CHAPTER 2

認証サービスおよびアカウンティング サービスの設定

この章では、ユーザ認証およびアカウンティング（AAA）サービスを実行し、Cisco Application Control Engine（ACE）モジュールにアクセスするユーザにレベルの高いセキュリティを提供するために ACE を設定する方法を説明します。AAA サービスを利用すると、複数の AAA サーバを使用して ACE へのアクセス コントロールと ACE にアクセスしたユーザ活動の追跡を行えます。ACE は、ユーザ名とパスワードの組み合わせに基づき、ローカル データベースを使用するローカル ユーザ認証、または外部 AAA サーバを使用するリモート ユーザ認証およびアカウンティングを実行します。

主な内容は、次のとおりです。

- AAA の概要
- 認証とアカウンティング設定のクイック スタート
- AAA サーバの設定
- ユーザ アカウントの作成
- RADIUS、TACACS+、または LDAP サーバのクライアントとしての ACE の設定
- ログイン認証手段の定義
- デフォルトのアカウンティング手段の定義
- AAA のステータスと統計情報の表示

AAA の概要

ACE へのユーザ アクセスに対して、AAA は認証サービスとアカウンティング サービスを組み合わせたマネジメント セキュリティを 提供します。ユーザが誰で、ユーザが何を行ったかの情報は AAA が ACE に知らせます。認証だけを利用することも、アカウンティングを併用することもできます。ACE は、コマンドライン インターフェイス（CLI）または SNMP（Simple Network Management Protocol）を含む ACE への管理アクセス手段に対してセキュリティを提供します。

ACE の CLI は、コンソール ポートから利用することも、あるいは Telnet または SSH セッションにより利用することもできます。Telnet もしくは SSH コネクションのいずれかを利用して ACE へログインする際は、ACE が AAA サーバを用いるように設定されていると、一時的な SNMP ユーザ エントリが自動的に作成されます。SNMPv3 プロトコル データ ユニットが SNMPv3 ユーザとして関連付けられた Telnet または SSH ログイン名を持つ場合は、ACE により認証が行われます。

認証プロセスの一部として、ACE は各ユーザを特定の仮想コンテキストにおけるユーザ ロールとドメインで構成されるアクセス権限ペアに関連付けます。各仮想コンテキストは独立したデバイスのように動作し、独自の設定内容、セキュリティ ポリシー、インターフェイス、ドメインを持ちます。1 つのユーザ コンテキストは他のユーザ コンテキストから独立して管理できます。1 つのドメインには 1 つの名前空間があり、ユーザはこの空間内で操作を行います。各ユーザは最低 1 つのドメインに関連付けられます。ユーザがドメイン内のオブジェクトに実行できる操作の種類と、そのユーザが利用できるコマンド セットは、ユーザに割り当てられたロールにより決まります。各コンテキスト内では仮想 AAA インスタンスが実行されています。このインスタンスは、ログインしようとするユーザの認証サービスや、ユーザ アクティビティをログに記録するアカウンティング サービスを提供します。

ACE 内の仮想コンテキストは、それぞれが独自の IP アドレスを持っています。ACE 内の各仮想コンテキストへのアクセスは、コンソール ポート、または Telnet か SSH のセッションによりこの IP アドレスを指定して行うことができます。また、ユーザはこの IP アドレスを使用して SNMP 要求を ACE に送信できます。

（注）　コンソール ポートからアクセスできるのは管理コンテキストのみです。他のすべてのコンテキストには Telnet または SSH を使用しないとアクセスできません。

各仮想コンテキストの管理者は、他のコンテキストから切り離された状態で次のアクションを実行できます。

- 別々の AAA サーバとそのパラメータの設定
- 複数のコンテキストにまたがる同一ユーザ名の作成。およびコンテキスト内および複数ドメイン内での、ユーザ名の重複のない固有ロールへの関連付け
- AAA サーバの共有、ただし各ユーザの認証は、それぞれのコンテキストに対して行い、かつ同一のパスワードにより行なう必要があります。
- ユーザ アカウンティング アクティビティのログ作成（ユーザがサイン インしたコンテキストにより区分）
- その仮想コンテキストで現在認証されているユーザを表示

特定の IP アドレスから ACE にアクセスする各ユーザに対しての認証は 1 度だけ必要になります。ユーザ認証手続きは、ACE 上で認証セッションが失効するまで維持されます。

AAA クライアントは ACE が実行し、このクライアントがユーザと AAA サーバを仲介します。ACE が 1 つのインターフェイス上で認証情報（ユーザ名とパスワード）を各ユーザに求め、もうひとつのインターフェイス上で、認証を求めるユーザが正しい認証情報を提供しているかどうかを指定の AAA サーバに対して問い合わせます。その応答により、ACE へのアクセスを許可してよいかどうかを判断します。

ACE は、ACE 自身が持っているローカル ユーザ データベース、またはリモート AAA サーバを利用して認証を実行します。ACE によるリモート認証とアクセス権限指定には、RADIUS（Remote Access Dial-In User Service）のほか、TACACS+（Terminal Access Controller Access Control System Plus）、または LDAP（Lightweight Directory Access Protocol (v3)）サーバを利用できます。

このセクションの内容は、次のとおりです。

- 認証の概要
- アカウンティング サービスの概要

ローカル データベースとリモート サーバのサポート

ACE は ACE に保存されているローカル データベースを利用したローカル認証、または 1 つ以上の AAA サーバを利用したリモート認証をサポートします。AAA リモート サーバは TACACS+、RADIUS、LDAP で分かれる独立したグループにまとめられます。サーバ グループに対しては、ACE はグループ内で最初のアクティブなサーバを選んで利用します。

（注） 「最初」とはそれぞれのサーバが設定される際の順序で決まります。

ユーザが ACE にログインすると、設定されている先頭のサーバから始まって ACE にサーバ応答が返るまで、1 つずつ順番にアクセスが行われます。

サーバ グループ 認証を行うようにサーバ グループが設定されている場合、ACE は認証要求をグループの先頭サーバに送信し、次の手順に従います。

- リモート AAA サーバが応答に失敗すると、ACE はグループ内のいずれかのリモート サーバが認証要求に応答するまで、次のサーバにコンタクトします。
- そのサーバ グループ内のすべての AAA サーバが応答に失敗すると、ACE はあらかじめ設定されている次のサーバ グループ内の AAA サーバにコンタクトします。
- すべてのリモート AAA サーバが応答に失敗すると、デフォルトでは ACE はローカル データベースを使って認証を試みます。

ユーザ名とパスワードがローカルまたはリモートで適切に認証されると、ACE はユーザにログインを許可し、ユーザに専用のロールを割り当てます（**role** コマンドで指定されたとおりに実行。ロールは各ユーザが使用できるコマンドの種類とリソースを決定します）。

ネットワーク 接続に障害が発生した場合は、グループの各サーバをアクティブとも非アクティブとも断定できません。しかし AAA サーバ選択に利用されるポリシーはサーバの状態を考慮に入れています。ACE は、タイムアウトしたサーバに認証要求を送ることで AAA サーバの稼動状態を監視します。ユーザ指定の回数以内に ACE がサーバから確認応答を受信できなかった場合は、そのサーバを無応答と判断し、サーバ グループで次に利用可能なサーバにコンタクトします。

AAA サーバに対してデッドタイム間隔が指定されている場合は、サーバ A への接続が失敗すると、ACE はサーバ A をサービス停止として記録し、デッドタイム間隔の経過中はサーバ A を無視します。次に ACE は、その AAA サーバが利用可能で認証要求を受信できるかどうかを確認するために、プローブ アクセス要求パケットを送信します。サーバがプローブ アクセス要求 パケットに応答すると、サーバ A へのコネクションが回復します。

このセクションの内容は、次のとおりです。

- ローカル データベース
- TACACS+ サーバ
- RADIUS サーバ
- LDAP ディレクトリ サーバ

ローカル データベース

CLI へのアクセス認証に対しては、ユーザ アカウントのアクセス権に対して ACE のローカル データベースを使用するように設定できます。ユーザがコンソール ポートまたは Telnet または SSH セッションを使用して ACE の CLI にアクセスを試みると、ACE はローカル ユーザ データベースを参照してユーザ名とパスワードを調べます。デフォルトでは、各ユーザを Network-Monitor ロールとして想定し、すべてのドメインで操作を許可します。

ローカル認証をフォールバック手段として指定している場合は、ACE はサーバ グループ内に指定した AAA サーバが認証できないときにローカル データベースにアクセスし、ユーザ認証とアカウンティングを試みます。

TACACS+ サーバ

TACACS+ による ACE へのユーザ アクセス コントロールでは、ACE と中央データベースの間で NAS（ネットワーク アクセス サーバ）情報を交換することにより、ユーザの身元が判断されます。TACACS+ は、RFC 1492 により規程されている UDP（User Datagram Protocol）ベースのアクセス コントロール プロトコルである TACACS の拡張版です。TACACS+ サーバと ACE 上の TACACS+ デーモンの間に流れるすべてのトラフィックを安全に配信し、暗号化するために、TACACS+ では TCP を利用します。

TACACS+ サーバは、ユーザ認証機能とアカウンティング機能を提供します。これらのサービスはいずれも TACACS+ の一部ですが、互いに独立しているため、TACACS+ がその一部に限定して設定されていたり、すべてを利用できるように設定されている可能性があります。

ユーザ パスワードの情報は TACACS+ プロトコルにより MD5 暗号アルゴリズムを利用して暗号化され、TACACS+ パケット ヘッダが付加されます。このヘッダ情報には、送信パケット（たとえば認証パケットなど）のタイプ、パケット順序番号、使用されている暗号の種類、パケットの全長が記述されています。パケットは TACACS+ プロトコルを使って TACACS+ サーバに転送されます。

ACE と TACACS+ サーバの間のセキュリティを確保するため、ACE と TACACS+ サーバの間のすべての通信に対して暗号キー（共有暗号鍵）を指定できます。適切な運用には、ACE と TACACS+ サーバの双方で同一の暗号キーを指定しなければなりません。

RADIUS サーバ

RADIUS はクライアント / サーバ型のアクセス プロトコルで、ACE に接続するユーザを NAS が認証する際に用いられます。ここでは NAS がクライアントとして機能し、ユーザ情報を 1 つまたは複数の RADIUS サーバに引き渡します。NAS は RADIUS サーバから受信した応答に基づいて、ネットワーク アクセスをユーザに許可または拒否します。RADIUS は RADIUS クライアントと RADIUS サーバの間のコネクションレス転送に UDP を用います。RADIUS プロトコルの動作の詳細については、RFC 2138 を参照してください。

ACE と RADIUS サーバの間のセキュリティを確保するため、ACE と RADIUS サーバの間のすべての通信に対して暗号キー（共有暗号鍵）を指定できます。適切な運用には、ACE と RADIUS サーバの双方で同一の暗号キーを指定しなければなりません。

LDAP ディレクトリ サーバ

LDAP は、X.500 Directory Access Protocol（DAP）ディレクトリ サービスへのアクセスを目的とするオープン標準規格のクライアント / サーバ型認証プロトコルです。LDAP は TCP/IP または他の種類のコネクション指向の転送サービスを利用します。ACE は認証と検索操作を単純化するために LDAP バージョン 3 のみをサポートしています。LDAP プロトコルの動作の詳細については、RFC 2251 を参照してください。

LDAP の情報モデルはエントリを基盤としています。1 つのエントリは属性の集合体であり、グローバルに重複のない名前（DN）を持っています。この DN は LDAP データベース中のエントリを参照するために使用されます。各エントリには、エントリの内容を記述する 1 つまたは複数の属性が含まれており、各属性には 1 つのタイプと 1 つまたは複数の値が定義されています。タイプは略号の文字列であり、たとえば、「common name」に対応する “cn” や、E メールを表す “mail” などがあります。

LDAP クライアント（ACE）が LDAP サーバによるユーザ認証を要求し、サーバが管理するディレクトリ データベースに対する検索を要求することにより、そのユーザのプロファイルが得られます。LDAP サーバはエントリを集めたディレクトリを管理しており、このディレクトリはディレクトリ情報ツリー（DIT; Directory Information Tree）と呼ばれる階層構造になっています。

