認証プロキシの設定

Cisco IOS Firewall 認証プロキシ機能では、動的かつユーザごとの認証と認可、業界標準の TACACS+ および RADIUS 認証プロトコルを使用したユーザの認証が可能です。ユーザによる接続の認証と認可により、ネットワーク攻撃に対するより強力な保護が可能になります。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報と注意事項については、ご使用のプラットフォームとソフトウェア リリースに対応したリリース ノートを参照してください。この章に記載されている機能の詳細、および各機能がサポートされているリリースのリストについては、「認証プロキシの機能情報」（P.35）を参照してください。

プラットフォーム サポートと Cisco IOS および Catalyst OS ソフトウェア イメージ サポートに関する情報を入手するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator には、http://tools.cisco.com/ITDIT/CFN/jsp/index.jsp からアクセスできます。Cisco.com のアカウントは必要ありません。

この章の構成

認証プロキシを設定するための前提条件

認証プロキシを設定する前に、次のことを確認してください。

- 認証プロキシが正しく機能するためには、クライアント ホストで次のブラウザ ソフトウェアが動作している必要があります。
  - Microsoft Internet Explorer 3.0 以降
  - Netscape Navigator 3.0 以降
- 認証プロキシには、標準のアクセス リストを使用するオプションがあります。認証プロキシを設定する前に、アクセス リストを使用してトラフィックをフィルタする方法について確実に理解する必要があります。アクセス リストを Cisco IOS Firewall とともに使用する方法の概要については、「Access Control Lists: Overview and Guidelines」の章を参照してください。
- 認証プロキシは、シスコの Authentication, Authorization, and Accounting（AAA; 認証、認可、アカウンティング）の枠組みで実装されているユーザ認証と認可を使用します。認証プロキシを設定する前に、AAA ユーザ認証、認可、およびアカウンティングの設定方法について理解する必要があります。ユーザ認証、認可、およびアカウンティングについては、「Authentication, Authorization, and Accounting (AAA)」の章を参照してください。
- Cisco IOS Firewall とともに認証プロキシを正常に実行するには、ファイアウォール上で CBAC を設定します。CBAC 機能の詳細については、「Configuring Context-Based Access Control」の章を参照してください。

認証プロキシを設定するための制約事項

- 認証プロキシは、HTTP 接続だけを開始します。
- HTTP サービスは、標準的な（ウェルノウン）ポートで動作している必要があります。HTTP の場合はポート 80 です。
- セキュアな認証のために、クライアント ブラウザで JavaScript がイネーブルになっている必要があります。
- 認証プロキシ アクセス リストは、ルータを通過するトラフィックに適用されます。ルータ宛のトラフィックは、Cisco IOS ソフトウェアで提供される既存の認証方式によって認証されます。
- 認証プロキシでは、同時使用がサポートされていません。つまり、2 人のユーザが同じホストから同時にログインしようとした場合、認証と認可は、最初に有効なユーザ名とパスワードを送信したユーザだけに適用されます。
- 複数の AAA サーバまたは異なる AAA サーバを使用したロード バランシングはサポートされていません。

認証プロキシの設定に関する情報

Cisco IOS Firewall 認証プロキシ機能を使用すると、ネットワーク管理者は、詳細なセキュリティ ポリシーをユーザごとに適用できます。以前は、ユーザの身元と関連する認可済みアクセスをユーザの IP アドレスに関連付けるか、1 つのセキュリティ ポリシーをユーザ グループまたはサブネットワーク全体に適用する必要がありました。現在では、ユーザごとのポリシーに基づいてユーザを特定し認可することができます。複数のユーザに一般的なポリシーを適用するのではなく、個人に対してアクセス権を調整できます。

認証プロキシ機能を使用すると、ユーザは、ネットワークにログインしたり、HTTP 経由でインターネットにアクセスでき、ユーザ固有のアクセス プロファイルが、CiscoSecure ACS または他の RADIUS または TACACS+ 認証サーバから自動的に取得されて適用されます。ユーザ プロファイルは、認証されたユーザからのアクティブ トラフィックが存在するときにのみ、アクティブになります。

認証プロキシは、Network Address Translation（NAT; ネットワーク アドレス変換）、Context-based Access Control（CBAC; コンテキストベース アクセス コントロール）、IP Security（IPSec）暗号化、Cisco Secure VPN Client（VPN クライアント）ソフトウェアなど、他の Cisco IOS セキュリティ機能と互換性があります。

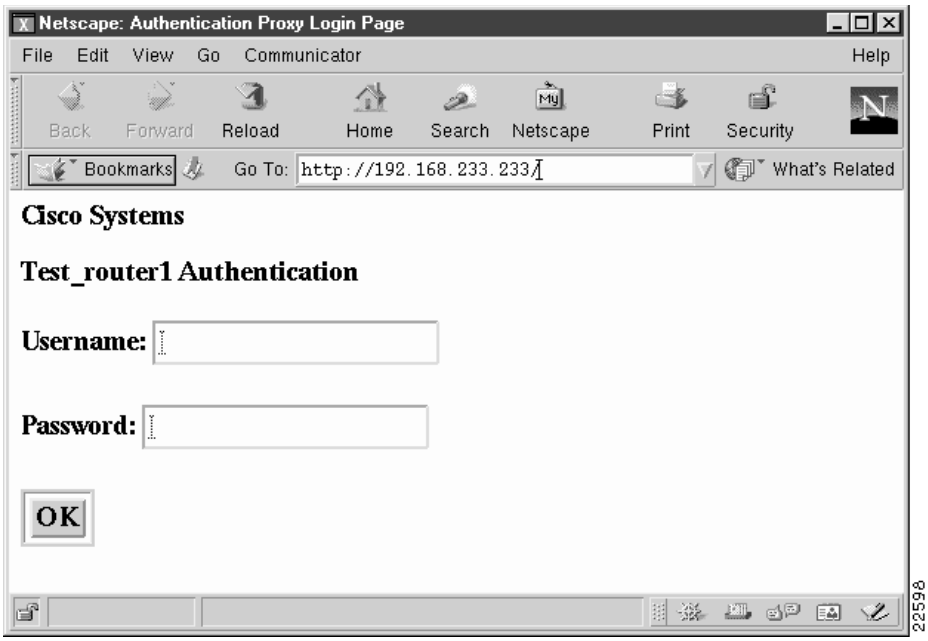
ここでは、次の各手順について説明します。

認証プロキシの仕組み

ユーザがファイアウォールを通じて HTTP セッションを開始すると、認証プロキシが起動されます。認証プロキシは、まずユーザが認証済みかどうかを確認します。ユーザの有効な認証エントリが存在する場合、認証プロキシによるそれ以上の介入なしに接続が完了します。エントリが存在しない場合、認証プロキシは HTTP 接続要求に対し、ユーザ名とパスワードの入力をユーザに求める応答を返します。

図 1 に、認証プロキシの HTML ログイン ページを示します。

図 1　　認証プロキシの HTML ログイン ページ



ユーザは、有効なユーザ名とパスワードを入力することで、認証サーバで正常に認証される必要があります。

認証が成功すると、ユーザの認可プロファイルが AAA サーバから取得されます。認証プロキシは、このプロファイル内の情報を使用して、動的な Access Control Entry（ACE; アクセス コントロール エントリ）を作成し、入力インターフェイスのインバウンド（入力）Access Control List（ACL; アクセス コントロール リスト）と、出力インターフェイスのアウトバウンド（出力）ACL に追加します（出力 ACL がインターフェイスに存在する場合）。この処理により、ファイアウォールは、認証済みユーザに、認可プロファイルで許可されたネットワークへのアクセスを許可します。たとえば、Telnet がユーザのプロファイルで許可されている場合、ユーザはファイアウォールを通じた Telnet 接続を開始できます。

