイネーブルにします。
ステップ 3	**ntp authentication-key** *number* **md5** *value*	認証鍵を定義します。デフォルトでは何も定義されていません。 • *number* には、鍵の番号を指定します。範囲は 1 ～ 4294967295 です。 • **md5** は、Message Digest Algorithm 5（MD5）を使用してメッセージ認証がサポートされることを指定します。 • *value* には、鍵に対する 8 文字までの任意のストリングを入力します。 WMIC とデバイスが同期化されるのは、両方に認証鍵が設定され、**ntp trusted-key** *key-number* コマンドで鍵番号が指定されていない場合に限ります。
ステップ 4	**ntp trusted-key** *key-number*	1 つまたは複数の鍵番号（ステップ 3 で定義）を指定します。この WMIC をピア NTP デバイスと同期化するには、NTP パケット内にこの鍵番号を設定しなければなりません。 デフォルトでは、信頼される鍵は定義されていません。 *key-number* には、ステップ 3 で定義された鍵を指定します。 このコマンドは、信頼されていないデバイスに対して WMIC が誤って同期化されないように保護します。
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。

NTP 認証をディセーブルにするには、**no ntp authenticate** コマンドをグローバル コンフィギュレーション モードで使用します。認証鍵を削除するには、**no ntp authentication-key** *number* コマンドをグローバル コンフィギュレーション モードで使用します。デバイス ID の認証をディセーブルにするには、**no ntp trusted-key** *key-number* コマンドをグローバル コンフィギュレーション モードで使用します。

次に、NTP パケットに認証鍵 42 を設定しているデバイスとのみ同期するように WMIC を設定する例を示します。

```
bridge(config)# ntp authenticate
bridge(config)# ntp authentication-key 42 md5 aNiceKey
bridge(config)# ntp trusted-key 42
```

NTP アソシエーションの設定

NTP アソシエーションは、ピア アソシエーション（他のデバイスに対する WMIC の同期化、またはその逆方向の同期化が可能）に設定したり、サーバ アソシエーション（他のデバイスに対する WMIC の同期化は可能だが、その逆方向は不可能）に設定することもできます。

別のデバイスとの NTP アソシエーションを形成するには、イネーブル EXEC モードで次の手順を実行します。

	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	**ntp peer** *ip-address* [**version** *number*] [**key** *keyid*] [**source** *interface*] [**prefer**]	ピアを同期化するか、またはピアによって同期化されるように、WMIC のシステム クロックを設定します（ピア アソシエーション）。
	または	または
	ntp server *ip-address* [**version** *number*] [**key** *keyid*] [**source** *interface*] [**prefer**]	タイム サーバによって同期化されるように WMIC のシステム クロックを設定します（サーバ アソシエーション）。 デフォルトでは、ピアまたはサーバ アソシエーションは定義されていません。 • ピア アソシエーションの *ip-address* には、クロックの同期化を行う、またはクロックの同期化の対象となるピアの IP アドレスを指定します。サーバ アソシエーションの場合は、クロックの同期化を行うタイム サーバの IP アドレスを指定します。 • （任意）*number* には、NTP のバージョン番号を指定します。範囲は 1 ～ 3 です。デフォルトでは、バージョン 3 が選択されています。 • （任意）*keyid* には、**ntp authentication-key** コマンドで定義された認証鍵を入力します。 • （任意）*interface* には、IP の送信元アドレスの取得元となるるインターフェイスを指定します。デフォルトでは、送信元 IP アドレスは発信インターフェイスから取得されます。 • （任意）**prefer** キーワードを指定すると、このピアまたはサーバは同期化を提供する優先ピアまたはサーバになります。これにより、ピア間およびサーバ間の切り換え回数が減少します。
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。

設定する必要があるのは、アソシエーションの一端のみです。他端のデバイスには自動的にアソシエーションが設定されます。デフォルトの NTP バージョン（バージョン 3）を使用しているにもかかわらず、NTP 同期化が発生しない場合は、NTP バージョン 2 を使用してください。インターネット上の多くの NTP サーバでは、バージョン 2 が稼動しています。