LDAP クライアントはこのディレクトリ データに対する操作を実行します。LDAP を利用すると、任意にユーザが設定した種別に合致するデータをディレクトリから検索できます。ユーザはどの部分のディレクトリを検索するか、何の情報を返すかを指定できます。検索したいディレクトリ データは、ブール式の条件式を用いる検索フィルタにより指定します。

（注）　ACE は LDAP サーバを利用した更新、比較、キャンセルの操作をサポートしません。さらに、要求していない通知を LDAP から受け取ることもサポートしません。サポートするメッセージには、bindRequest、bindResponse、unbindRequest、searchRequest、searchResEntry、および searchResDone が含まれます。

認証の概要

認証の役割は、有効なユーザ名とパスワードの入力を要求することにより、ACE CLI へのユーザ アクセスを制限することにあります。ACE の CLI は、コンソール ポートから利用することも、あるいは Telnet または SSH セッションにより利用することもできます。ACE への各管理アクセス パスには、次にあげるセキュリティ制御オプションのうち、1 つまたは複数を設定できます。ローカル データベース、リモート（RADIUS、TACACS+、LDAP）、またはパスワード確認なし。

ホストは有効なユーザ名とパスワードの入力を ACE から求められます。指定された RADIUS、TACACS+、または LDAP サーバがユーザ名とパスワードの認証を終えると、ACE はそのユーザにアクセス権を与えます。

アカウンティング サービスの概要

アカウンティング サービスは、ACE で各ユーザが実行する管理セッション中の有用な情報のログを記録して保存します。この情報を利用して作成したレポートは、トラブルシューティングと監査に利用できます。アカウンティング サービスのログは、ACE にローカルに保存することも、リモート AAA サーバに送信することもできます。ACE もユーザを認証するように設定されていると、AAA サーバはアカウンティング情報をユーザ名により保管します。アカウンティング情報には ACE に入力されたユーザ コマンドのほか、各セッションの実行時間、セッションの開始と終了の時刻も含まれます。

ACE のユーザ コマンドが TACACS+ または RADIUS サーバ上でログに記録される場合は、サーバは各コマンドの前に「<0:>」または「<1:>」を付加して、ACE CLI からユーザがコマンドを入力したときの成功または失敗を示します。たとえば、ACE 上でユーザがあるコマンドを入力しようとした際に、ユーザが適切なロール権限を持っていなかった場合は「<1:>」を表示して失敗を示します。

認証とアカウンティング設定のクイック スタート

ACE で認証とアカウンティングの情報作成、設定を行うための手順の概要を表 2-1 に示します。各ステップには作業を完了するための CLI コマンドを示します。

表 2-1　認証とアカウンティングの設定のクイック スタート

作業内容とコマンドの例
1. TACACS+、RADIUS、または LDAP サーバ上で、認証とアカウンティングのサービスを設定します。

2. 複数のコンテキストで操作している場合、対象のコンテキストで操作しているか CLI プロンプトを確認します。必要に応じて、正しいコンテキストに直接ログインするか変更してください。
`host1/Admin# changeto C1`
`host1/C1#`
ここからは特に明記しないかぎり、表中の例では管理コンテキストを使用します。ACECLI アクセスの認証に必要なローカル データベースへのアクセスを行うためのコンテキストとユーザ アカウントの作成の詳細については、『*Cisco Application Control Engine Module Virtualization Configuration Guide*』を参照してください。

3. **config** と入力してコンフィギュレーション モードに入ります。
`host1/Admin# config`
`Enter configuration commands, one per line. End with CNTL/Z`
`host1/Admin(config)#`

4. AAA サーバのパラメータを個別に設定します。たとえば、RADIUS サーバの認証パラメータを設定するには、次のように入力します。
`host1/Admin(config)# radius-server host 192.168.2.3 key HostKey`
`host1/Admin(config)# radius-server host 192.168.2.3 key 7 secret_1256`
`host1/Admin(config)# radius-server host 192.168.2.3 auth-port 1645`
`host1/Admin(config)# radius-server host 192.168.2.3 acct-port 1646`
`host1/Admin(config)# radius-server host 192.168.2.3 authentication`
`host1/Admin(config)# radius-server host 192.168.2.3 accounting`
`host1/Admin(config)# radius-server host 192.168.2.3 timeout 25`
`host1/Admin(config)# radius-server host 192.168.2.3 retransmit 3`

表 2-1　認証とアカウンティングの設定のクイック スタート（続き）

作業内容とコマンドの例

5. TACACS+、RADIUS、または LDAP サーバ用に独立したサーバ グループを設定します。たとえば、RADIUS サーバ グループを作成するには、次のように入力します。
`host1/Admin(config)# aaa group server radius RAD_Server_Group1`
`host1/Admin(config-radius)# server 192.168.252.1`
`host1/Admin(config-radius)# server 192.168.252.2`
`host1/Admin(config-radius)# server 192.168.252.3`
`host1/Admin(config-radius)# deadtime 15`

6. ACE CLI へのログインに使用する認証手段を設定します。 `host1/Admin(config)# aaa authentication login console group RAD_Server_Group1 local none`
7. デフォルトのアカウンティング手段を設定します。 `host1/Admin(config)# aaa accounting default group RAD_Server_Group1 local`

8. （任意）設定の変更内容をフラッシュ メモリに保存します。
`host1/Admin(config)# exit`
`host1/Admin# copy running-config startup-config`

AAA サーバの設定

この章では、Cisco Secure Access Control Server（ACS）などの TACACS+ または RADIUS サーバのセットアップ方法を説明します。また、OpenLDAP Project から入手できる OpenLDAP ソフトウェアなど、LDAP ディレクトリ サーバをセットアップする際の、一般的な注意事項も記載します。この章は、AAA サーバと、AAA クライアントとして機能する ACE が適切に通信を行うための手引きとして作成されています。

Cisco Secure ACS や OpenLDAP ソフトウェア、その他の AAA サーバの設定の詳細については、各ソフトウェアのマニュアルを参照してください。

TACACS+ サーバの設定

このセクションの内容は、次のとおりです。

- TACACS+ サーバにおける認証情報の設定
- TACACS+ サーバ アカウンティング サービスの設定
- TACACS+ サーバで仮想化をサポートするためのプライベート属性の定義

（注） ACE が TACACS+ サーバを利用して適切にユーザ認証を行うには、ユーザ名とパスワードが ACE と TACACS+ サーバの両方で同一でなければなりません。

TACACS+ サーバにおける認証情報の設定

Cisco Secure ACS で TACACS+ の認証情報を設定する手順は、次のとおりです。

ステップ 1　Cisco Secure ACS HTML インターフェイスの Network Configuration セクションを開き、Add AAA Client ページを開きます。

ステップ 2　次のセクションを設定します。

- AAA クライアント ホスト名 — ACE に割り当てる名前を入力します。
- AAA クライアント IP アドレス — TACACS+ サーバとの通信に使用するイーサネット インターフェイスの IP アドレスを入力します。

- キー **—** ACE と Cisco Secure ACS がトランザクションを認証するために使用する共有暗号鍵を入力します。共有暗号鍵は、Cisco Secure ACS と ACE の両方で同一のものを指定しなくてはなりません。キーは大文字と小文字を区別します。
- 認証に使用 **—** [**TACACS+ (Cisco IOS)**] を選択します。

（注） [TACACS+ (Cisco IOS)] ドロップダウン項目は、Cisco TACACS+ 認証機能に付けるタイトルです。[TACACS+ (Cisco IOS)] セレクションは、TACACS+ 認証プロトコルをサポートする Cisco Systems のアクセス サーバ、ルータ、ファイアウォールを使用する場合の TACACS+ オプションを有効化します。これには ACE のサポートも含まれます。

ステップ 3　[**Submit + Restart**] をクリックします。

TACACS+ サーバ アカウンティング サービスの設定

Cisco Secure ACS に必要な TACACS+ のアカウンティング サービスを設定する手順は、次のとおりです。

ステップ 1　Cisco Secure ACS インターフェイスの System Configuration セクション、Logging Configuration ページで、[**CSV TACACS+ Accounting**] をクリックします。CSV TACACS+ Accounting File Configuration ページが表示されます。

ステップ 2　[**Log to CSV TACACS+ Accounting report**] チェックボックスが選択されていることを確認します。

ステップ 3　Attributes カラムの Select Columns To Log の下で、ログに記録する属性をクリックします。[**->**] をクリックして、属性を Logged Attributes カラムに移動させます。[**Up**] または [**Down**] をクリックして、ログの中の希望する位置にこの属性のカラムを移動させます。必要な属性すべてが Logged Attributes カラムで所定の位置に収まるまで繰り返します。

ステップ 4　属性を Logged Attributes へ移し終えたら [**Submit**] をクリックします。

TACACS+ サーバで仮想化をサポートするためのプライベート属性の定義

複数のコンテキストにまたがる同一のユーザ名の作成と、コンテキスト内および複数ドメイン内での専用ロールへのユーザ名の関連付けを行えます。各コンテキストは TACACS+ サーバを共有できますが、ユーザはコンテキストごとの認証とコンテキストで同一のパスワードが必要です。

ユーザが ACE へのログインを試みると、認証を行うために ACE の TACACS+ クライアントがリモート TACACS+ サーバにユーザ名とパスワードを送信します。TACACS+ サーバは認証の一環としてユーザ プロファイルを取得します。ユーザ認証に成功すると、TACACS+ サーバは ACE の TACACS+ クライアントに認証ステータスとともにユーザ プロファイルを返します。ログインを試みるユーザに関連付けられているコンテキストが、TACACS+ サーバから取得したユーザ プロファイルのコンテキストに合致した場合は、TACACS+ クライアントはリモート サーバから得たユーザ プロファイルに更新します。しかしコンテキストが合致しなかった場合は、ユーザ プロファイルはデフォルト ロール（Network-Monitor）とデフォルト ドメイン（default-domain）に更新されます。

ユーザにシェル コマンド 認証を設定するには、TACACS+ サーバ上のユーザ プロファイルに Exec シェルの実行を設定します。次の形式に従った文字列を使ってカスタム属性を定義します。

シェル :< コンテキスト名 >=< ロール >< ドメイン 1>< ドメイン 2>...< ドメイン N>

または

シェル :< コンテキスト名 >*< ロール >< ドメイン 1>< ドメイン 2>...< ドメイン N>

（注） Cisco IOS のコマンド 認証を使用するときは、シェル コマンドの文字列に等号（=）ではなく、必ずアスタリスク（*）を使用します。等号は、Cisco IOS ソフトウェアで必須フィールドが続くことを示すものです。Cisco IOS ソフトウェアは、role フィールドを認識しないため、この場合に等号を使用すると、Cisco IOS の認証が失敗します。

（注） ユーザ プロファイル属性は TACACS+ サーバ グループに対して重要な設定機能を提供します。認証実行中にユーザ プロファイルがサーバから取得できなかった場合、またはプロファイルをサーバから取得できてもプロファイル中のコンテキスト名（複数の場合もある）が、ユーザがログインを試みているコンテキストに合致しなかった場合は、認証に成功するとデフォルト　　ロール（Network-Monitor）とデフォルト ドメイン（default-domain）がそのユーザに割り当てられます。

Cisco Secure ACS で TACACS+ のロール情報およびドメイン情報を設定する手順は、次のとおりです。

ステップ 1　Cisco Secure ACS HTML インターフェイスの Interface Configuration セクションを開いて、TACACS+ (Cisco IOS) ページにアクセスします。次のアクションを実行します。

- **a.** このページの TACACS+ Services セクション、User カラムまたは Group カラム（設定による）の下で、[**Shell (exec)**] チェックボックスをオンにします。
- **b.** Advanced Configuration Options セクションの下で、[**Display a window for each service selected in which you can enter customized TACACS+ attributes**] チェックボックスをオンにします。
- **c.** [**Submit**] をクリックします。

ステップ 2　Cisco Secure ACS HTML インターフェイスの Interface Configuration セクションの Advanced Options ページを開きます。次のアクションを実行します。

- **a.** [**Per-user TACACS+/RADIUS Attributes**] チェックボックスをオンにします。
- **b.** [**Submit**] をクリックします。