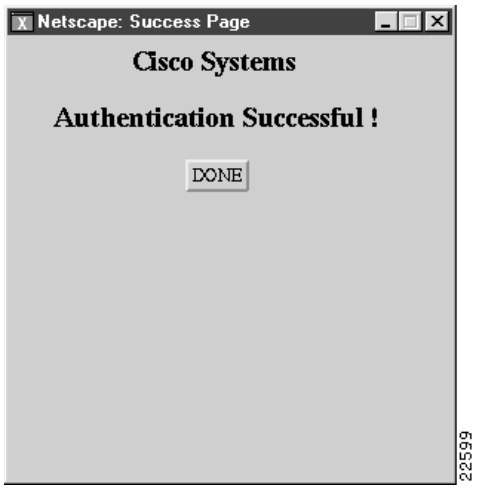
認証が失敗した場合、認証プロキシは、ユーザに失敗したことを報告し、何度か再試行するかどうかを訪ねます。5 回続けて認証に失敗した場合、2 分間待ってから、認証プロキシを起動するために別の HTTP セッションを開始する必要があります。

ログイン ページは、ユーザが Web サーバの情報にアクセスするための要求を行うたびに更新されます。

認証プロキシは、ダウンロードしたアクセス リスト中の送信元 IP アドレスを認証済みホストの送信元 IP アドレスで置き換えることで、ユーザ プロファイルの各アクセス リスト エントリをカスタマイズします。

認証プロキシは、動的な ACE をインターフェイス設定に追加するのと同時に、ログインが成功したことを確認するメッセージをユーザに送信します。図 2 に、HTML ページのログイン ステータスを示します。

図 2　　認証プロキシのログイン ステータス メッセージ



認証プロキシは、各ユーザ プロファイルに対し非アクティビティ（アイドル）タイマーを設定します。ファイアウォール経由のアクティビティがある限り、ユーザのホストから送信された新しいトラフィックによって認証プロキシは起動されず、認可済みのユーザ トラフィックに対してファイアウォールを通じたアクセスが許可されます。

アイドル タイマーが満了した場合、認証プロキシはユーザのプロファイル情報と動的なアクセス リスト エントリを削除します。この処理が実行されると、クライアントからのトラフィックはブロックされます。ユーザは、別の HTTP 接続を開始し、認証プロキシを起動する必要があります。

セキュアな認証

認証プロキシでは、JavaScript を使用して、クライアント ブラウザを使用したセキュアな認証が実現されます。セキュアな認証は、クライアントが、誤って認証プロキシ ルータ以外のネットワーク Web サーバにユーザ名とパスワードを送ることを防ぎます。

ここでは、次の各手順について説明します。

- 「JavaScript を使用した操作」
- 「JavaScript を使用しない場合の操作」

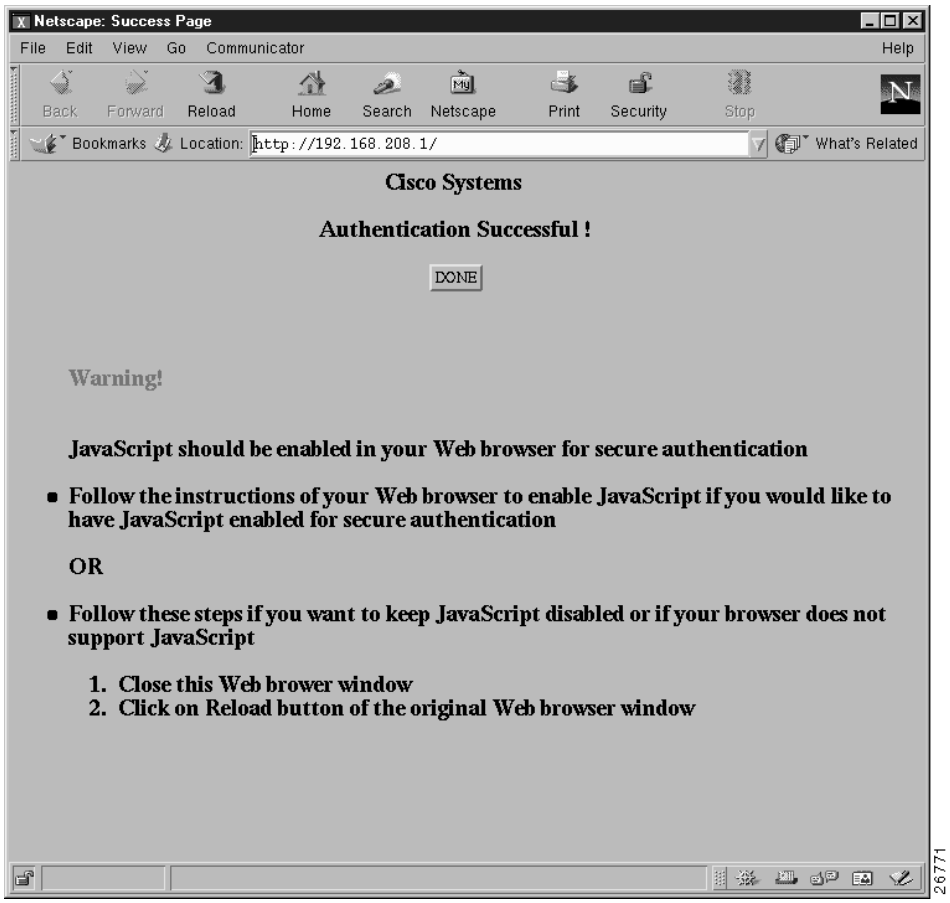
JavaScript を使用した操作

ユーザは、HTTP 接続を開始する前に、ブラウザ上で JavaScript をイネーブルにする必要があります。ブラウザで JavaScript がイネーブルになっている状態で、セキュアな認証が自動的に実行され、図 2 に示す認証メッセージが表示されます。ユーザの HTTP 接続は自動的に完了します。

JavaScript を使用しない場合の操作

クライアント ブラウザが JavaScript をサポートしていない場合や、サイトのセキュリティ ポリシーでユーザが JavaScript をイネーブルにすることが禁止されている場合、ログインしようとするとポップアップ ウィンドウに手動で接続を完了するための手順が表示されます。図 3 に、ブラウザで JavaScript がディセーブルになっている場合の認証プロキシのログイン ステータス メッセージを示します。

図 3　JavaScript がディセーブルになっている場合の認証プロキシのログイン ステータス メッセージ



このウィンドウを閉じるには、ブラウザの [File] メニューの [Close] をクリックします。

ポップアップ ウィンドウを閉じた後、認証ログイン ページが表示されているブラウザ ウィンドウの [Reload]（Internet Explorer の場合は [Refresh]）をクリックする必要があります。ユーザの最後の認証の試みが成功した場合、[Reload] をクリックすると、ユーザが取得しようとしている Web ページが表示されます。ユーザの最後の試みが失敗した場合、[Reload] をクリックすると、認証プロキシがクライアントの HTTP トラフィックを再度代行受信し、ユーザ名とパスワードの入力を求める別のログイン ページが表示されます。

JavaScript がイネーブルになっていない場合、サイト管理者は、「JavaScript を使用しないユーザ接続の確立」で説明するように、ポップアップ ウィンドウを閉じるための正しい手順を実行するよう、ユーザに忠告することを推奨します。

認証プロキシの使用

ユーザに対して透過的に動作する Cisco IOS Firewall のいくつかの機能と異なり、認証プロキシ機能では、クライアント ホスト上でいくつかの対話が必要です。表 1 で、認証プロキシとクライアント ホストの対話について説明します。