ピアまたはサーバ アソシエーションを削除するには、**no ntp peer** *ip-address* または **no ntp server** *ip-address* コマンドをグローバル コンフィギュレーション モードで使用します。

次に、NTP バージョン 2 を使用して IP アドレス 172.16.22.44 にあるピアのクロックとシステム クロックを同期化するように WMIC を設定する例を示します。

```
bridge(config)# ntp server 172.16.22.44 version 2
```

NTP ブロードキャスト サービスの設定

NTP が稼動するデバイス間の通信（*アソシエーション*）は、通常スタティックに設定されます。各デバイスには、アソシエーションを形成する必要があるすべてのデバイスの IP アドレスが付与されます。アソシエーションを形成するデバイス ペア間で NTP メッセージを交換することによって、正確な時刻の維持が可能になります。ただし、LAN 環境では、代わりに IP ブロードキャスト メッセージを使用するように NTP を設定できます。この代替手段では、単にブロードキャスト メッセージを送受信するように各デバイスを設定すればよいので、設定の複雑さが緩和されます。ただし、この場合は、情報の流れは一方向に限定されます。

ネットワーク上で時刻情報をブロードキャストする NTP ブロードキャスト サーバ（ルータなど）がある場合、WMIC はインターフェイスごとに NTP ブロードキャスト パケットを送受信できます。WMIC は NTP ブロードキャスト パケットをピアへ送信して、ピアを同期化できます。また、WMIC は NTP ブロードキャスト パケットを受信して、自身のクロックを同期化することもできます。ここでは、NTP ブロードキャスト パケットの送信手順および受信手順について説明します。

NTP ブロードキャスト パケットをピアに送信し、ピアが自身のクロックを WMIC に同期化するよう、WMIC 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	**interface** *interface-id*	インターフェイス コンフィギュレーション モードを開始し、NTP ブロードキャスト パケットを送信するインターフェイスを指定します。
ステップ 3	**ntp broadcast** [**version** *number*] [**key** *keyid*] [***destination-address***]	インターフェイスからピアへの NTP ブロードキャスト パケットの送信をイネーブルにします。 デフォルトでは、この機能はすべてのインターフェイスでディセーブルです。 • （任意）*number* には、NTP のバージョン番号を指定します。範囲は 1 ～ 3 です。バージョンを指定しない場合は、バージョン 3 が使用されます。 • （任意）*keyid* には、ピアにパケットを送信するときに使用する認証鍵を指定します。 • （任意）*destination-address* には、この WMIC に対してクロックを同期化しているピアの IP アドレスを指定します。
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。

インターフェイスからの NTP ブロードキャスト パケットの送信をディセーブルにするには、**no ntp broadcast** インターフェイス コンフィギュレーション コマンドを使用します。

次に、NTP バージョン 2 パケットを送信するようにインターフェイスを設定する例を示します。

```
bridge(config)# interface gigabitethernet0/1
bridge(config-if)# ntp broadcast version 2
```

次の手順に従って NTP ブロードキャスト パケットを受信するように、接続されているピアを設定します。NTP ブロードキャスト パケットを接続されているピアから受信するように WMIC 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	**interface** *interface-id*	インターフェイス コンフィギュレーション モードを開始し、NTP ブロードキャスト パケットを受信するインターフェイスを指定します。
ステップ 3	**ntp broadcast client**	インターフェイスでの NTP ブロードキャスト パケットの受信をイネーブルにします。 デフォルトでは、インターフェイスは NTP ブロードキャスト パケットを受信しません。
ステップ 4	**exit**	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 5	**ntp broadcastdelay** *microseconds*	（任意）WMIC と NTP ブロードキャスト サーバ間のラウンドトリップ遅延の予測値を変更します。 デフォルトは 3000 ミリ秒です。指定できる範囲は 1 ～ 999999 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。