ステップ 3　Cisco Secure ACS HTML インターフェイスの User Setup セクションを開いて、仮想化に使用するユーザ プロファイル属性を定義したい既存のユーザ名をダブルクリックします。User Setup ページが表示されます。

ステップ 4　このページの TACACS+ Settings セクションの下で、次の設定を行います。

- [**Shell (exec)**] チェックボックスをオンにします。
- [**Custom attributes**] チェックボックスをオンにします。
- 指定するコンテキストに用いるユーザ　ロールと関連ドメインを、次の形式に従って Custom 属性の下のテキスト ボックスに入力します。

```
shell:<contextname>=<role> <domain1> <domain2>...<domainN>
```

たとえば、選択したユーザを C1 コンテキストとともにロール ROLE1 とドメイン DOMAIN1 に割り当てるには、**shell:C1=ROLE1 DOMAIN1** と入力します。

等号（=）は次のようにアスタリスク（*）に置き換えることもできます。

```
shell:<contextname>*<role> <domain1> <domain2>...<domainN>
```

Cisco IOS コマンド認証も使用している場合は、上記のシェル文字列を使用します。

ステップ 5　未知のサービスを許可するため、このページの Checking This option Will PERMIT all UNKNOWN Services セクションの下で、[**Default (Undefined) Services**] チェックボックスをオンにします。

ステップ 6　TACACS+ のロール設定とドメイン設定を終えたら [**Submit**] をクリックします。

たとえば、USER1 がロール ADMIN とドメイン MYDOMAIN1 に割り当てられている場合（shell:Admin=ADMIN MYDOMAIN1）は、次のいずれかが発生します。

- USER1 が Admin コンテキストからログインした場合は、そのユーザは自動的に Admin ロールと MyDomain1 ドメインに割り当てられます。
- USER1　が別のコンテキストからログインすると、そのユーザは自動的にデフォルト ロール（Network-Monitor）とデフォルト ドメイン（default-domain）に割り当てられます。この場合、認証の実行中にユーザ プロファイル属性が TACACS+ サーバから取得されることはありません。

RADIUS サーバの設定

このセクションの内容は、次のとおりです。

RADIUS サーバ認証サービスの設定

Cisco Secure ACS で RADIUS の認証情報を設定する手順は、次のとおりです。

ステップ 1　Cisco Secure ACS HTML インターフェイスの Network Configuration セクションを開きます。

（注）　Network Device Groups（NDG）を使用している場合は、AAA クライアント エントリに追加する NDG の名前もクリックする必要があります。

ステップ 2　AAA Clients テーブルの下で [**Add Entry**] を選択します。Add AAA Client ページが表示されます。

ステップ 3　Add AAA Client ページでエントリを設定する手順は、次のとおりです。

- AAA クライアント ホスト名 — ACE に割り当てる名前を入力します。
- AAA クライアント IP アドレス — イーサネット管理ポートに使用する IP アドレス、または ACE 回線の IP アドレスを入力します（ACE が Cisco Secure ACS) との通信をどのように設定されているかによります)。
- キー — ACE と Cisco Secure ACS がトランザクションを認証するために使用する共有暗号鍵を入力します。共有暗号鍵は、Cisco Secure ACS と ACE の両方で同一のものを指定しなくてはなりません。キーは大文字と小文字を区別します。
- 使用する認証方式 — AAA プロトコルを選択します。RADIUS の場合は、AAA クライアントとの通信に使用するベンダーを選択します。

ステップ 4　[**Submit + Restart**] をクリックします。

Cisco Secure ACS は AAA クライアント エントリを保存してサービスを再起動します。この後、ACE からの RADIUS 要求に応じて処理を行うようになります。

RADIUS サーバ アカウンティング サービスの設定

Cisco Secure ACS で RADIUS のアカウンティング サービスを設定する手順は、次のとおりです。

ステップ 1　[**System Configuration**] > [**Logging**] > [**CSV RADIUS Accounting**] と選択します。CSV RADIUS Accounting File Configuration ページが表示されます。

ステップ 2　[**Log to CSV RADIUS Accounting report**] チェックボックスがオンになっていることを確認します。

ステップ 3　RADIUS アカウンティング ログに表示させたい RADIUS 属性が Logged Attributes リストに表示されていることを Select Columns To Log テーブルで確認します。標準の RADIUS 属性の他にも、Real Name、ExtDB Info、Logged Remotely など、Cisco Secure ACS が提供するいくつかの特別なログ属性が表示されます。これらの属性についての詳細は Cisco Secure ACS のユーザ マニュアルを参照してください。

ステップ 4　（任意）Cisco Secure ACS for Windows サーバを利用している場合は、ログ ファイル管理を選択できます。これは RADIUS アカウント ファイルの最大サイズ、保持する数、保持する期間、保存場所を決定します。

（注）　HTML インターフェイスの Network Configuration セクションで AAA サーバ エントリを設定すると、Cisco Secure ACS を利用してアカウンティング データを他の AAA サーバへ送ることができます。詳細については適当な Cisco Secure ACS ユーザ マニュアルを参照してください。

ステップ 5　属性を Logged Attributes へ移し終えたら [**Submit**] をクリックします。

RADIUS アカウンティングの設定に加えた変更が保存、実装されます。

RADIUS サーバの仮想化をサポートするプライベート属性の定義

複数のコンテキストにまたがる同一のユーザ名の作成と、コンテキスト内および複数ドメイン内での専用ロールへのユーザ名の関連付けを行えます。各コンテキストは RADIUS サーバを共有できますが、ユーザはコンテキストごとの認証と同一パスワードが必要です。

ユーザが ACE へのログインを試みると、認証を行うために ACE の RADIUS クライアントがリモートの RADIUS サーバにユーザ名とパスワードを送信します。RADIUS サーバは認証の一環としてユーザ プロファイルを取得します。ユーザが認証に成功すると、RADIUS サーバは ACE の RADIUS クライアントに認証ステータスとともにユーザ プロファイルを返します。ログインを試みているユーザに関連付けられたコンテキストが RADIUS サーバから取得したユーザ プロファイルのコンテキストに合致した場合は、RADIUS クライアントはリモート サーバから得たユーザ プロファイルに更新します。しかしコンテキストが合致しなかった場合は、ユーザ プロファイルはデフォルト ロール（Network-Monitor）とデフォルト ドメイン（default-domain）に更新されます。

RADIUS サーバ上のユーザ プロファイルは、ベンダー ID Cisco（09）および下位属性タイプ CiscoAVPair（タイプ 01）の ベンダー固有属性として、次の形式の文字列で設定します。

シェル :< コンテキスト名 >=< ロール > < ドメイン 1> < ドメイン 2>...< ドメイン N>

（注）　ユーザ プロファイル属性は RADIUS サーバ グループに対して重要な設定機能を提供します。認証実行中にユーザ プロファイルがサーバから取得できなかった場合、またはプロファイルをサーバから取得できてもプロファイル中のコンテキスト名（複数の場合もある）が、ユーザがログインを試みているコンテキストに合致しなかった場合は、認証に成功するとデフォルト ロール（Network-Monitor）とデフォルト ドメイン（default-domain）がそのユーザに割り当てられます。

Cisco Secure ACS で RADIUS のロール情報およびドメイン情報を設定する手順は、次のとおりです。

ステップ 1　Cisco Secure ACS HTML インターフェイスの User Setup セクションを開いて、仮想化に使用するユーザ プロファイル属性を定義したい既存のユーザ名をダブルクリックします。User Setup ページが表示されます。

ステップ 2　このページの Cisco IOS/PIX RADIUS Attributes セクションの下で、次の設定を行います。

- [**[009\001] cisco-av-pair**] チェックボックスをオンにします。
- 指定するコンテキストに用いるユーザ　ロールと関連ドメインを、次の形式に従って [[009\001] cisco-av-pair] チェックボックスの下のテキスト ボックスに入力します。

  ```
  shell:<contextname>=<role> <domain1> <domain2>...<domainN>
  ```

  たとえば、選択したユーザを C1 コンテキストとともにロール ROLE1 とドメイン DOMAIN1 に割り当てるには、**shell:C1=ROLE1 DOMAIN1** と入力します。

ステップ 3　RADIUS のロール設定とドメイン設定を終えたら [**Submit**] をクリックします。

たとえば、USER1 がロール ADMIN とドメイン MYDOMAIN1 に割り当てられている場合（shell:Admin=ADMIN MYDOMAIN1）は、次のいずれかが発生します。

- USER1 が Admin コンテキストからログインした場合は、そのユーザは自動的に Admin ロールと MyDomain1 ドメインに割り当てられます。
- USER1　が別のコンテキストからログインすると、そのユーザは自動的にデフォルト ロール（Network-Monitor）とデフォルト ドメイン（default-domain）に割り当てられます。この場合、認証の実行中にユーザ プロファイル属性が RADIUS サーバから取得されることはありません。

LDAP サーバの設定

この章は OpenLDAP サーバや Microsoft Active Director サーバなど、LDAP ディレクトリ サーバの設定方法を説明します。この章は、LDAP サーバと、LDAP クライアントとして機能する ACE が適切に通信を行うための手引きとして作成されています。

OpenLDAP ディレクトリ サーバを設定する手順は、次のとおりです。

ステップ 1　製品に添付のサンプル slapd.conf（通常は /usr/local/etc/openldap/slapd.conf のようにインストールされています）を編集して、BDB データベースの定義、スキーマ定義、rootDN、root パスワードを準備します。

ステップ 2　プライベート属性（コンテキスト ID とユーザ プロファイル）とプライベート オブジェクト クラスの定義を加えるためにプライベート スキーマを追加するか、既存のオブジェクト クラスを修正します。このスキーマを slapd.conf に追加します。

ステップ 3　LDAP サーバの slapd を起動します。

（注）　slapd　は多くの種類のプラットフォームで稼動しているスタンドアローン型の LDAP ディレクトリ サーバです。

ステップ 4　LDAP データベースを作成します。すなわち、データベースを格納する LDIF 形式のファイルを作成します。LDIF ファイル（example.ldif）が次の内容を含んでいることを確認します。

```
dn: dc=example,dc=com
objectClass: dcObject
objectClass:organization
dc: example
o: Example Corporation
description: The Example Corporation
dn: cn=Manager,dc=example,dc=com
```

objectclass: organizationalRole

cn: Manager

ステップ 5　**ldapadd** を実行して、これらのエントリを自分のディレクトリに挿入します。例を示します。

```
ldapadd -x -D "cn=Manager,dc=example,dc=com" -w secret -f
example.ldif
```

LDAP サーバで仮想化をサポートするためのプライベート属性の定義

ACE 上の LDAP クライアントには、LDAP サーバが管理するデータベース構造の詳細仕様に対応する機能はありません。そのかわり、データベース内のエントリが {userid, contextid} のペアにより重複なく識別され、このエントリにユーザ プロファイル属性が格納されているものと想定します。LDAP クライアントは、ACE 上で設定された検索フィルタを使用して、これら 2 つの属性を基に検索を実行します。LDAP サーバは、正しいユーザ エントリと、エントリの一部分であるユーザ プロファイル属性を見つけ出し、この情報を検索応答に含めて返します。

LDAP クライアントは仮想化を必須としないアプリケーションで運用できます。この場合、ユーザ エントリはユーザ名のみにより重複なく識別されます。検索フィルタは $userid 変数のみを含むように設定します（$contextid を含まない）。これら 2 つのプライベート属性は、ACE の CLI から **attribute user-profile** コマンドを入力して定義します（「LDAP サーバ グループへのユーザ プロファイル属性タイプの設定」の章を参照）。

ユーザ プロファイル属性の値は、次の形式で定義します。

シェル :< コンテキスト名 >=< ロール >< ドメイン 1>< ドメイン 2>...< ドメイン N>

（注） ユーザ プロファイル属性は LDAP サーバ グループに対して重要な設定機能を提供します。認証実行中にユーザ プロファイルがサーバから取得できなかった場合、またはプロファイルをサーバから取得できてもプロファイル中のコンテキスト名（複数の場合もある）が、ユーザがログインを試みているコンテキストに合致しなかった場合は、認証に成功するとデフォルト ロール（Network-Monitor）とデフォルト ドメイン（default-domain）がそのユーザに割り当てられます。