表 1　認証プロキシとクライアント ホストの対話

認証プロキシのクライアントとの動作	説明
HTTP 接続の開始	ユーザが現在ファイアウォール ルータで認証済みでない場合、ユーザが HTTP 接続を開始すると認証プロキシが起動されます。ユーザがすでに認証済みの場合、認証プロキシはユーザに対して透過的です。
ログイン ページを使用したログイン	認証プロキシを起動すると、HTML ベースのログイン ページが生成されます。ユーザは、AAA サーバで認証されるために、ユーザ名とパスワードを入力する必要があります。図 1 に、認証プロキシのログイン ページを示します。
クライアントでのユーザの認証	ログインの試行の後の認証プロキシの動作は、ブラウザで JavaScript がイネーブルになっているかどうかで変わります。JavaScript がイネーブルになっており、認証が成功した場合、認証プロキシは、図 2 に示すように、認証のステータスを示すメッセージを表示します。認証ステータスが表示された後、プロキシは自動的に HTTP 接続を完了します。 JavaScript がディセーブルになっており、認証が成功した場合、認証プロキシは、接続を完了するための追加の手順を表示したポップアップ ウィンドウを生成します。図 3 を参照してください。 いずれの場合も、認証が成功しなかった場合は、ユーザはログイン ページから再度ログインする必要があります。

認証プロキシを使用すべき場合

認証プロキシを使用するのが望ましい状況は次のとおりです。

- ホストの IP アドレスやグローバル アクセス ポリシーに基づいてアクセス コントロールを設定するのではなく、認証サーバによって提供されているサービスを使用して、個人ごと（ユーザごと）にアクセス権を管理する場合。任意のホスト IP アドレスからのユーザを認証および認可することにより、ネットワーク管理者は、DHCP を使用してホスト IP アドレスを設定できるようにもなります。
- ファイアウォールを通じたイントラネットやインターネット サービスまたはホストへのアクセスを許可する前に、ローカル ユーザを認証および認可する場合。
- ファイアウォールを通じたローカル サービスまたはホストへのアクセスを許可する前に、リモート ユーザを認証および認可する場合。
- 特定のエクストラネット ユーザに対するアクセスを制御する場合。たとえば、企業パートナーの財務責任者を、あるアクセス権のセットを使用して認証および認可し、同じパートナーの技術責任者を、別のアクセス権のセットを使用するように認可することができます。
- 認証プロキシを VPN クライアント ソフトウェアとともに使用して、ユーザを検証し、特定のアクセス権を割り当てる場合。
- 認証プロキシを AAA アカウンティングとともに使用して、課金、セキュリティ、またはリソース割り当てのために使用可能な「開始」および「停止」アカウンティング レコードを生成することで、ユーザが認証済みホストからのトラフィックを追跡できるようにする場合。

認証プロキシの適用

認証プロキシは、ユーザごとの認証と認可を行うルータの任意のインターフェイスで、インバウンド方向に適用します。認証プロキシをインターフェイスでインバウンド方向に適用することで、ユーザの初期接続要求は、ファイアウォールによる他の処理に渡される前に、認証プロキシによって代行受信されます。ユーザが AAA サーバによる認証に失敗すると、接続要求はドロップされます。

認証プロキシの適用方法は、セキュリティ ポリシーに依存します。たとえば、インターフェイスを通過するすべてのトラフィックをブロックし、認証プロキシ機能をイネーブルにして、ユーザが開始したすべての HTTP 接続に対して認証と認可を義務付けることができます。ユーザは、AAA サーバで正常に認証されない限り、サービスの利用が認可されません。

認証プロキシ機能では、標準のアクセス リストを使用し、どのホストまたはホスト グループからの初期 HTTP トラフィックに対してプロキシを起動するかを指定できます。

図 4 に示す認証プロキシは、LAN インターフェイスに適用されており、すべてのネットワーク ユーザは、初期接続時に認証される必要があります（すべてのトラフィックは各インターフェイスでブロックされます）。

図 4　ローカル インターフェイスでの認証プロキシの適用

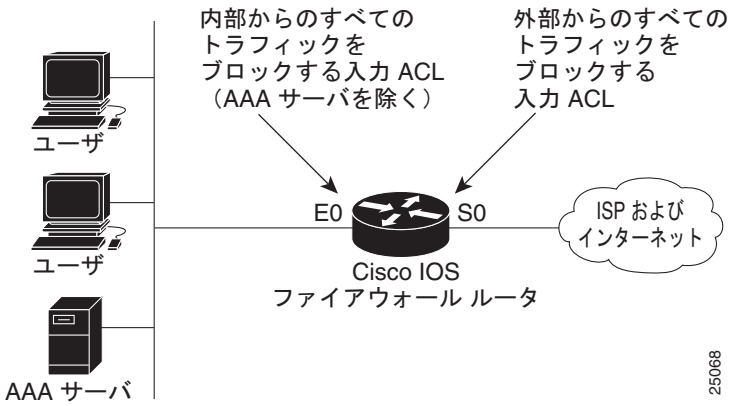
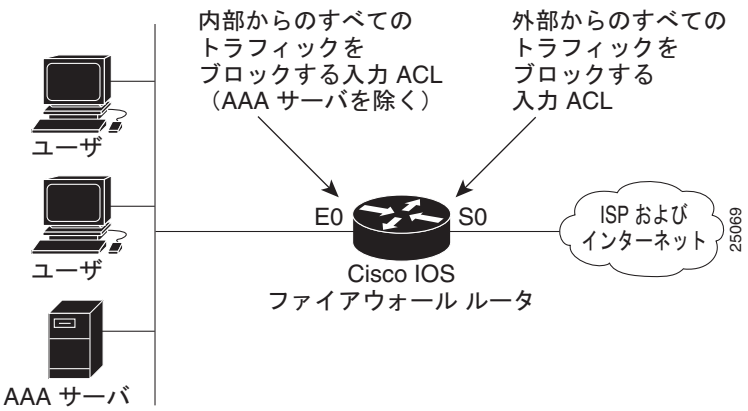


図 5 に示す認証プロキシは、ダイヤルイン インターフェイスに適用され、すべてのネットワーク トラフィックが各インターフェイスでブロックされます。

図 5　外部インターフェイスでの認証プロキシの適用



ワンタイム パスワード（OTP）を使用した動作

One-Time Password（OTP; ワンタイム パスワード）を使用する場合、ユーザはユーザ名とワンタイム パスワードを HTML のログイン ページに通常どおり入力します。

ユーザは、最初の 3 回の試行の間に正しいトークン パスワードを入力する必要があります。入力を 3 回間違えた場合、2 つの有効なトークン パスワードを続けて入力しないと、AAA サーバでの認証が許可されません。

他のセキュリティ機能との互換性

この認証プロキシは、次に示す Cisco IOS ソフトウェアおよび Cisco IOS のセキュリティ機能と互換性があります。

- Cisco IOS Firewall Intrusion Detection System（IDS）
- NAT
- CBAC
- IPSec 暗号化
- VPN クライアント ソフトウェア

認証プロキシは、Cisco IOS Firewall IDS および IPSec 暗号化機能と透過的に連動します。次のセクションでは、NAT、CBAC、および VPN クライアント ソフトウェアの各機能と認証プロキシの関係について説明します。

- 「NAT の互換性」
- 「CBAC との互換性」
- 「VPN クライアントの互換性」

NAT の互換性

認証プロキシ機能は、ACL と認証が NAT 変換の前に完了している場合にだけ、NAT と互換性があります。NAT は認証プロキシ機能と互換性がありますが、認証プロキシを使用するうえで NAT は必須ではありません。

CBAC との互換性

認証プロキシは、CBAC セキュリティ機能と互換性がありますが、認証プロキシ機能を使用するために CBAC は必須ではありません。

認証プロキシの認可は、手動で作成された ACL の先頭に動的に追加されるアクセス コントロール エントリ（ACE）を返します。それ以降、ACL を「保護された側」のインバウンドインターフェイスに適用し、認可されたユーザの送信元 IP アドレスのリモート ネットワークへのアクセスを許可または禁止します。