インターフェイスでの NTP ブロードキャスト パケットの受信をディセーブルにするには、**no ntp broadcast client** コマンドをコンフィギュレーション モードで使用します。ラウンドトリップ遅延の予測値をデフォルト設定に変更するには、**no ntp broadcastdelay** コマンドをグローバル コンフィギュレーション モードで使用します。

次に、NTP ブロードキャスト パケットを受信するようにインターフェイスを設定する例を示します。

```
bridge(config)# interface gigabitethernet0/1
bridge(config-if)# ntp broadcast client
```

NTP アクセス制限の設定

NTP アクセスを制御するには、アクセス リストを使用できます。

アクセス グループの作成および基本 IP アクセス リストの割り当て

アクセス リストを使用して NTP サービスへのアクセスを制御するには、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	**ntp access-group** {**query-only** | **serve-only** | **serve** | **peer**} *access-list-number*	アクセス グループを作成し、基本 IP アクセス リストを適用します。 キーワードの意味は次のとおりです。 • **query-only**：NTP 制御クエリーのみを許可します。 • **serve-only**：時刻要求のみを許可します。 • **serve**：時刻要求と NTP 制御クエリーは許可しますが、リモート デバイスに対する WMIC の同期化は許可しません。 • **peer**：時刻要求と NTP 制御クエリー、およびリモート デバイスに対する WMIC の同期化を許可します。 *access-list-number* には、1 ～ 99 の標準の IP アクセス リスト番号を入力します。
ステップ 3	**access-list** *access-list-number* **permit** *source* [***source-wildcard***]	アクセス リストを作成します。 • *access-list-number* には、ステップ 2 で指定した番号を入力します。 • **permit** キーワードを入力すると、条件が一致した場合にアクセスを許可します。 • *source* には、WMIC へのアクセスが許可されているデバイスの IP アドレスを入力します。 • （任意）*source-wildcard* には、送信元に適用するワイルドカード ビットを入力します。 **（注）** アクセス リストを作成するときは、アクセス リストの末尾に暗黙的な拒否ステートメントがデフォルトで存在し、それ以前のステートメントで一致が見つからなかったすべてのパケットに適用されることに注意してください。
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。

アクセス グループのキーワードは、次の順序で（制限が緩い順に）スキャンされます。

1. **peer**：時刻要求と NTP 制御クエリーを許可し、さらに、アクセス リストの基準を満たすアドレスを持つデバイスとの WMIC の同期化を許可します。
2. **serve**：時刻要求と NTP 制御クエリーを許可しますが、アクセス リストの基準を満たすアドレスを持つデバイスとの WMIC の同期化は許可しません。

3. **serve-only**：アクセス リストの基準を満たすアドレスを持つデバイスからの時刻要求のみを許可します。
4. **query-only**：アクセス リストの基準を満たすアドレスを持つデバイスからの NTP 制御クエリーのみを許可します。

送信元 IP アドレスとアクセス リストが複数のアクセス タイプで一致する場合は、最初のアクセス タイプが認可されます。アクセス グループを指定しなかった場合は、すべてのアクセス タイプがすべてのデバイスに対して認可されます。いずれかのアクセス グループが指定されている場合は、指定されたアクセス タイプのみが認可されます。

WMIC の NTP サービスに対するアクセス制御を削除するには、**no ntp access-group** {**query-only** | **serve-only** | **serve** | **peer**} コマンドをグローバル コンフィギュレーション モードで使用します。

次に、アクセス リスト 99 でピアとの同期化を許可するように WMIC を設定する例を示します。ただし、アクセス リスト 42 では、時刻要求のみが許可されるように WMIC のアクセスは制限されています。

```
bridge# configure terminal
bridge(config)# ntp access-group peer 99
bridge(config)# ntp access-group serve-only 42
bridge(config)# access-list 99 permit 172.20.130.5
bridge(config)# access list 42 permit 172.20.130.6
```