仮想化を要件としている場合は、LDAP サーバはスキーマに定義した contextid 属性が必要になります。ユーザごとに別のロールとドメインを割り当てる必要がある場合は、user-profile 属性（ロールとドメインが組み合わさった情報）が必要になります。slapd（LDAP ディレクトリ サーバ）が使用する attributetype ディレクティブの拡張方法については、LDAP クライアントのマニュアルを参照してください。

LDAP サーバで仮想化をサポートするためにプライベート属性を定義する手順は、次のとおりです。

ステップ 1　プライベート属性（コンテキスト ID とユーザ プロファイル）とプライベート オブジェクト クラスの定義を加えるためにプライベート スキーマを追加します。次に例を示します。

```
attributetype (2.5.4.55 NAME ( 'ctxid' 'contextid' )
         DESC 'virtual context name'
         SUP name )

attributetype ( 2.5.4.56 NAME ( 'usrprof' 'userprofile' )
         DESC 'user profile'
         SUP name )

objectclass ( 2.5.6.30 NAME 'ctxperson'
         DESC 'a person'
         SUP top STRUCTURAL
         MUST cn
         MAY  ( $ ctxid $ usrprof ) )
```

この例は任意に定義された OID を含んでいます。自分で定義する OID は、LDAP サーバ データベースに存在する他の既存 OID と重複しないようにしてください。

ステップ 2　このプライベート スキーマを設定に加えます（OpenLDAP の場合は sladp.conf）。

ステップ 3　コンテキスト ID とユーザ プロファイルが入るエントリを格納する LDAP データベースを、LDAP Data Interchange Format（LDIF）形式で定義します。LDIF フォーマットは RFC 2849 に定義されています。次に例を示します。

```
dn: ctxid=admin,cn=john,ou=employees,dc=example,dc=com
objectClass: ctxperson
ctxid: admin
cn: john
usrprof: shell:Admin=ROLE-1 DOMAIN-1
userPassword: xxxxxxxx
```

ステップ 4　LDAP サーバを起動します（OpenLDAP の場合は slapd）。

LDAP クライアントと LDAP サーバは次のように通信を開始します。

- LDAP クライアントがバインド要求を送信します。この要求送信には設定済みの rootDN となる DN と、サーバ グループに対して設定済みの root パスワードとなるパスワードを含んでいます。
- バインドに成功すると、LDAP クライアントは次の項目を含む検索要求を送信します。
  - baseDN — 設定済みの baseDN
  - スコープ — Subtree
  - 検索フィルタ — $userid と $contextid で構成される設定済みのフィルタ（それぞれ実際のユーザ名とコンテキスト名に置き換わる）
  - 属性 — userprofile に設定済みの属性タイプ
- 検索に成功すると、LDAP サーバは合致した DN とユーザ プロファイル属性値を検索結果から抽出します。合致した DN はそのユーザに対する DN となります。

- そのユーザとして再バインドします。DN にはユーザの DN を含み、パスワードにはユーザのパスワードを含むバインド要求を LDAP クライアントから送信させます。
- このバインドが成功すると、LDAP　サーバは認証成功メッセージを返します。このメッセージにはユーザ プロファイル属性値も含まれています。
- LDAP クライアントが LDAP サーバにアンバインド要求を送信します。

ユーザ アカウントの作成

仮想コンテキストに関連付けられたすべてのユーザのアカウント情報が ACE に保存されています。認証情報、ユーザ名、ユーザ パスワード、パスワード有効期限、所属ユーザ ロールは、すべて各ユーザのプロファイルの一部として保存されています。

ACE のグローバル管理者は各コンテキストの 1 ユーザをコンテキスト管理者に指定できます。コンテキスト管理者は自分のコンテキスト、または ACE のコンテキストのうち責任を持つコンテキストにログインし、別のユーザを追加作成できます。

新規ユーザは、ユーザ ロールを割り当てないとデフォルトのユーザ ロール Network-Monitor が割り当てられます。デフォルトでは、ユーザはすべてのドメインで操作を行うことができます。Admin コンテキストで作成したユーザの場合、アクセスのデフォルト スコープはデバイス全体となります。その他のコンテキストで作成したユーザは、コンテキスト全体がアクセスのデフォルト スコープとなります。ユーザのアクセスを制限するには、ロールとドメインの組み合わせを割り当てる必要があります。

ACE のコンテキストにユーザを割り当てる際は、次のことに注意してください。

- 複数のコンテキストにまたがって同一のユーザ名を作成できます。このユーザ名は 1 つのコンテキスト内の複数のドメイン内で、重複のない固有のロールに関連付けることができます。ユーザは 1 コンテキスト内で最大 10 ドメインまで重複のない固有のロールに関連付けることができます。
- 各コンテキストは RADIUS サーバを共有できますが、ユーザはコンテキストごとの認証と同一パスワードが必要です。
- ACE では、ログに記録されたユーザ アカウンティング アクティビティのすべてを、ユーザがサイン インしたコンテキストにより区別します。

CLI アクセスの認証に必要な ACE ローカル データベースへのアクセスを行うためのコンテキストとユーザ アカウントの作成の詳細情報は、『*Cisco Application Control Engine Module Virtualization Configuration Guide*』を参照してください。

RADIUS、TACACS+、または LDAP サーバのクライアントとしての ACE の設定

サーバとリモート認証プロトコルの識別方法として、1 つ以上の AAA サーバ グループを指定できます（RADIUS、TACACS+、または LDAP）。各サーバ グループには複数の AAA サーバ（同じサーバ タイプ）を設定できます。

各 AAA サーバに設定できるものは次のとおりです。

- サーバの IP アドレスとポート。
- ACE と AAA サーバ（RADIUS と TACACS+ のみ）の間の通信を認証するための暗号キー（共有暗号鍵）。
- ACE が認証要求をタイムアウトしたサーバに再送信する回数。この回数に達すると、その AAA サーバを無応答とみなして、グループ内の次の AAA サーバにコンタクトします（RADIUS と TACACS+ のみ）。
- ACE が認証要求に対するサーバからの応答を待つ時間間隔。これを越えると、そのサーバに新しい認証要求を再送信します。
- AAA サーバが稼働中で認証要求を受信できるかどうかを確認するために ACE が AAA サーバにプローブ パケットを送信する時間間隔。この回数だけ認証要求を送信してもサーバが応答しないときからデッドタイム間隔が開始します。
- TACACS+、RADIUS、または LDAP サーバ用に独立したサーバ グループ

このセクションの内容は、次のとおりです。

ACE での RADIUS の設定

ACE は、認証サービスとアカウンティングサービスを行うリモート RADIUS サーバと通信するため、RADIUS プロトコルをサポートしています。この章では、ACE を RADIUS サーバのクライアントとして運用するための設定方法を説明します。

このセクションの内容は、次のとおりです。

RADIUS サーバのパラメータの設定

radius-server host コマンドを使用すると、RADIUS サーバの IP アドレス、暗号キー、宛先 UDP ポート、その他のオプションを指定できます。**radius-server host** コマンドを複数使用して複数の RADIUS サーバを設定することもできます。

このコマンドの構文は次のとおりです。

radius-server host *ip_address* [**key** *shared_secret* [**0** *shared_secret* | **7** *shared_secret*]] [**auth-port** *port_number*] [**acct-port** *port_number*] [**authentication**] [**accounting**] [**timeout** *seconds*] [**retransmit** *count*]

引数、キーワード、オプションの内容は次のとおりです。

- *ip_address* — RADIUS サーバの IP アドレス。ドットで区切った 10 進数の形式で入力します（例、192.168.11.1）。
- **key** —（任意）ACE と RADIUS サーバ で実行されている RADIUS デーモンの間の通信に認証キーを利用できます。このキーは RADIUS サーバで使用される暗号キーと一致するテキスト文字列です。このキーは **radius-server key** コマンドのグローバル設定を書き換えます。キーを指定しないとグローバル値が使用されます。RADIUS のキーは永続性ストレージに必ず暗号化して保存されます。実行コンフィギュレーション画面においてもキーは暗号化した形式で表示されます。
- *shared_secret* — RADIUS クライアントとサーバの間の通信の認証に使用するキー。共有暗号鍵は RADIUS サーバで設定されたものと一致している必要があります。共有暗号鍵はスペースを入れず、大文字と小文字を区別して、63 文字までの文字列で入力してください。
- **0** —（任意）RADIUS クライアントとサーバの間の通信の認証を行うためのキーを、平文のテキスト（0 で指定）で設定します。

- **7 —**（任意）RADIUS クライアントとサーバの間の通信の認証を行うためのキーを、暗号化したテキスト（7 で指定）で設定します。
- **auth-port** *port_number* **—**（任意）RADIUS サーバへの認証要求の通信に使用する UDP 宛先ポートを指定します。デフォルトでは、RADIUS の認証用ポートは 1812 です（RFC 2138 と RFC 2139 の定義どおり）。利用する RADIUS サーバが 1812 以外のポートを使用する場合は、RADIUS サーバを起動する前に **auth-port** キーワードを使用して ACE に適切なポートを設定してください。引数 *port_number* は RADIUS のポート番号です。有効な値は 1 ～ 65535 です。
- **acct-port** *port_number* **—**（任意）RADIUS サーバへのアカウンティング要求の通信に使用する UDP 宛先ポートを指定します。デフォルトでは、RADIUS のアカウンティング用ポートは 1813 です（RFC 2138 と RFC 2139 の定義どおり）。利用する RADIUS サーバが 1813 以外のポートを使用する場合は、RADIUS サーバを起動する前に **acct-port** キーワードを使用して ACE に適切なポートを設定してください。引数 *port_number* は RADIUS のポート番号です。有効な値は 1 ～ 65535 です。
- **authentication —**（任意）RADIUS サーバを認証のみに利用する設定です。

（注） 認証かアカウンティングのいずれかのオプションを指定しないと、RADIUS サーバはアカウンティングと認証の両方に利用されます。

- **accounting —**（任意）RADIUS サーバをアカウンティングのみに利用する設定です。

（注） 認証かアカウンティングのいずれかのオプションを指定しないと、RADIUS サーバはアカウンティングと認証の両方に利用されます。

- **timeout** *seconds* **—**（任意）デフォルトでは、ACE は、認証要求に対する RADIUS サーバの応答を 1 秒だけ待ってから、認証要求を同じサーバに再送信します。認証リクエストに対する RADIUS サーバの応答を待って応答要求を再送信するまでの時間間隔を変更するには、**timeout** キーワードを使用します。有効な入力値は 1 ～ 60 秒で、デフォルトは 1 秒です。このコマンドは指定したサーバに対して、**radius-server timeout** コマンドを使って割り当てたグローバル設定を書き換えます。

- **retransmit** *count* **—** （任意）デフォルトでは ACE は、タイムアウトした RADIUS サーバに対して認証リクエストを 1 回だけ送信したあとに送信をやめ、次に見つかる AAA サーバにコンタクトします。retransmit オプションは、ACE が認証要求をタイムアウトしたサーバに再送信する回数です。この回数に達すると、そのサーバを無応答とみなして、グループ内の次のサーバにコンタクトします。**aaa authentication login** コマンドまたは **aaa accounting default** コマンドを使ってローカル データベースをローカル フォールバック手法として設定してある場合に、グループ内のすべてのサーバが認証とアカウンティングに利用できないと、ACE はそれらをローカル データベースで試みます。有効な入力値は 1 ～ 5 回で、デフォルトは 1 回です。このコマンドは指定したサーバに対して、**radius-server retransmit** コマンドを使って割り当てたグローバル設定を書き換えます。

たとえば、RADIUS サーバの認証パラメータを設定するには、次のように入力します。

```
host1/Admin(config)# radius-server host 192.168.2.3 key HostKey
host1/Admin(config)# radius-server host 192.168.2.3 key 7 secret_1256
host1/Admin(config)# radius-server host 192.168.2.3 auth-port 1645
host1/Admin(config)# radius-server host 192.168.2.3 acct-port 1646
host1/Admin(config)# radius-server host 192.168.2.3 authentication
host1/Admin(config)# radius-server host 192.168.2.3 accounting
host1/Admin(config)# radius-server host 192.168.2.3 timeout 25
host1/Admin(config)# radius-server host 192.168.2.3 retransmit 3
```