VPN クライアントの互換性

ネットワーク管理者は、認証プロキシを使用して、VPN クライアント トラフィックに対し、追加のセキュリティ レイヤとアクセス コントロールを適用できます。VPN クライアントが HTTP 接続を開始した場合、認証プロキシはまず既存のクライアント認証を確認します。クライアントが認証済みの場合、認可されたトラフィックは許可されます。クライアントが認証済みでない場合、HTTP 要求によって認証プロキシが起動され、ユーザに対しユーザ名とパスワードの入力が求められます。

ユーザ認証が成功した場合、認証プロキシは AAA サーバからユーザ プロファイルを取得します。ユーザ プロファイル エントリ内の送信元アドレスは、復号化されたパケット内の、認証済み VPN クライアントの IP アドレスで置き換えられます。

AAA アカウンティングとの互換性

認証プロキシを使用して、課金やセキュリティ監査で使用するために十分な情報を含む「開始」および「停止」アカウンティング レコードを生成できます。そうすることで、認証プロキシ サービスを使用する認証済みホストの動作をモニタできます。

認証プロキシのキャッシュと関連付けられている動的アクセス コントロール リストが作成されると、認証プロキシは認証済みホストからのトラフィックの追跡を開始します。アカウンティングでは、このイベントに関するデータが、他のユーザのデータとともにデータ構造に保存されます。アカウンティング開始オプションがイネーブルになっている場合、この時点でアカウンティング レコード（「開始」レコード）を生成できます。認証済みホストからの以降のトラフィックは、認証プロキシによって作成された動的な ACL がパケットを受信すると記録されます。

認証プロキシのキャッシュが満了して削除されると、経過時間などの追加のデータがアカウンティング情報に追加され、「停止」レコードがサーバに送信されます。この時点で、情報がデータ構造から削除されます。

認証プロキシ ユーザ セッションに対するアカウンティング レコードは、キャッシュおよび動的 ACL の使用に関連付けられます。

（注） アカウンティング レコードは、RADIUS と TACACS+ の両方に対し、RADIUS アトリビュート 42、46、および 47 を含んでいる必要があります。

RADIUS アトリビュートの詳細については、付録「RADIUS アトリビュート」を参照してください。

DoS 攻撃（サービス拒絶攻撃）からの保護

認証プロキシは、受信 HTTP 要求のレベルをモニタします。各要求に対し、認証プロキシはユーザのログイン クレデンシャルの入力を求めます。オープン要求が多い場合、ルータが DoS 攻撃を受けていることを示している可能性があります。認証プロキシは、オープン要求のレベルを制限し、オープン要求の数が 40 未満になるまで、追加の要求をドロップします。

ファイアウォールが、認証が必要な大量の接続要求を受信している場合、正規のネットワーク ユーザが接続を行うときに遅延が発生したり、接続が拒否されて接続の再試行が必要になることがあります。

認証プロキシでのスプーフィングの危険性

認証プロキシが起動されると、ユーザ アクセス権を持つインターフェイスを一時的に再設定することで、ファイアウォール中に動的な開口が作成されます。この開口が存在する間に、別のホストが認証済みユーザのアドレスを偽装し、ファイアウォールの背後へのアクセスを獲得する可能性があります。認証プロキシは、アドレス スプーフィングの問題を起こしません。この問題は、ユーザの関心事としてここに明記しています。スプーフィングは、すべてのアクセス リストにつきまとう問題であり、認証プロキシは特にこの問題に対処していません。

Lock-and-Key 機能との比較

Lock-and-Key は、認証とダイナミック アクセス リストを使用して、ファイアウォールを通じたユーザ アクセスを可能にする、Cisco IOS Firewall のもう 1 つの機能です。表 2 に、認証プロキシと Lock-and-Key 機能の比較を示します。

表 2　　認証プロキシ機能と Lock-and-Key 機能の比較

Lock-and-Key	認証プロキシ
Telnet 接続要求により起動	HTTP 接続要求により起動
TACACS+、RADIUS、またはローカル認証	TACACS+ または RADIUS 認証および認可
アクセス リストはルータだけで設定	アクセス リストは必ず AAA サーバから取得
アクセス権は、ユーザのホスト IP アドレスに基づいて許可	アクセス権は、ユーザごとおよびホスト IP アドレスごとに許可
アクセス リストは、各ホスト IP アドレスに対し 1 つに制限	アクセス リストは、AAA サーバ上のユーザ プロファイルによって定義された複数のエントリを持つことが可能
固定の IP アドレスを特定のユーザに関連付け。ユーザは、その IP アドレスを持つホストからログインする必要あり。	DHCP ベースのホストの AP アドレスを許可。つまり、ユーザは、任意のホストからログインし、認証と認可を受けることが可能。

認証プロキシは、ユーザごとのセキュリティ ポリシーを提供する任意のネットワーク環境で使用します。Lock-and-Key は、ローカル認証と、ホスト アドレスに基づく限定的な数のルータベースのアクセス コントロール ポリシーの恩恵を受けるネットワーク環境で使用します。Lock-and-Key は、Cisco Secure Integrated Software を使用しない環境で使用します。

認証プロキシの設定方法

認証プロキシ機能を設定するには、次の手順を実行します。

- 「AAA の設定」（必須）
- 「認証プロキシ用の HTTP サーバの設定」（必須）
- 「認証プロキシの設定」（必須）
- 「認証プロキシの確認」（任意）

この章に示すコマンドを使用した認証プロキシの設定例については、この章の最後にある「認証プロキシの設定例」のセクションを参照してください。

AAA の設定

AAA サービス用に認証プロキシを設定する必要があります。認可をイネーブルにし認可方式を定義するには、次の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。

	コマンド	目的
ステップ 1	`router(config)# aaa new-model`	ルータで AAA 機能をイネーブルにします。
ステップ 2	`router(config)# aaa authentication login default TACACS+ RADIUS`	ログイン時の認証方式リストを定義します。
ステップ 3	`router(config)# aaa authorization auth-proxy default [method1 [method2...]]`	**auth-proxy** キーワードを使用して、AAA 方式に対する認証プロキシをイネーブルにします。
ステップ 4	`router(config)# aaa accounting auth-proxy default start-stop group tacacs+`	**auth-proxy** キーワードを使用して、認可ポリシーを、ダウンロード可能なダイナミック ACL として設定します。このコマンドは、認証プロキシのアカウンティングをアクティブ化します。
ステップ 5	`router(config)# tacacs-server host hostname`	AAA サーバを指定します。RADIUS サーバの場合は、**radius server host** コマンドを使用します。
ステップ 6	`router(config)# tacacs-server key key`	ルータと AAA サーバとの間の通信用の認証および暗号化キーを設定します。RADIUS サーバの場合、**radius server key** コマンドを使用します。
ステップ 7	`router(config)# access-list access-list-number permit tcp host source eq tacacs host destination`	AAA サーバがトラフィックをファイアウォールに返すのを許可する ACL エントリを作成します。送信元アドレスは AAA サーバの IP アドレスで、宛先は AAA サーバが存在するルータ インターフェイスの IP アドレスです。

認証プロキシでは、ファイアウォール ルータで AAA を設定するのに加えて、ユーザごとのアクセス プロファイル設定が AAA サーバ上に必要です。認証プロキシをサポートするために、ここに示す概要に従い、AAA 認可サービス **auth-proxy** を AAA サーバ上で設定します。

- **auth-proxy** キーワードに対する個別の認可セクションを定義して、ダウンロード可能なユーザ プロファイルを指定します。このキーワードは、EXEC などの他の種類のサービスと干渉しません。次に、TACACS サーバ上のユーザ プロファイルの例を示します。

```
default authorization = permit
key = cisco
user = newuser1 {
login = cleartext cisco
```

```
service = auth-proxy
{
priv-lvl=15
proxyacl#1="permit tcp any any eq 26"
proxyacl#2="permit icmp any host 60.0.0.2"
proxyacl#3="permit tcp any any eq ftp"
proxyacl#4="permit tcp any any eq ftp-data"
proxyacl#5="permit tcp any any eq smtp"
proxyacl#6="permit tcp any any eq telnet"
}
}
```