特定のインターフェイスでの NTP サービスのディセーブル化

NTP は、すべてのインターフェイスでデフォルトでイネーブルです。

特定のインターフェイスで NTP パケットの受信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	**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	**interface** *interface-id*	インターフェイス コンフィギュレーション モードを開始し、ディセーブルにするインターフェイスを指定します。
ステップ 3	**ntp disable**	インターフェイスでの NTP パケットの受信をディセーブルにします。 デフォルトでは、すべてのインターフェイスが NTP パケットを受信します。
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。

インターフェイスでの NTP パケットの受信を再イネーブルにするには、**no ntp disable** コマンドをインターフェイス コンフィギュレーション モードで使用します。

NTP パケットの送信元 IP アドレスの設定

WMIC が NTP パケットを送信すると、送信元 IP アドレスは、通常 NTP パケットが送信されたインターフェイスのアドレスに設定されます。すべての NTP パケットに特定の送信元 IP アドレスを使用する場合は、**ntp source** コマンドをグローバル コンフィギュレーション モードで使用します。アドレスは指定されたインターフェイスから取得されます。このコマンドは、特定のインターフェイスのアドレスを返信パケットの宛先として使用できない場合に便利です。

送信元 IP アドレスの取得元となるインターフェイスを設定するには、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	**ntp source** *type number*	送信元 IP アドレスの取得元となるインターフェイスのタイプおよび番号を指定します。 デフォルトでは、送信元アドレスは発信インターフェイスによって決定されます。
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。

指定されたインターフェイスは、すべての宛先に送信されるすべてのパケットの送信元アドレスに使用されます。送信元アドレスを特定のアソシエーションに使用する場合は、グローバル コンフィギュレーション モードの **ntp peer** または **ntp server** コマンドで **source** キーワードを使用します（「NTP アソシエーションの設定」（P.4-49）を参照）。

NTP 設定の表示

NTP 情報を表示するには、イネーブル EXEC モードで次のコマンドを使用します。

- **show ntp associations** [**detail**]
- **show ntp status**

これらの出力に表示されるフィールドの詳細については、『*Cisco IOS Configuration Fundamentals Command Reference for Release 12.1*』を参照してください。

ユーザの居住地域の夏時間が定期的なパターンに従わない場合に、次の夏時間のイベントの正確な日時を設定するには、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	**clock summer-time** *zone* **date** [*month date year hh*:*mm month date year hh*:*mm* [*offset*]] または **clock summer-time** *zone* **date** [*date month year hh*:*mm date month year hh*:*mm* [*offset*]]	最初の日付で夏時間の開始日を、2 番めの日付で終了日を設定します。 夏時間はデフォルトでディセーブルです。 • *zone* には、夏時間が有効な場合に表示されるタイム ゾーンの名前（たとえば PDT）を入力します。 • （任意）*week* には、月の何番めの週かを指定します（1 ～ 5、または **last**）。 • （任意）*day* には、曜日を指定します（Sunday、Monday など）。 • （任意）*month* には、月を指定します（January、February など）。 • （任意）*hh*:*mm* には、時刻を時間（24 時間形式）と分で指定します。 • （任意）*offset* には、夏時間の間、追加する分の数を指定します。デフォルトは 60 です。
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。

clock summer-time コマンドの最初の部分は夏時間の開始時期を、2 番めの部分は終了時期を指定します。すべての時刻は、現地のタイム ゾーンを基準にしています。開始時間は標準時を基準にしています。終了時間は夏時間を基準にしています。開始月が終了月よりあとの場合は、南半球であると想定されます。

夏時間をディセーブルにするには、**no clock summer-time** コマンドをグローバル コンフィギュレーション モードで使用します。

次に、夏時間が 2005 年 10 月 12 日の午前 2 時に開始し、2006 年 4 月 26 日の午前 2 時に終了するように設定する例を示します。

```
bridge(config)# clock summer-time pdt date 12 October 2000 2:00 26 April 2001 2:00
```