RADIUS サーバ認証の設定をデフォルトに戻すには、次のように入力します。

```
host1/Admin(config)# no radius-server host 192.168.2.3 acct-port 1646
```

RADIUS の NAS-IP-Address 属性の設定

一般的に、RADIUS サーバは RADIUS パケットの IP ヘッダ内に埋め込まれた送信元 IP アドレスを確認することにより、その RADIUS 要求の送信元を把握します。また、一部のサーバでは、RADIUS クライアントを識別するために RADIUS の NAS-IP-Address 属性が使われています。しかし、この方法では ACE 内部のプライベート ネットワーク インターフェイスの IP アドレスを外部にさらす可能があります。

デフォルトでは NAS-IP-Address は設定されていません。ACE は RADIUS サーバの IP アドレスに対してルート ルックアップを行い、その結果を利用します。RADIUS の NAS-IP-Address 属性を指定するには、**radius-server attribute nas-ipaddr** コマンドを使用します。この属性を利用すると、RADIUS 属性 4、すなわち各コンテキストに対応する NAS-IP-Address として使用する任意の IP アドレスを設定できます。**radius-server attribute nas-ipaddr** コマンドを利用すると、ACE が RADIUS サーバから見て単一の RADIUS クライアントとして動作します。設定された NAS-IP-Address は、送出されるすべての RADIUS 認証要求とアカウンティング パケットの中にカプセル化されます。

このコマンドの構文は次のとおりです。

radius-server attribute nas-ipaddr *nas_ip_address*

引数 *nas_ip_address* は、RADIUS の NAS-IP-Address、すなわち属性 4 として使用される IP アドレスを設定します。

たとえば、次のように入力すると RADIUS の NAS-IP-Address を指定できます。

```
host1/Admin(config)# radius-server attribute nas-ipaddr 192.168.1.1
```

RADIUS の NAS-IP-Address を削除してデフォルト設定に戻すには、次のように入力します。

```
host1/Admin(config)# no radius-server attribute nas-ipaddr 192.168.1.1
```

RADIUS サーバ事前共有キーのグローバルな設定

radius-server key コマンドを使用すると、ACE と各 RADIUS サーバで実行される RADIUS デーモン間の通信に使用する認証キーをグローバルに設定できます。このキーは RADIUS サーバで使用される暗号キーと一致するテキスト文字列です。RADIUS のキーは ACE の永続性ストレージに必ず暗号化して保存されます。このグローバルなキーは、指定サーバ グループ内にあり、**radius-server host** コマンドを使って共有暗号鍵を個々に設定されていない RADIUS サーバに適用されます。

このコマンドの構文は次のとおりです。

radius-server key {*shared_secret* | **0** *shared_secret* | **7** *shared_secret*}

引数とキーワードの内容は次のとおりです。

- *shared_secret* — RADIUS クライアントとサーバの間の通信の認証に使用するキー。共有暗号鍵は RADIUS サーバで設定されたものと一致している必要があります。共有暗号鍵はスペースを入れず、大文字と小文字を区別して、63 文字までの文字列で入力してください。
- **0** — RADIUS クライアントとサーバの間の通信の認証を行うためのキーを、平文のテキスト（0 で指定）で設定します。
- **7** — RADIUS クライアントとサーバの間の通信の認証を行うためのキーを、暗号化したテキスト（7 で指定）で設定します。

たとえば、RADIUS サーバに暗号化したテキスト（7 で指定）に組み込んで送る 1 つの認証キーをグローバルに設定するには、次のように入力します。

```
host1/Admin(config)# radius-server key 7 abe4DFeeweo00o
```

このキーを削除するには、次のように入力します。

```
host1/Admin(config)# no radius-server key 7 abe4DFeeweo00o
```

RADIUS サーバ デッドタイム間隔のグローバルな設定

デッドタイム間隔として設定された時間が経過する間、ACE は、その RADIUS サーバが利用可能で認証要求を受信できるかどうかを確認するために、プローブ アクセス要求パケットを送信します。**radius-server retransmit** コマンドまたは **radius-server host retransmit** コマンドのいずれかで設定された数の認証要求送信にサーバが応答しないと、デッドタイム間隔が始まります。サーバがプローブ アクセス要求パケットに応答すると、ACE は認証要求をそのサーバに送信します。

応答のないサーバが稼動しているかどうかを ACE が確認する時間間隔をグローバルに設定するには、**radius-server deadtime** コマンドを使用します。

このコマンドを使用すると、ACE は認証要求に応答しなかったすべての RADIUS サーバを非稼動（dead）とマーク付けします。このアクションにより、要求がタイムアウトするまで待たずとも、次に設定されているサーバを試行することができます。ACE は引数 *minutes* で指定した時間の経過中に要求を追加送信し、非稼動のマークが付いた RADIUS サーバを無視します。

このコマンドの構文は次のとおりです。

radius-server deadtime *minutes*

引数 *minutes* は、トランザクション要求に無応答の RADIUS サーバを ACE が無視する分単位の長さです。有効な入力値は 0 ～ 1440 分（24 時間）で、デフォルトは 0 です。

たとえば、認証要求に応答しない RADIUS サーバに 15 分間のデッドタイム間隔をグローバルに設定するには、次のように入力します。

```
host1/Admin(config)# radius-server deadtime 15
```

RADIUS サーバのデッドタイム間隔を 0 に設定するには、次のように入力します。

```
host1/Admin(config)# no radius-server deadtime 15
```

RADIUS サーバへの再送信回数のグローバルな設定

デフォルトでは、ACE は RADIUS サーバへの認証要求を 1 回送信したあとでそのサーバを無応答と判断し、グループ内の次のサーバにコンタクトします。ACE が RADIUS サーバに認証要求を送信する回数をグローバルに変更するには、**radius-server retransmit** コマンドを使用します。**aaa authentication login** コマンドまたは **aaa accounting default** コマンドを使ってローカル データベースをローカル フォールバック手法として設定してある場合に、グループ内のすべてのサーバが認証とアカウンティングに利用できないと、ACE はそれらをローカル データベースで試みます。フォールバック手段が存在しない場合、ACE はそのサーバ グループ内の AAA サーバの 1 つへのコンタクトを続けます。

ACE はこのグローバルな再送信回数の値を、**radius-server host** コマンドを使って個々に設定されていない RADIUS サーバに適用します。

このコマンドの構文は次のとおりです。

radius-server retransmit *count*

引数 *count* は、次に用意されているサーバに ACE がコンタクトを試みる前に RADIUS サーバへの接続を試みる回数です。値の範囲は 1 ～ 5 回で、デフォルトは 1 です。

たとえば、再送信の回数をグローバルに 3 に設定するには、次のように入力します。

```
host1/Admin(config)# radius-server retransmit 3
```

送信試行回数をデフォルトの 1 に戻すには、次のように入力します。

```
host1/Admin(config)# no radius-server retransmit 3
```

RADIUS サーバ タイムアウト値のグローバルな設定

デフォルトでは、ACE は無応答サーバへの認証要求に RADIUS サーバが応答するまで 1 秒待ってから認証要求をそのサーバへ再送信します。RADIUS サーバに認証要求を再送信する前に ACE が RADIUS サーバの応答を待つ時間間隔をグローバルに変更するには、**radius-server timeout** コマンドを使用します。ACE はこのグローバルなタイムアウト値を、**radius-server host** コマンドを使って個々に値を設定されていない RADIUS サーバに適用します。

このコマンドの構文は次のとおりです。

radius-server timeout *seconds*

引数 *seconds* は RADIUS サーバへ再送信を繰り返す間隔の秒数です。有効な入力値は 1 ～ 60 秒で、デフォルトは 1 秒です。

たとえば、タイムアウト値をグローバルに 30 秒に設定するには、次のように入力します。

```
host1/Admin(config)# radius-server timeout 30
```

送信試行間隔をデフォルトの 1 秒に戻すには、次のように入力します。

```
host1/Admin(config)# no radius-server timeout 30
```

ACE での TACACS+ の設定

ACE は、認証サービスとアカウンティングサービスを行う TACACS+ サーバと通信するための TACACS+ プロトコルをサポートしています。この章では、ACE を TACACS+ サーバのクライアントとして運用するための設定方法説明します。

このセクションの内容は、次のとおりです。

TACACS+ サーバのパラメータの設定

tacacs-server host コマンドを利用すると、TACACS+ サーバの IP アドレス、暗号キー、宛先 ポート、その他のオプションを指定できます。**tacacs-server host** コマンドを複数回使用して複数の TACACS+ サーバを設定することもできます。

このコマンドの構文は次のとおりです。

tacacs-server host *ip_address* [**key** *shared_secret* [**0** *shared_secret* | **7** *shared_secret*]] [**port** *port_number*] [**timeout** *seconds*]

引数、キーワード、オプションの内容は次のとおりです。

- *ip_address* — TACACS+ サーバの IP アドレス。ドットで区切った 10 進数の形式で入力します（例、192.168.11.1）。
- **key** — （任意）ACE と TACACS+ サーバ で実行されているデーモンの間の通信に認証キーを利用できます。このキーは TACACS+ サーバで使用される暗号キーと一致するテキスト文字列です。このキーは **tacacs-server key** コマンドのグローバル設定を書き換えます。キーを指定しないとグローバル値が使用されます。TACACS+ のキーは永続性ストレージに必ず暗号化して保存されます。実行中の設定もキーを暗号化した形式で表示します。
- *shared_secret* — TACACS+ クライアントとサーバの間の通信の認証に使用するキー共有暗号鍵は TACACS+ サーバで設定されたものと一致している必要があります。共有暗号鍵はスペースを入れず、大文字と小文字を区別して、63 文字までの文字列で入力してください。

- **0 —**（任意）TACACS+ クライアントとサーバの間の通信の認証を行うためのキーを、平文のテキスト（0 で指定）で設定します。
- **7 —**（任意）TACACS+ クライアントとサーバの間の通信の認証を行うためのキーを、暗号化したテキスト（7 で指定）で設定します。
- **port** *port_number* **—** TACACS+ サーバへの認証要求の通信に使用する TCP 宛先ポートを指定します。デフォルトでは、TACACS+ の認証用ポートは 49 です（RFC 1492 の定義どおり）。利用する TACACS+ サーバが 49 以外のポートを使用する場合は、TACACS+ サーバを起動する前に **port** キーワードを使用して、ACE に適切なポートを設定してください。引数 *port_number* は TACACS+ のポート番号を指定します。有効な値は 1 ～ 65535 です。
- **timeout** *seconds* **—**（任意）デフォルトでは、ACE は、認証要求に対する TACACS+ サーバの応答を 1 秒だけ待ってからタイムアウト発生を宣言し、グループ内の次のサーバにコンタクトを試みます。**aaa authentication login** コマンドまたは **aaa accounting default** コマンドを使ってローカル データベースをローカル フォールバック手法として設定してある場合に、グループ内のすべてのサーバが認証とアカウンティングに利用できないと、ACE はそれらをローカル データベースで試みます。認証リクエストに対する TACACS+ サーバの応答を待つ時間間隔を変更するには **timeout** キーワードを使用します。有効な入力値は 1 ～ 60 秒で、デフォルトは 1 秒です。このコマンドは指定したサーバに対して、**tacacs-server timeout** コマンドを使って割り当てたグローバル設定を書き換えます。

たとえば、TACACS+ サーバの認証パラメータを設定するには、次のように入力します。

```
host1/Admin(config)# tacacs-server host 192.168.3.2 key HostKey
host1/Admin(config)# tacacs-server host 192.168.3.2 tacacs3 key 7 1234
host1/Admin(config)# tacacs-server host 192.168.3.2 port 1645
host1/Admin(config)# tacacs-server host 192.168.3.2 timeout 5
```

TACACS+ サーバ 認証の設定をデフォルトに戻すには、次のように入力します。

```
host1/Admin(config)# no tacacs-server host tacacs3 key 7 1234
```