- AAA サーバのユーザ設定でサポートされる唯一のアトリビュートは、proxyacl#n です。プロファイル中のアクセス リストを設定する際には、proxyacl#n アトリビュートを使用します。アトリビュート proxyacl#n は、RADIUS と TACACS+ の両方の attribute-value（AV）のペア用です。
- すべてのユーザの特権レベルは 15 に設定する必要があります。
- AAA サーバ上のユーザ プロファイル内のアクセス リストには、**permit** キーワードだけを含むアクセス コマンドが必要です。
- 各ユーザ プロファイル アクセス リスト エントリの **any** キーワードに、送信元アドレスを設定します。ユーザ プロファイルがファイアウォールにダウンロードされるとき、アクセス リスト中の送信元アドレスは、認証プロキシ要求を行うホストの送信元アドレスで置き換えられます。
- サポートされる AAA サーバは次のとおりです。
  - CiscoSecure ACS 2.1.x for Windows NT
  - CiscoSecure ACS 2.3 for Windows NT
  - CiscoSecure ACS 2.2.4 for UNIX
  - CiscoSecure ACS 2.3 for UNIX
  - TACACS+ サーバ（vF4.02.alpha）
  - Ascend RADIUS サーバ radius-980618（必須のアトリビュートと値のペアのパッチ）
  - Livingston RADIUS サーバ（v1.16）

AAA サーバの設定例については、「AAA サーバのユーザ プロファイル例」のセクションを参照してください。

認証プロキシ用の HTTP サーバの設定

この作業は、ファイアウォール上で HTTP サーバをイネーブルにし、認証プロキシ用に HTTP サーバの AAA 認証方式を設定するために使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip http server**
4. **ip http access-class** *access-list-number*

手順の詳細

	コマンド	目的
ステップ 1	**enable** **例：** `Router> enable`	特権 EXEC モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** **例：** `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip http server** **例：** `Router# ip http server`	ルータ上で HTTP サーバをイネーブルにします。認証プロキシは HTTP サーバを使用してクライアントと通信し、ユーザ認証を行います。
ステップ 4	**ip http access-class** *access-list-number* **例：** `router(config)# configure terminal`	HTTP サーバのアクセス リストを指定します。「インターフェイスの設定例」のセクションで設定する標準のアクセス リスト番号を使用します。

認証プロキシの設定

認証プロキシを設定するには次のコマンドを使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip auth-proxy auth-cache-time** *min*
4. **ip auth-proxy auth-proxy-banner**
5. **ip auth-proxy name** *auth-proxy-name* **http** [**auth-cache-time** *min*] [**list** {*acl* | *acl-name*}]
6. **interface** *type*
7. **ip auth-proxy** *auth-proxy-name*

手順の詳細

	コマンド	目的
ステップ 1	**enable** **例：** `Router> enable`	特権 EXEC モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** **例：** `Router# 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。

	コマンド	目的
ステップ 3	**ip auth-proxy auth-cache-time** *min* **例：** `Router(config)# ip auth-proxy auth-cache-time 5`	（任意）グローバル認証プロキシ アイドル タイムアウト値を分単位で設定します。タイムアウトが発生すると、ユーザ認証エントリと、関連付けられているダイナミック アクセス リストがすべて削除されます。デフォルト値は 60 分です。 **（注）** このオプションは、任意の認証プロキシ ルールに対して使用し、任意の CBAC 検査ルールのアイドル タイムアウト値よりも大きな値に設定します。認証プロキシが認証キャッシュとそれに関連付けられているダイナミック ユーザ ACL を削除するとき、CBAC によってモニタされているいくつかのアイドル接続が存在する可能性があり、ユーザ固有の ACL を削除すると、これらのアイドル接続がハングするおそれがあります。CBAC のアイドル タイムアウトが短ければ、アイドル タイムアウトが発生したとき（つまり、認証プロキシがユーザ プロファイルを削除する前）に CBAC はこれらの接続をリセットします。
ステップ 4	**ip auth-proxy auth-proxy-banner** **例：** `Router(config)# configure terminal`	（任意）認証プロキシのログイン ページにファイアウォール ルータの名前を表示します。デフォルトではバナーはディセーブルになっています。

	コマンド	目的
ステップ 5	**ip auth-proxy name** *auth-proxy-name* **http** [**auth-cache-time** *min*] [**list** {*acl* |*acl-name*}] **例：** `Router(config)# ip auth-proxy name HQ_users http`	認証プロキシ ルールを作成します。ルールは、認証プロキシの適用方法を定義します。このコマンドは、HTTP プロトコル トラフィックを開始する接続を、認証プロキシ名に関連付けます。名前付きのルールをアクセス コントロール リスト（ACL）に関連付け、どのホストが認証プロキシ機能を使用するかを制御できます。標準のアクセス リストが定義されていない場合、名前付き認証プロキシ ルールが、接続開始パケットが設定済みのインターフェイスで受信されるすべてのホストからの HTTP トラフィックを代行受信します。 （任意）**auth-cache-time** オプションは、グローバル認証プロキシ キャッシュ タイマーを上書きします。このオプションにより、特定の認証プロキシ ルールに対し、タイムアウト値をより詳細に制御できます。値を指定しない場合、プロキシ ルールは、**ip auth-proxy auth-cache-time** コマンドで設定された値を使用します。 （任意）**list** オプションを使用すると、標準のアクセス リスト、拡張（1 ～ 199）アクセス リスト、または名前付きアクセス リストを、名前付き認証プロキシ ルールに適用できます。アクセス リスト中のホストによって開始された HTTP 接続は、認証プロキシによって代行受信されます。
ステップ 6	**interface** *type* **例：** `Router(config)# interface Ethernet0/0`	認証プロキシを適用するインターフェイス タイプを指定して、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 7	**ip auth-proxy** *auth-proxy-name* **例：** `Router(config-if)# ip auth-proxy HQ_users http`	インターフェイス コンフィギュレーション モードで、名前付き認証プロキシのルールをインターフェイスに適用します。このコマンドにより、指定の名前を持つ認証プロキシのルールがイネーブルになります。

認証プロキシの確認

認証プロキシの設定の確認には、次のいくつかの項目が含まれます。

- 「認証プロキシの設定の確認」（任意）
- 「JavaScript を使用したユーザ接続の確立」（任意）
- 「JavaScript を使用しないユーザ接続の確立」（任意）

認証プロキシの設定の確認

現在の認証プロキシの設定を確認するには、特権 EXEC モードで **show ip auth-proxy configuration** コマンドを使用します。

コマンド	目的
router# **show ip auth-proxy configuration**	認証プロキシの設定を表示します。

次の例で、グローバル認証プロキシ アイドル タイムアウト値は 60 分に設定され、名前付き認証プロキシ ルールは「pxy」であり、この名前付きルールのアイドル タイムアウト値は 1 分です。表示内容は、ホスト リストが指定されていないことを示しています。つまり、そのインターフェイスでのすべての接続開始 HTTP トラフィックに認証プロキシ ルールが適用されます。

```
router# show ip auth-proxy configuration
Authentication cache time is 60 minutes
Authentication Proxy Rule Configuration
Auth-proxy name pxy
http list not specified auth-cache-time 1 minutes
```

認証プロキシがルータで正常に設定されていることを確認するには、ルータを通じて HTTP 接続を開始するようユーザに依頼します。そのユーザに対し、AAA サーバで認証と認可が設定されている必要があります。ユーザ認証が成功した場合、ファイアウォールはそのユーザの HTTP 接続を完了します。認証が成功しなかった場合は、アクセス リストと AAA サーバの設定を確認します。

特権 EXEC モードで **show ip auth-proxy cache** コマンドを使用し、ユーザ認証エントリを表示します。

コマンド	目的
router# **show ip auth-proxy cache**	ユーザ認証エントリのリストを表示します。

認証プロキシ キャッシュにより、ホストの IP アドレス、送信元ポート番号、認証プロキシのタイムアウト値、接続の状態が一覧表示されます。認証プロキシの状態が HTTP_ESTAB の場合、ユーザ認証が成功したことを示します。

```
router# show ip auth-proxy cache
Authentication Proxy Cache
 Client IP 192.168.25.215 Port 57882, timeout 1, state HTTP_ESTAB
```

1 分間（この名前付きルールのタイムアウト値）待ち、ユーザに再度接続を試みるよう依頼します。1 分後、ユーザの接続は拒否されます。これは、認証プロキシにより、ユーザの認証エントリと、関連付けられているすべてのダイナミック ACL が削除されたためです。ユーザに対し新しい認証ログイン ページが表示され、ファイアウォールを通じてアクセスするには再度ログインする必要があります。

JavaScript を使用したユーザ接続の確立

クライアント ブラウザで JavaScript をイネーブルにした状態で認証プロキシを使用したクライアント接続を確認するには次の手順を実行します。

ステップ 1 クライアント ホストから、ファイアウォールを通じて HTTP 接続を開始します。これにより、認証プロキシのログイン ページが生成されます。

ステップ 2 認証プロキシのログイン ページで、ユーザ名とパスワードを入力します。

ステップ 3 [OK] をクリックしてユーザ名とパスワードを AAA サーバに送信します。

ログインが成功したか失敗したかを示すポップアップ ウィンドウが表示されます。認証に成功した場合、接続が自動的に完了します。認証が失敗した場合、認証プロキシは、ユーザに失敗したことを報告し、何度か再試行するかどうかを訪ねます。

（注） 認証に 5 回失敗した場合、ユーザは 2 分間待ってから、認証プロキシを起動する別の HTTP セッションを開始する必要があります。

JavaScript を使用しないユーザ接続の確立

セキュアな認証を行うために、認証プロキシの設計では JavaScript が必要です。ブラウザで JavaScript をイネーブルにせずに認証プロキシを使用することもできますが、ユーザがネットワーク接続を正しく確立しなかった場合にセキュリティ リスクが生じます。次に、JavaScript をディセーブルにした状態で接続を確立するための正しい手順を示します。ネットワーク管理者は、このセクションの手順を使用して、接続を適切に確立する方法をユーザに指示することを強く推奨します。

（注） この手順に従わないと、ユーザのクレデンシャルが認証プロキシ以外のネットワーク Web サーバに渡されたり、認証プロキシによってログインが拒否されるおそれがあります。

クライアント ブラウザで JavaScript がイネーブルでないときに認証プロキシを使用したクライアント接続を確認するには、次の手順を実行します。

ステップ 1 ファイアウォールを通じて HTTP 接続を開始します。

これにより、認証プロキシのログイン ページが生成されます。

ステップ 2 クライアントで、認証プロキシのログイン ページから、ユーザ名とパスワードを入力します。

ステップ 3 [OK] をクリックしてユーザ名とパスワードを AAA サーバに送信します。

ログインが成功したか失敗したかを示すポップアップ ウィンドウが表示されます。ポップアップ ウィンドウに認証が成功したことが表示される場合は、ステップ 7 に進みます。

ステップ 4 ポップアップ ウィンドウに、認証失敗のメッセージが表示される場合は、ブラウザの [File] メニューの [Close] をクリックします。

（注） ポップアップ ウィンドウを閉じるために、[Reload]（Internet Explorer の場合は [Refresh]）をクリックしないでください。

ステップ 5 元の認証ログイン ページで、ブラウザ ツールバーの [Reload]（Internet Explorer の場合は [Refresh]）クリックします。ユーザのログイン クレデンシャルがフォームからクリアされます。

（注） [OK] をクリックしないでください。再度ログインする前に、ユーザ名とパスワードをクリアし、フォームをリロードするには、[Reload] または [Refresh] をクリックする必要があります。

ステップ 6 ユーザ名とパスワードを再度入力します。

認証に成功した場合、ウィンドウが開き、認証成功を示すメッセージが表示されます。認証失敗のメッセージがウィンドウに表示される場合は、ステップ 4 に進みます。

ステップ 7 ブラウザの [File] メニューで [Close] をクリックします。

ステップ 8 元の認証プロキシのログイン ページで、[Reload]（Internet Explorer の場合は [Refresh]）クリックします。

認証プロキシは、Web サーバとの認証済みの接続を完了します。

認証プロキシのモニタおよびメンテナンス

ここでは、ダイナミック アクセス リスト エントリを表示する方法と、認証エントリを手動で削除する方法について説明します。ここでは、次の各手順について説明します。

- 「ダイナミック ACL エントリの表示」
- 「認証プロキシのキャッシュ エントリの削除」

ダイナミック ACL エントリの表示

ダイナミック アクセス リスト エントリは、使用中に表示できます。管理者またはアイドル タイムアウト パラメータによって認証プロキシ エントリがクリアされた後は、表示できなくなります。表示される一致の数は、アクセス リスト エントリがヒットした回数を示します。

認証プロキシによって現在確立されているダイナミック アクセス リスト エントリと一時的なアクセス リスト エントリを表示するには、特権 EXEC モードで **show ip access-lists** コマンドを使用します。

コマンド	目的
`router# show ip access-lists`	ダイナミック ACL エントリを含め、ファイアウォールで設定済みの標準アクセス リストおよび拡張アクセス リストを表示します。

次の例では、ACL 105 が、認証プロキシを設定する入力インターフェイスでインバウンド方向に適用されています。最初の表示は、認証前の ACL の内容を示しています。2 番めの表示は、AAA サーバによるユーザ認証後の同じ表示を示しています。

（注） NAT が設定されている場合、**show ip access list** コマンドにより、ダイナミック ACL エントリの変換後のホスト IP アドレスか、接続を開始したホストの IP アドレスが表示される場合があります。NAT の外部インターフェイスに対して ACL が適用される場合、変換後のアドレスが表示されます。ACL が NAT の内部インターフェイスに適用される場合、接続を開始するホストの IP アドレスが表示されます。**show ip auth-proxy cache** コマンドで、常に接続を開始したホストの IP アドレスが表示されます。

たとえば、次に示すのは、認証プロキシの前の ACL エントリのリストです。

```
Router# show ip access-lists
    .
    .
    .
Extended IP access list 105
 deny tcp any any eq telnet
 deny udp any any
 permit tcp any any (28 matches)
 permit ip any any
```

次の出力例は、ユーザ認証後の ACL エントリのリストを示しています。

```
Router# show ip access-lists
    .
    .
    .
Extended IP access list 105
! The ACL entries following user authentication are shown below.
 permit tcp host 192.168.25.215 any eq 26
 permit icmp host 192.168.25.215 host 60.0.0.2
 permit tcp host 192.168.25.215 any eq telnet
 permit tcp host 192.168.25.215 any eq ftp
 permit tcp host 192.168.25.215 any eq ftp-data
 permit tcp host 192.168.25.215 any eq smtp
 deny tcp any any eq telnet
 deny udp any any
 permit tcp any any (76 matches)
 permit ip any any
```

認証プロキシのキャッシュ エントリの削除

認証プロキシを使用中の場合、ダイナミック アクセス リストは、認証エントリの追加および削除に伴って動的に増減します。認証エントリのリストを表示するには、**show ip auth-proxy cache** コマンドを使用します。認証エントリを手動で削除するには、特権 EXEC モードで **clear ip auth-proxy cache** コマンドを使用します。

コマンド	目的
router# **clear ip auth-proxy cache** {* | *host ip address*}	タイムアウト前にファイアウォールから認証プロキシ エントリを削除します。すべての認証キャッシュ エントリを削除するにはアスタリスクを使用します。単一のホストのエントリを削除するには、特定の IP アドレスを入力します。