事前共有キーのグローバルな設定

tacacs-server key コマンドを使用すると、ACE と各 TACACS+ サーバで実行される TACACS+ デーモン間の通信に使用する認証キーをグローバルに設定できます。このキーは TACACS+ サーバで使用される暗号キーと一致するテキスト文字列です。TACACS+ のキーは ACE の永続性ストレージに必ず暗号化して保存されます。このグローバルなキーは、指定サーバ グループ内の **tacacs-server host** コマンドを使って共有暗号鍵を個々に設定されていない TACACS+ サーバに適用されます。

このコマンドの構文は次のとおりです。

tacacs-server key *shared_secret* **|** **0** *shared_secret* | **7** *shared_secret*

引数とキーワードの内容は次のとおりです。

- *shared_secret* **—** TACACS+ クライアントとサーバの間の通信の認証に使用するキー共有暗号鍵は TACACS+ サーバで設定されたものと一致している必要があります。共有暗号鍵はスペースを入れず、大文字と小文字を区別して、63 文字までの文字列で入力してください。
- **0 —** TACACS+ クライアントとサーバの間の通信の認証を行うためのキーを、平文のテキスト（0 で指定）で設定します。
- **7 —** TACACS+ クライアントとサーバの間の通信の認証を行うためのキーを、暗号化したテキスト（7 で指定）で設定します。

たとえば、TACACS+ クライアントとサーバの間の通信の認証を行うためのキーを、暗号化したテキスト（7 で指定）でグローバルに設定するには、次のように入力します。

```
host1/Admin(config)# tacacs-server key 7 abe4DFeeweo00o
```

このキーを削除するには、次のように入力します。

```
host1/Admin(config)# no tacacs-server key 7 abe4DFeeweo00o
```

TACACS+ サーバ デッドタイム間隔のグローバルな設定

デッドタイム間隔として設定された時間が経過する間、ACE は、その TACACS+ サーバが利用可能で認証要求を受信できるかどうかを確認するために、プローブ アクセス要求パケットを送信します。デッドタイム間隔は、認証要求を 1 回送信してサーバが応答しないときから開始します。サーバがプローブ アクセス要求パケットに応答すると、ACE は認証要求をそのサーバに再送信します。

応答のないサーバが稼動しているかどうかを ACE が確認する時間間隔をグローバルに設定するには、**tacacs-server deadtime** コマンドを使用します。

このコマンドを使用すると、ACE　は認証要求に応答しなかったすべての TACACS+ サーバを非稼動（dead）とマーク付けします。このアクションにより、要求がタイムアウトするまで待たずとも、次に設定されているサーバを試行することができます。ACE は引数 *minutes* が経過する間要求を追加送信することにより、非稼動のマークが付いた TACACS+ サーバを無視します。

このコマンドの構文は次のとおりです。

tacacs-server deadtime *minutes*

引数 *minutes* は、トランザクション要求に無応答の TACACS+ サーバを ACE が無視する分単位の長さです。有効な入力値は 0 ～ 1440 分（24 時間）で、デフォルトは 0 です。

たとえば、認証要求に応答しない TACACS+ サーバに 15 分間のデッドタイム間隔をグローバルに設定するには、次のように入力します。

```
host1/Admin(config)# tacacs-server deadtime 15
```

TACACS+ サーバのデッドタイム間隔を 0 に設定するには、次のように入力します。

```
host1/Admin(config)# no tacacs-server deadtime 15
```

TACACS+ サーバ タイムアウト値のグローバルな設定

デフォルトでは、ACE は、認証要求に対する TACACS+ サーバからの応答の受信を 1 秒だけ待ってから、タイムアウト発生を宣言し、グループ内の次のサーバにコンタクトを試みます。TACACS+ サーバに認証要求を再送信する前に ACE が TACACS+ サーバの応答を待つ時間間隔をグローバルに変更するには、**tacacs-server timeout** コマンドを使用します。ACE はこのグローバルなタイムアウト値を、**tacacs-server host** コマンドを使って個々に設定されていない TACACS+ サーバに適用します。

このコマンドの構文は次のとおりです。

tacacs-server timeout *seconds*

引数 *seconds* は秒で表したタイムアウト値です。有効な入力値は 1 ～ 60 秒で、デフォルトは 1 秒です。

たとえば、タイムアウト値をグローバルに 30 秒に設定するには、次のように入力します。

```
host1/Admin(config)# tacacs-server timeout 30
```

送信試行間隔をデフォルトの 1 秒に戻すには、次のように入力します。

```
host1/Admin(config)# no tacacs-server timeout 30
```

ACE での LDAP の設定

ACE は、認証サービスを行うリモート LDAP ディレクトリ サーバと通信するため、LDAP プロトコルをサポートしています。この章では、ACE を LDAP サーバのクライアントとして運用するための設定方法を説明します。

このセクションの内容は、次のとおりです。

- LDAP サーバのパラメータの設定
- LDAP サーバ ポートのグローバルな設定
- LDAP サーバ タイムアウト値のグローバルな設定

LDAP サーバのパラメータの設定

ldap-server host コマンドを利用すると、LDAP サーバのホスト名または IP アドレス、宛先 ポート、その他のオプションを指定できます。**ldaps-server host** コマンドを複数使用して複数の LDAP サーバを設定することもできます。

このコマンドの構文は次のとおりです。

ldap-server host *ip_address* [**port** *port_number*] [**timeout** *seconds*] [**rootDN** *"DN_string"* [**password** *bind_password*]]

引数、キーワード、オプションの内容は次のとおりです。

- *ip_address* **—** LDAP サーバの IP アドレス。ドットで区切った 10 進数の形式で入力します（例、192.168.11.1）。
- **port** *port_number* **—**（任意）LDAP ディレクトリ サーバへの認証要求の通信に使用する TCP 宛先ポートを指定します。デフォルトでは LDAP サーバのポートは 389 です。389 以外のポートを使用する場合は、LDAP サービスを起動する前に **port** キーワードを使用して ACE 設定してください。引数 *port_number* は LDAP のポート番号です。効な値は 1 ～ 65535 です。このコマンドは指定したサーバに対して、**ldap-server port** コマンドを使って割り当てたグローバル設定を書き換えます。
- **timeout** *seconds* **—**（任意）ACE が LDAP サーバのタイムアウト発生を宣言する前に LDAP サーバからの応答を待つ秒数を指定します。デフォルトでは、ACE は認証要求に対する LDAP サーバの応答を 5 秒だけ待ってから、タイムアウト発生を宣言し、グループ内の次のサーバにコンタクトを試みます。認証リクエストに対する LDAP サーバの応答を待つ時間間隔を変更するには、**timeout** キーワードを使用します。有効な入力値は 1 ～ 60 秒で、デフォルトは 5 秒です。このコマンドは指定したサーバに対して、**ldap-server timeout** コマンドを使って割り当てたグローバル設定を書き換えます。
- **rootDN** *"DN_string"* **—**（任意）LDAP サーバ ディレクトリへの操作に対して、アクセス コントロールまたは管理者による制限のパラメータにより規制されないユーザの DN（識別名）を定義します。rootDN ユーザは LDAP サーバ データベースに対するルート ユーザになります。63 文字までの英数字を「"」記号で囲んで入力します。デフォルトでは何も入っていません。
- **password** *bind_password* **—**（任意）LDAP サーバ ディレクトリの rootDN に適用するバインド パスワード（rootpw）を定義します。63 文字までの英数字を、「"」記号で囲まずに入力します。デフォルトでは何も入っていません。

たとえば、LDAP サーバの認証パラメータを設定するには、次のように入力します。

```
host1/Admin(config)# ldap-server host 192.168.2.3 port 2003
host1/Admin(config)# ldap-server host 192.168.2.3 timeout 60
host1/Admin(config)# ldap-server host 192.168.2.3 rootDN
"cn=manager,dc=cisco,dc=com" password lab
```

LDAP サーバの認証設定を削除するには、次のように入力します。

```
host1/Admin(config)# no ldap-server host 192.168.2.3
```

LDAP サーバ ポートのグローバルな設定

デフォルトでは、LDAP ディレクトリ サーバへの認証要求を通信するための TCP 宛先ポートは 389 です。利用する LDAP サーバが 389 以外のポートを使用する場合は、LDAP サービスを起動する前に **ldap-server port** コマンドを使用して、ACE にグローバルに設定してください。このグローバルなポート設定は、**ldap-server host** コマンドを使用して TCP ポート値を個々に設定されていない LDAP サーバに適用されます。

このコマンドの構文は次のとおりです。

ldap-server port *port_number*

引数 *port_number* は LDAP サーバへの宛先ポートです。有効な値は 1 ～ 65535 で、デフォルトは TCP ポート 389 です。

たとえば、TCP ポートをグローバルに設定するには、次のように入力します。

```
host1/Admin(config)# ldap-server port 2003
```

デフォルトの TCP ポート 389 に戻すには、次のように入力します。

```
host1/Admin(config)# no ldap-server port 2003
```

LDAP サーバ タイムアウト値のグローバルな設定

デフォルトでは、ACE は、認証要求に対する LDAP サーバからの応答の受信を 5 秒だけ待ってから、タイムアウト発生を宣言し、グループ内の次のサーバにコンタクトを試みます。ACE がタイムアウト発生を宣言する前に LDAP サーバが応答に返信するまで待つ時間間隔をグローバルに変更するには、**ldap-server timeout** コマンドを使用します。ACE はこのグローバルなタイムアウト値を、**ldap-server host** コマンドを使って個々に値を設定されていない LDAP サーバに適用します。

このコマンドの構文は次のとおりです。

ldap-server timeout *seconds*

引数 *seconds* は秒で表したタイムアウト値です。有効な入力値は 1 ～ 60 秒で、デフォルトは 5 秒です。

たとえば、タイムアウト値をグローバルに 30 秒に設定するには、次のように入力します。

```
host1/Admin(config)# ldap-server timeout 30
```

送信試行間隔をデフォルトの 5 秒に変更するには、次のように入力します。

```
host1/Admin(config)# no ldap-server timeout 30
```

AAA サーバ グループの設定

このセクションの内容は、次のとおりです。

TACACS+、RADIUS、LDAP サーバ グループの設定

サーバ グループとは、特定種類のサーバ ホストを集めた名簿のようなものです。ACE は複数の TACACS+ サーバ、RADIUS サーバ、および LDAP サーバを名称付きサーバ グループとして設定できます。AAA サーバ ホストを種類別のリストにグループ化します。ACE がサーバ ホストを検索する際は、それぞれのホストがグループに指定された順番で行います。

TACACS+、RADIUS、LDAP の各サーバを独立したサーバ グループに設定するには、**aaa group server** コマンドを使用します。サーバ グループの設定はいつでも行うことができますが、**aaa authentication login** コマンド、または **aaa accounting default** コマンドを使用して各 AAA サービスに適用しなければ有効になりません。

ACE では各コンテキストあたり、最大 100 サーバ グループを設定できます。

ACE はユーザ認証およびアカウンティングを行う際に、サーバ グループの最初に並ぶサーバにコンタクトを試みます。そのサーバが利用できない場合は、グループ内で次に並ぶ設定サーバにコンタクトを試みます。そのグループのすべてのサーバが利用できなかったときは、次に設定されているサーバ グループのサーバを試します。ACE は、認証要求がいずれかの AAA サーバにより受け入れられるまで、このプロセスを繰り返します。あるサーバ グループ内で指定した AAA サーバが利用できず、かつフォールバック手段としてローカル認証を指定してある場合は（**aaa authentication login** コマンド で指定）、ACE は ACE 上のローカル データベースに対してユーザ認証を試みます。フォールバック手段が存在しない場合、ACE はそのサーバ グループ内の AAA サーバの 1 つへのコンタクトを続けます。

このコマンドの構文は次のとおりです。

aaa group server {**ldap** | **radius** | **tacacs+**} *group_name*

引数とキーワードの内容は次のとおりです。

- **ldap —** LDAP ディレクトリ サーバ グループを 1 つ指定します。
- **radius —** RADIUS サーバ グループを 1 つ指定します。
- **tacacs+ —** TACACS+ サーバ グループを 1 つ指定します。
- *group_name* **—** サーバのグループ。サーバ グループ名は 64 文字までの英数字です。