認証プロキシの設定例

認証プロキシ機能を設定するには、ルータと AAA サーバの両方の設定を変更する必要があります。以降のセクションでは、認証プロキシの設定例について説明します。

- 「認証プロキシの設定例」
- 「認証プロキシ、IPSec、および CBAC の設定例」
- 「認証プロキシ、IPSec、NAT、および CBAC の設定例」
- 「AAA サーバのユーザ プロファイル例」

これらの例全体で、感嘆符（!）はコメント行を示します。コメント行は、説明している設定エントリの前に記載されています。

認証プロキシの設定例

以降の例では、特定の認証プロキシの設定エントリを取り上げています。これらの例は、完全なルータ設定を表すものではありません。認証プロキシを使用した完全なルータの設定は、この章の後のセクションに含まれています。

ここでは、次の例について説明します。

- 「AAA の設定例」
- 「HTTP サーバの設定例」
- 「認証プロキシの設定例」
- 「インターフェイスの設定例」

AAA の設定例

```
aaa new-model
aaa authentication login default group tacacs group radius
! Set up the aaa new model to use the authentication proxy.
aaa authorization auth-proxy default group tacacs group radius
! Define the AAA servers used by the router.
aaa accounting auth-proxy default start-stop group tacacs+
! Set up authentication proxy with accounting.
tacacs-server host 172.31.54.143
tacacs-server key cisco
radius-server host 172.31.54.143
radius-server key cisco
```

HTTP サーバの設定例

```
! Enable the HTTP server on the router.
ip http server
! Set the HTTP server authentication method to AAA.
ip http authentication aaa
! Define standard access list 61 to deny any host.
access-list 61 deny any
! Use ACL 61 to deny connections from any host to the HTTP server.
ip http access-class 61
```

認証プロキシの設定例

```
! Set the global authentication proxy timeout value.
ip auth-proxy auth-cache-time 60
! Apply a name to the authentication proxy configuration rule.
ip auth-proxy name HQ_users http
```

インターフェイスの設定例

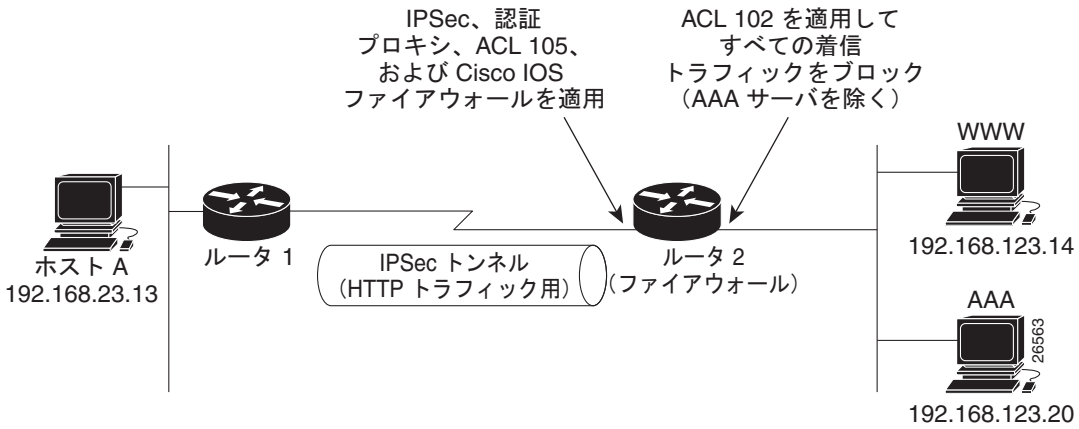
```
! Apply the authentication proxy rule at an interface.
interface e0
    ip address 10.1.1.210 255.255.255.0
    ip auth-proxy HQ_users
```

認証プロキシ、IPSec、および CBAC の設定例

次の例は、認証プロキシ、IPSec および CBAC 機能を使用するルータ設定を示します。図 6 に、設定を示します。

（注） 本機能を Cisco IOS ソフトウェア リリース 12.3(8)T 以降で使用する場合は、『Crypto Access Check on Clear-Text Packets』を参照してください。

図 6　　認証プロキシ、IPSec、および CBAC の設定例



この例では、ホスト A が Web サーバ（WWW）との HTTP 接続を開始します。ルータ 1 とルータ 2 間の HTTP トラフィックは、IPSec を使用して暗号化されます。認証プロキシ、IPSec、および CBAC は、ルータ 2 上のインターフェイス Serial0 で設定され、ファイアウォールとして機能しています。ACL 105 は、インターフェイス Serial0 ですべてのトラフィックをブロックします。ACL 102 は、ルータ 2 上のインターフェイス Ethernet0 に適用され、AAA サーバからのトラフィックを除くそのインターフェイス上のすべてのトラフィックをブロックします。

ホスト A が Web サーバとの HTTP 接続を開始すると、認証プロキシはホスト A でユーザ名とパスワードを入力するようユーザに要求します。これらのクレデンシャルは、認証および許可のために AAA サーバで検証されます。認証が正常に行われると、ユーザごとの ACL がファイアウォールにダウンロードされ、サービスが許可されます。

次の例では、完全を期すためにルータ 1 とルータ 2 の両方の設定を示します。

ルータ 1 の設定例

```
! Configure Router 1 for IPSec.
version 12.0
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router1
!
logging buffered 4096 debugging
no logging console
enable secret 5 $1$E0OB$AQF1vFZM3fLr3LQAOsudL/
enable password junk
!
username Router2 password 0 welcome
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco1234 address 10.0.0.2
!
crypto ipsec transform-set rule_1 ah-sha-hmac esp-des esp-sha-hmac
!
!
 crypto map testtag 10 ipsec-isakmp
 set peer 10.0.0.2
 set transform-set rule_1
 match address 155
!
interface Ethernet0/0
 ip address 192.168.23.2 255.255.255.0
 no ip directed-broadcast
 no ip route-cache
 no ip mroute-cache
!
interface Serial3/1
 ip address 10.0.0.1 255.0.0.0
 no ip directed-broadcast
 encapsulation PPP
 ip route-cache
 no ip mroute-cache
 no keepalive
 no fair-queue
 clockrate 56000
 crypto map testtag
!
!
ip classless
ip route 192.168.123.0 255.255.255.0 10.0.0.2
! Identify the IPSec specific traffic.
access-list 155 permit tcp host 192.168.23.13 host 192.168.123.14 eq www
access-list 155 permit tcp host 192.168.23.13 eq www host 192.168.123.14
```