CLI は TACACS+、RADIUS、または LDAP サーバの設定モードを表示します。これまでに設定した 1 つまたは複数のサーバの名前を確認できます。このサーバをサーバ グループに追加できます。

このサーバ設定モード コマンドの構文は次のとおりです。

server *ip_address*

引数 *ip_address* は、サーバ グループに追加できる既存の RADIUS、TACACS+、または LDAP サーバの IP アドレスです。ドットで区切った 10 進数の形式で入力します（例、192.168.11.1）。サーバ設定モードで **server** コマンドを複数入力すると、複数のサーバをサーバ グループに追加できます。複数のサーバ グループに同一のサーバを所属させることもできます。

たとえば、RADIUS サーバ グループを作成するには、次のように入力します。

```
host1/Admin(config)# aaa group server radius RAD_Server_Group1
host1/Admin(config-radius)# server 192.168.252.1
host1/Admin(config-radius)# server 192.168.252.2
host1/Admin(config-radius)# server 192.168.252.3
```

サーバ グループからサーバを 1 つ削除するには、次のように入力します。

```
host1/Admin(config-radius)# no server 192.168.252.3
```

サーバ グループを 1 つ削除するには、次のように入力します。

```
host1/Admin(config)# no aaa group server radius RAD_Server_Group1
```

また、TACACS+、RADIUS、および LDAP の各サーバ グループに対して、次のパラメータも設定できます。

- TACACS+ サーバ グループには、そのサーバ グループに対するデッドタイム間隔を指定できます。「TACACS+ サーバ グループへのデッドタイム間隔の設定」を参照してください。
- RADIUS サーバ グループには、そのサーバ グループに対するデッドタイム間隔を指定できます。「RADIUS サーバ グループへのデッドタイム間隔の設定」を参照してください。
- LDAP サーバ グループには、次のパラメータを指定できます。
  - ユーザ プロファイル属性 — 「LDAP サーバ グループへのユーザ プロファイル属性タイプの設定」の章を参照してください。

- ベース DN — 「LDAP サーバ グループへのベース DN の設定」の章を参照してください。
- LDAP 検索フィルタ — 「LDAP サーバ グループへの検索フィルタの設定」の章を参照してください。

TACACS+ サーバ グループへのデッドタイム間隔の設定

TACACS+ サーバ グループには、そのサーバ グループに対するデッドタイム間隔を指定できます。デッドタイム間隔として設定された時間が経過する間、ACE は、その TACACS+ サーバが利用可能で認証要求を受信できるかどうかを確認するために、プローブ アクセス要求パケットを送信します。デッドタイム間隔は、認証要求を 1 回送信してサーバが応答しないときから開始します。サーバがプローブ アクセス要求パケットに応答すると、ACE は認証要求をそのサーバに再送信します。

応答のないサーバ グループが稼動しているかどうかを ACE が確認する時間間隔をグローバルに設定するには、**deadtime** コマンドを使用します。

このコマンドを使用すると、ACE　は認証要求に応答しなかったすべての TACACS+ サーバを非稼動（dead）とマーク付けします。このアクションにより、要求がタイムアウトするまで待たずとも、次に設定されているサーバを試行することができます。ACE は引数 *minutes* が経過する間要求を追加送信することにより、非稼動のマークが付いた TACACS+ サーバを無視します。

このコマンドの構文は次のとおりです。

deadtime *minutes*

引数 *minutes* は、トランザクション要求に無応答の TACACS+ サーバを ACE が無視する分単位の長さです。有効な入力値は 0 ～ 1440 分（24 時間）で、デフォルトは 0 です。

たとえば、認証要求に応答しない TACACS+ サーバに 15 分間のデッドタイム間隔をグローバルに設定するには、次のように入力します。

```
host1/Admin(config-tacacs)# deadtime 15
```

RADIUS サーバのデッドタイム間隔を 0 にリセットするには、次のように入力します。

```
host1/Admin(config-tacacs)# no deadtime 15
```

RADIUS サーバ グループへのデッドタイム間隔の設定

RADIUS サーバ グループには、そのサーバ グループに対するデッドタイム間隔を指定できます。デッドタイム間隔として設定された時間が経過する間、ACE は、その RADIUS サーバが利用可能で認証要求を受信できるかどうかを確認するために、プローブ アクセス要求パケットを送信します。認証要求を 1 回送信してサーバが応答しないときからデッドタイム間隔が開始します。サーバがプローブ アクセス要求パケットに応答すると、ACE は認証要求をそのサーバに再送信します。

応答のないサーバ グループが稼動しているかどうかを ACE が確認する時間間隔をグローバルに設定するには、**deadtime** コマンドを使用します。

このコマンドを使用すると、ACE は認証要求に応答しなかったすべての RADIUS サーバを非稼動（dead）とマーク付けします。このアクションにより、要求がタイムアウトするまで待たずとも、次に設定されているサーバを試行することができます。ACE は引数 *minutes* が経過する間要求を追加送信し、非稼動のマークが付いた RADIUS サーバを無視します。

このコマンドの構文は次のとおりです。

deadtime *minutes*

引数 *minutes* は、トランザクション要求に無応答の RADIUS サーバを ACE が無視する分単位の長さです。有効な入力値は 0 ～ 1440 分（24 時間）で、デフォルトは 0 です。

たとえば、認証要求に応答しない RADIUS サーバに 15 分間のデッドタイム間隔をグローバルに設定するには、次のように入力します。

```
host1/Admin(config-radius)# deadtime 15
```

RADIUS サーバのデッドタイム間隔を 0 にリセットするには、次のように入力します。

```
host1/Admin(config-radius)# no deadtime 15
```

LDAP サーバ グループへのユーザ プロファイル属性タイプの設定

LDAP サーバは検索要求の一環としてユーザ プロファイルを取得します。検索要求を行う際は、LDAP サーバからユーザ プロファイル属性を取得するために、LDAP クライアントは検索要求の中にこの属性タイプ（指定された文字列）を含めます。LDAP サーバがユーザ プロファイル属性を適切に識別できるように、検索要求は LDAP サーバが使用する属性タイプと一致（LDAP サーバのプライベート スキーマで定義されたとおり）していなければなりません。LDAP サーバは、検索フィルタを使用して自分のデータベースからユーザ プロファイル エントリを見つけます。該当のエントリが見つかると、LDAP サーバはエントリに保存されていたユーザ プロファイル属性を格納した検索応答を返信します。この値の中には、そのユーザのそのコンテキストにおけるロールとドメインのペアが含まれています。

ユーザ プロファイル属性の値は、次の形式で定義します。

シェル :< コンテキスト名 >=< ロール > < ドメイン 1> < ドメイン 2>...< ドメイン N>

（注） ユーザ プロファイル属性は LDAP サーバ グループに対して重要な設定機能を提供します。認証実行中にユーザ プロファイルがサーバから取得できなかった場合、またはプロファイルをサーバから取得できてもプロファイル中のコンテキスト名（複数の場合もある）が、ユーザがログインを試みているコンテキストに合致しなかった場合は、認証に成功するとデフォルト ロール（Network-Monitor）とデフォルト ドメイン（default-domain）がそのユーザに割り当てられます。

この属性タイプはユーザ プロファイル属性に用いられます。この属性はプライベートであるため、LDAP サーバ データベースはユーザ プロファイルに対して同一の属性タイプを使用しなければなりません。LDAP クライアント（ACE）は、ダウンロードしたい属性としてこの属性タイプを含む検索要求を送信します。参照が成功すると、検索応答にこの属性値が格納されます。この属性値には、この特定コンテキストにおけるユーザ ロールとドメインのペアに相当する文字列が含まれていなければなりません。

LDAP サーバに使用させるユーザ プロファイル属性を指定するには **attribute user-profile** コマンドを使用します。

LDAP サーバ グループには LDAP ユーザ プロファイル属性をサブ設定レベルで設定できます（「AAA サーバ グループの設定」の章で説明する方法で作成）。

このコマンドの構文は次のとおりです。

attribute user-profile *text*

引数 *text* はユーザ プロファイルです。ユーザ プロファイルは 63 文字までの英数字で、スペースを入れず、「"」記号で囲まずに入力します。

たとえば、LDAP ユーザ プロファイル属性を設定するには、次のように入力します。

```
host1/Admin(config-ldap)# attribute user-profile usrprof
```

ユーザ プロファイル属性を削除するには、次のように入力します。

```
host1/Admin(config-ldap)# no attribute user-profile usrprof
```

LDAP サーバ グループへのベース DN の設定

LDAP サーバ グループの作成の際は、LDAP ディレクトリ ツリーの最上層が基点となります。これはベース DN と呼ばれます。ベース DN は LDAP サーバ ディレクトリでの検索を行うために利用するものです。ベース DN は “dc=your,dc=domain” のような形式で表現できます。ベース DN は DNS ドメイン名を基点として利用し、いくつかのドメイン コンポーネントに分かれます。LDAP ディレクトリ ツリーへの検索に使用するベース DN を設定するには、base-DN server group コマンドを使用します。

（注） LDAP サーバ グループにはベース DN を必ず設定しなくてはなりません。これを設定しないと、ユーザの認証が行われません。

ベース DN はサブモードで LDAP サーバ グループ（「AAA サーバ グループの設定」の章で説明する方法で作成）に設定できます。

このコマンドの構文は次のとおりです。

base-DN *text*

引数 *text* は検索ベースを識別する名前です。ベース DN は 63 文字までの英数字で、スペースを入れず、「"」記号で囲んで入力します。

たとえば、ベース DN を設定するには、次のように入力します。

```
host1/Admin(config)# aaa group server ldap LDAP_Server_Group1
host1/Admin(config-ldap)# base-DN "dc=sns,dc=cisco,dc=com"
```

設定したベース DN を削除するには、次のように入力します。

```
host1/Admin(config-ldap)# no base-DN "dc=sns,dc=cisco,dc=com"
```

LDAP サーバ グループへの検索フィルタの設定

LDAP サーバ グループに対しては、ACE はデータベースでユーザを参照するために検索フィルタを LDAP サーバに送信します。検索フィルタを利用すると、検索の比較基準を定義できるため、より効率的で効果的な検索が可能になります。検索フィルタは LDAP クライアントがサーバに送信する検索要求の中に使用され、DIT 中から当該ユーザのノードを見つけ出します。使用する的確なフィルタを設定するには **filter search-user** コマンドを使用します。$user と $contextid は要求を送信する際に実際の値に置き換わります。

検索フィルタは RFC 2254 に定義された形式に従う必要があります。LDAP クライアントは、設定された $userid と $contextid をクライアントが認証しようとしているユーザ ID と関連付けられた仮想コンテキスト名に置き換えたあと、検索フィルタを格納した検索要求を送信します。ACE では $userid と $contextid をユーザ ID とコンテキスト名のプレースホルダーとして利用できます。

（注） LDAP サーバ グループには検索フィルタを必ず設定しなくてはなりません。これを設定しないと、ユーザの認証が行われません。

LDAP サーバ グループには LDAP 検索フィルタをサブ設定レベルで設定します（「AAA サーバ グループの設定」の章で説明する方法で作成）。

このコマンドの構文は次のとおりです。

filter search-user *text*

引数 *text* は検索フィルタです。検索フィルタは 63 文字までの英数字で、スペースを入れず、「"」記号で囲んで入力します。

たとえば、検索フィルタを設定するには、次のように入力します。

```
host1/Admin(config)# aaa group server ldap LDAP_Server_Group1
host1/Admin(config-ldap)# filter search-user "(&(objectclass=person)
(&(cn=$userid)(cid=$contextid)))"
```

この検索フィルタを削除するには、次のように入力します。

```
host1/Admin(config-ldap)# no filter search-user
"(&(objectclass=person)(&(cn=$userid)(cid=$contextid)))"
```

ログイン認証手段の定義

認証とは、ACE にコンソール ポートや、Telnet または SSH のセッションから CLI でログインを試みている人物の身元を確認するプロセスです。この身元確認は、ACE へのアクセスを試みている人物が提供するユーザ名とパスワードの組み合わせに基づいています。

ACE は ACE に保存されている参照用データベースを利用したローカル認証、または 1 つ以上の TACACS+、RADIUS、LDAP の各サーバを利用したリモート認証をサポートします。設定した AAA サーバが認証要求に応答しなかった場合のフォールバック認証手段として、ACE 上のローカル データベースを指定できます。

ユーザ認証に使用するデフォルトのログイン手段は Telnet セッションまたは SSH セッションです。デフォルトの認証手段を書き換えて、コンソール ポートによる認証に指定できます。

ACE CLI へのログインに使用する認証手段を設定するには、**aaa authentication login** コマンドを設定モードで使用します。

このコマンドの構文は次のとおりです。

aaa authentication login {{**console** | **default**} {{**group** *group_name*} {**local**} {**none**}}} | **error-enable**

引数、キーワード、オプションの内容は次のとおりです。

- **console —** コンソール ポートを使用するログイン認証を指定します。サーバ グループごとに設定されます。
- **default —** デフォルトの認証手段を指定します（Telnet または SSH でのログイン）。サーバ グループごとに設定されます。
- **group** *group_name* **—** ログイン認証プロセスを、**aaa group server** コマンドで定義された TACACS+、RADIUS、または LDAP サーバに関連付けます。サーバ グループ名は 64 文字までの英数字です。
- **local —** ログイン認証手段として、ACE 上のローカル データベースを指定します。サーバが応答しない場合は、フォールバック認証手段としてローカル データベースが使用されます。

- **none —** ACE がパスワード確認を行わないように指定します。このオプションを設定すると、ユーザは有効なパスワードを提供しなくても ACE にログインできます。**none** オプションの指定を許されているのは、Admin ロールのユーザだけです。

注意　このオプションの使用には注意を払ってください。**none** を指定すると、ユーザは誰でも自由に ACE にアクセスできるようになります。

- **error-enable —** リモート AAA サーバが応答しない場合にログイン エラー メッセージの表示を可能にします。現在の表示ステータスを確認するには、**show aaa authentication login error-enable** コマンドを使用します。ユーザがログインを試みた際にリモート AAA サーバが認証要求に応答しなかった場合、ACE はローカル ユーザ データベースに切り替えてログイン手続きを進めます。error-enabled 機能を有効化してあると、ユーザのターミナルに次のメッセージが表示されます。

  ```
  Remote AAA servers unreachable; local authentication done.
  ```

たとえば、TacServers サーバ グループを利用したコンソール認証を可能にするには、次のように入力します（フォールバック手段としてのローカル ログインの設定も伴っています）。

```
host1/Admin(config)# aaa authentication login console group TacServers
local none
```

ログイン認証のパスワード確認は有効なままです。

たとえば、パスワード確認をさせない場合は次のように入力します。

```
host1/Admin(config)# no aaa authentication login console group
TacServers local none
```

たとえば、ローカルの認証手段を元に戻すには、次のように入力します。

```
host1/Admin(config)# no aaa authentication login console group
TacServers local none
```

デフォルトのアカウンティング手段の定義

各ユーザの ACE への管理セッションから収集されるログ情報をアカウンティングと呼んでいます。この情報はトラブルシューティングと監査を目的としたレポートの生成に利用できます。アカウンティングは ACE 上のローカルに、または RADIUS サーバもしくは TACACS+ サーバを利用するリモートに実装できます。

デフォルトのアカウンティング手段を設定するには **aaa accounting default** コマンドを使用します。TACACS+ サーバまたは RADIUS サーバの個別グループとして識別される作成済み AAA サーバ グループを指定することも、ACE 上のローカル データベースを指定することもできます。

このコマンドの構文は次のとおりです。

aaa accounting default {**group** *group_name*} {**local**} {**none**}

引数とキーワードの内容は次のとおりです。

- **group** *group_name* **—** アカウンティング手段を、**aaa group server** コマンドで定義済みの TACACS+、または RADIUS サーバに関連付けます。サーバ グループ名は 64 文字までの英数字です。
- **local —** アカウンティング手段として、ACE 上のローカル データベースを指定します。
- **none —** ACE がパスワード確認を行わないように指定します。パスワードの確認が行われなくなります。このオプションを設定すると、ユーザは有効なパスワードを提供しなくてもログインできるようになります。

注意　このオプションの使用には注意を払ってください。設定後、ユーザは誰でも自由に ACE にアクセスできるようになります。

たとえば、リモート TACACS+ サーバを利用してユーザ アカウンティングを可能にするには、次のように入力します（フォールバック手段としてのローカル ログインの設定も伴っています）。

```
host1/Admin(config-context)# aaa accounting default group TacServers
local
```

デフォルトのアカウンティング手段に戻すには、次のように入力します。

```
host1/Admin(config-context)# no aaa accounting default group
TacServers local
```

AAA のステータスと統計情報の表示

このセクションの内容は、次のとおりです。

AAA グループの表示

設定したサーバ グループは **show aaa groups** コマンドで表示させることができます。このコマンドの構文は次のとおりです。

show aaa groups

たとえば、設定済みのサーバ グループを表示するには、次のように入力します。

```
host1/Admin# show aaa groups
    TACACS:
        TACACS_group1
    RADIUS:
        RAD_group1
    LDAP:
        LDAP_group2
```

RADIUS サーバ設定情報の表示

設定した RADIUS サーバおよびグループのパラメータは、**show aaa groups** コマンドで表示させることができます。

このコマンドの構文は次のとおりです。

show radius-server [**groups** | **sorted**]

オプションのキーワードは次のとおりです。

- **groups —**（任意）設定済みの RADIUS サーバ グループの情報。
- **sorted —**（任意）RADIUS サーバの情報を名前で並べ替えて表示します。

たとえば、設定済みの RADIUS サーバのパラメータを表示するには、次のように入力します。

```
host1/Admin# show radius-server
    retransmission count:1
    timeout value:1
    deadtime value:20
    total number of servers:2

    following RADIUS servers are configured:
            192.168.34.45:
                    available for authentication on port:1812
                    available for accounting on port:1813
            192.168.2.3:
                    available for authentication on port:1812
                    available for accounting on port:1813
                    RADIUS shared secret:********
```

たとえば、設定済みの RADIUS サーバ グループを表示するには、次のように入力します。

```
host1/Admin# show radius-server groups
    total number of groups:2

    following RADIUS server groups are configured:
            group radius:
                    server: all configured radius servers
            group RAD_Server_Group:
                    deadtime is 0
```

たとえば、RADIUS サーバ グループを並べ替えて表示するには、次のように入力します。

```
host1/Admin# show radius-server sorted
    retransmission count:1
    timeout value:1
    deadtime value:20
    total number of servers:2

    following RADIUS servers are configured:
            192.168.34.45:
                    available for authentication on port:1812
                    available for accounting on port:1813
            192.168.2.3:
                    available for authentication on port:1812
                    available for accounting on port:1813
                    RADIUS shared secret:********
```

TACACS+ サーバ設定情報の表示

設定した TACACS+ サーバおよびグループのパラメータは **show tacacs-server** コマンドで表示させることができます。

このコマンドの構文は次のとおりです。

show tacacs-server [**groups** | **sorted**]

オプションのキーワードは次のとおりです。

- **groups —**（任意）設定済みの TACACS+ サーバ グループの情報。
- **sorted —**（任意）TACACS+ サーバの情報を名前で並べ替えて表示します。

たとえば、設定済みの TACACS+ サーバ パラメータを表示するには、次のように入力します。

```
host1/Admin# show tacacs-server
    Global TACACS+ shared secret:tacacsPword
    timeout value:30
    total number of servers:3

    following TACACS+ servers are configured:
    192.168.58.91:
    available on port:2
    cisco.com:
    available on port:49
    192.168.22.95:
    available on port:49
    TACACS+ shared secret:MyKey
```

たとえば、設定済みの TACACS+ サーバ グループを表示するには、次のように入力します。

```
host1/Admin# show tacacs-server groups
    total number of groups:1

    following TACACS+ server groups are configured:
    group TacServers:
    server 192.168.58.91 on port 2
```

たとえば、RADIUS サーバを並べ替えて表示するには、次のように入力します。

```
host1/Admin# show tacacs-server sorted
    timeout value:1
    total number of servers:1
```

LDAP サーバ設定情報の表示

設定した LDAP サーバおよびサーバ グループのパラメータは、**show ldap-server** コマンドで表示させることができます。

このコマンドの構文は次のとおりです。

show ldap-server [**groups**]

任意で **groups** キーワードを使うと、設定されている LDAP サーバ グループの情報を表示します。

設定済みの LDAP サーバ パラメータを表示するには、次のように入力します。

```
host1/Admin# show ldap
    timeout : 5
            port : 389
    total number of servers : 1
```

設定済みの LDAP サーバ グループを表示するには、次のように入力します。

```
host1/Admin# show ldap-server groups
    total number of groups: 1

    following LDAP server groups are configured:
        group LDAP_Server_Group1:
            baseDN: "dc=sns,dc=cisco,dc=com"
            user profile attribute: usrprof
            search filter: "(&(objectclass=person)
    (&(cn=$userid)(cid=$contextid)))"
```

アカウンティング設定情報の表示

show aaa accounting コマンドを使用すると、ACE のアカウンティングの設定情報を表示できます。

このコマンドの構文は次のとおりです。

show aaa accounting

たとえば、アカウンティング設定情報を表示するには、次のように入力します。

```
host1/Admin# show aaa accounting
    default: local
```

アカウンティング ログ情報の表示

show accounting log コマンドを使用すると、ACE のアカウンティング ログ設定情報を表示できます。

このコマンドの構文は次のとおりです。

show accounting log [*size*]

任意で引数 *size* を使用すると、ローカルのアカウンティング ログ ファイルの大きさを 0 ～ 250,000 バイトの範囲で表示します。デフォルトは 250,000 バイトです。

たとえば、アカウンティング ログ情報を表示するには、次のように入力します。

```
host1/Admin# show aaa accounting log
    Sat Jan  1 00:02:55 2000:start:/dev/ttyS00_946684975:admin:
    Sat Nov  5 00:20:04 2005:update:/dev/ttyS00_946684975:admin:0:ft
    interface vlan
    50
    Sat Nov  5 00:20:05 2005:update:/dev/ttyS00_946684975:admin:1:ip
    address 12.1.1.
    2 255.255.255.0
    Sat Nov  5 00:20:05 2005:update:/dev/ttyS00_946684975:admin:1:peer
    ip 12.1.1.1 2
    55.255.255.0
    Sat Nov  5 00:20:05 2005:update:/dev/ttyS00_946684975:admin:1:no
    shutdown
    Sat Nov  5 00:20:12 2005:update:/dev/ttyS00_946684975:admin:0:ft
    peer 1
    Sat Nov  5 00:20:12
    2005:update:/dev/ttyS00_946684975:admin:0:ft-interface vlan
    50
    Sat Nov  5 00:20:41 2005:update:/dev/ttyS00_946684975:admin:0:log
    console 6
    Sat Nov  5 00:20:58 2005:update:/dev/ttyS00_946684975:admin:0:ft
    group 1
    Sat Nov  5 00:20:58 2005:update:/dev/ttyS00_946684975:admin:0:peer
    1
    Sat Nov  5 00:20:58
    2005:update:/dev/ttyS00_946684975:admin:0:priority 50
    Sat Nov  5 00:20:58
    2005:update:/dev/ttyS00_946684975:admin:0:associate-context
    Admin
    Sat Nov  5 00:20:58
    2005:update:/dev/ttyS00_946684975:admin:0:inservice
    .
    .
```

認証設定情報の表示

show aaa authentication コマンドを使用すると、ACE の認証設定情報を表示できます。

このコマンドの構文は次のとおりです。

show aaa authentication [**login error-enable**]

任意で **login error-enable** キーワードを使用すると、現在のログイン エラー メッセージ表示ステータスを表示できます。

たとえば、設定済みの認証パラメータを表示するには、次のように入力します。

```
host1/Admin# show aaa authentication
    default: group TacServers local none
    console: local

host1/Admin# show aaa authentication login error-enable
    enabled
```