ルータ 2 の設定例

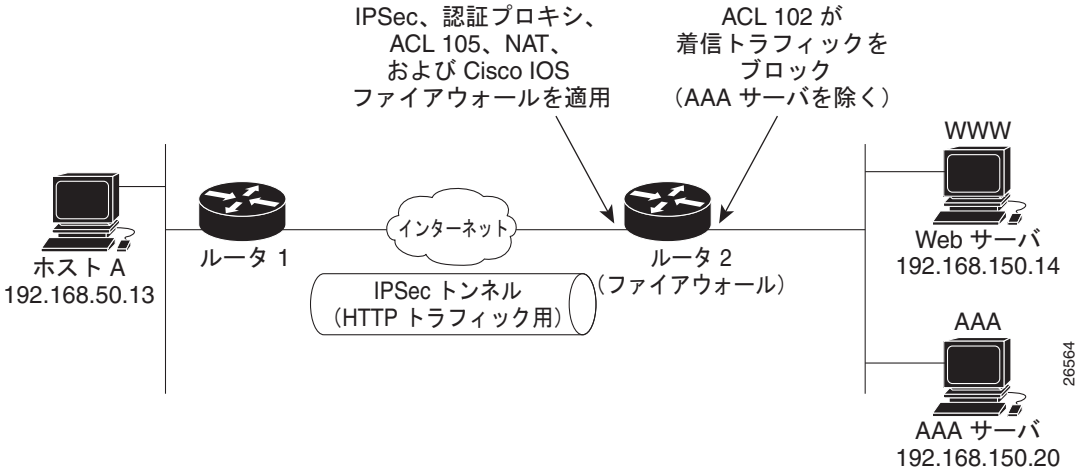
```
! Configure Router 2 as the firewall, using the authentication proxy, IPSec, and CBAC.
version 12.0
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router2
!
logging buffered 4096 debugging
aaa new-model
aaa authentication login default group tacacs
aaa authentication login console_line none
aaa authentication login special none
aaa authentication ppp default group tacacs
aaa authorization exec default group tacacs
! Configure AAA for the authentication proxy.
aaa authorization auth-proxy default group tacacs+
enable password junk
!
! Create the CBAC inspection rule HTTP_TEST.
ip inspect name rule22 http
ip inspect name rule22 tcp
ip inspect name rule22 ftp
ip inspect name rule22 smtp
!
! Create the authentication proxy rule PXY.
ip auth-proxy name pxy http
! Turn on display of the router name in the authentication proxy login page.
ip auth-proxy auth-proxy-banner
ip audit notify log
ip audit po max-events 100
!
! Configure IPSec.
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco1234 address 10.0.0.1
!
crypto ipsec transform-set rule_1 ah-sha-hmac esp-des esp-sha-hmac
!
 crypto map testtag 10 ipsec-isakmp
 set peer 10.0.0.1
 set transform-set rule_1
 match address 155
!
! Apply the CBAC inspection rule and the authentication proxy rule at interface
! Serial0/0.
interface Serial0/0
 ip address 10.0.0.2 255.0.0.0
 ip access-group 105 in
 no ip directed-broadcast
 ip inspect rule22 in
 ip auth-proxy pxy
 encapsulation ppp
 no ip route-cache
 no ip mroute-cache
 no keepalive
 no fair-queue
crypto map testtag
!
interface Ethernet0/1
 ip address 192.168.123.2 255.255.255.0
```

```
 ip access-group 102 in
 no ip directed-broadcast
 ip route-cache
 no ip mroute-cache
!
no ip classless
ip route 192.168.23.0 255.255.255.0 10.0.0.1
ip route 192.168.50.0 255.255.255.0 16.0.0.1
! Configure the HTTP server.
ip http server
ip http access-class 15
ip http authentication aaa
!
! Create ACL 15 to block all traffic for the http server.
access-list 15 deny any
! Create ACL 102 to block all traffic inbound on interface Ethernet0/1 except for
! traffic from the AAA server.
access-list 102 permit tcp host 192.168.123.20 eq tacacs host 192.168.123.2
access-list 102 deny   tcp any any
access-list 102 deny   udp any any
access-list 102 permit ip any any
! Create ACL 105 to block all traffic inbound on interface Serial0/0. Permit only IP
! protocol traffic.
access-list 105 deny   tcp any any
access-list 105 deny   udp any any
access-list 105 permit ip any any
! Identify the IPSec specific traffic.
access-list 155 permit tcp host 192.168.123.14 host 192.168.23.13 eq www
access-list 155 permit tcp host 192.168.123.14 eq www host 192.168.23.13
!
! Define the AAA server host and encryption key.
tacacs-server host 192.168.123.14
tacacs-server key cisco
!
line con 0
 exec-timeout 0 0
 login authentication special
 transport input none
line aux 0
 transport input all
 speed 38400
 flowcontrol hardware
line vty 0 4
 password lab
```

認証プロキシ、IPSec、NAT、および CBAC の設定例

次の例は、認証プロキシ、IPSec、NAT および CBAC 機能を使用するルータ設定を示します。図 7 に、設定を示します。

図 7　　認証プロキシ、IPSec、および CBAC の設定例



この例では、ホスト A が Web サーバ（WWW）との HTTP 接続を開始します。ルータ 1（インターフェイス BRI0）とルータ 2（インターフェイス Serial2）の間の HTTP トラフィックは、IPSec を使用して暗号化されます。認証プロキシは、ファイアウォールとして動作するルータ 2 で設定されます。認証プロキシ、NAT、および CBAC は、インターフェイス Serial2 で設定され、ファイアウォールとして機能しています。ACL 105 は、インターフェイス Serial2 ですべてのトラフィックをブロックします。ACL 102 は、ルータ 2 上のインターフェイス Ethernet0 に適用され、AAA サーバからのトラフィックを除くそのインターフェイス上のすべてのトラフィックをブロックします。この例で、認証プロキシは標準の ACL 10 を使用して、認証プロキシ機能を使用するホストを指定しています。

ACL 10 内のいずれかのホストが Web サーバとの HTTP 接続を開始すると、認証プロキシは、そのホストのユーザに対し、ユーザ名とパスワードの入力を求めます。これらのクレデンシャルは、認証および許可のために AAA サーバで検証されます。認証が正常に行われると、ユーザごとの ACL がファイアウォールにダウンロードされ、サービスが許可されます。

次の例では、完全を期すためにルータ 1 とルータ 2 の両方の設定を示します。

- 「ルータ 1 の設定例」
- 「ルータ 2 の設定例」

ルータ 1 の設定例

```
! Configure router 1 for IPSec.
version 12.0
 service timestamps debug uptime
 service timestamps log uptime
 no service password-encryption
 !
 hostname Router1
 !
 logging buffered 4096 debugging
 no logging console
```

```
 !
isdn switch-type basic-5ess
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco1234 address 16.0.0.2
 crypto ipsec transform-set rule_1 ah-sha-hmac esp-des esp-sha-hmac
 !
  !
  crypto map testtag 10 ipsec-isakmp
  set peer 16.0.0.2
  set transform-set rule_1
  match address 155
 !
 !
 process-max-time 200
 !
 interface BRI0
  ip address 16.0.0.1 255.0.0.0
  no ip directed-broadcast
  encapsulation ppp
  dialer idle-timeout 5000
  dialer map ip 16.0.0.2 name router2 broadcast 50006
  dialer-group 1
  isdn switch-type basic-5ess
  crypto map testtag
 !
 interface FastEthernet0
  ip address 192.168.50.2 255.255.255.0
  no ip directed-broadcast
 !
 ip classless
 ip route 192.168.150.0 255.255.255.0 16.0.0.2
 no ip http server
! Identify the IPSec specific traffic.
 access-list 155 permit tcp host 192.168.50.13 host 192.168.150.100 eq www
 access-list 155 permit tcp host 192.168.50.13 eq www host 192.168.150.100
 dialer-list 1 protocol ip permit
 !
 line con 0
  exec-timeout 0 0
  transport input none
 line aux 0
 line vty 0 4
  password lab
  login
```

ルータ 2 の設定例

```
! Configure router 2 as the firewall, using the authentication proxy, IPSec, NAT, and
! CBAC.
 version 12.0
 service timestamps debug uptime
 service timestamps log uptime
 no service password-encryption
 !
 hostname router2
 !
 logging buffered 4096 debugging
 aaa new-model
 aaa authentication login default group tacacs+
 aaa authentication login console_line none
```

```
 aaa authorization exec default group tacacs+
 ! Configure AAA for the authentication proxy.
 aaa authorization auth-proxy default group tacacs+
!
 ! Create the CBAC inspection rule "rule44."
 ip inspect name rule44 http java-list 5
 ip inspect name rule44 tcp
 ip inspect name rule44 ftp
 ip inspect name rule44 smtp
 !
 ! Create the authentication proxy rule "pxy." Set the timeout value for rule
 ! pxy to three minutes. Standard ACL 10 is applied to the rule.
 ip auth-proxy name pxy http list 10 auth-cache-time 3
 isdn switch-type primary-5ess
 !
 ! Configure IPSec.
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco1234 address 16.0.0.1
 !
 !
 crypto ipsec transform-set rule_1 ah-sha-hmac esp-des esp-sha-hmac
 !
  !
  crypto map testtag 10 ipsec-isakmp
  set peer 16.0.0.1
  set transform-set rule_1
  match address 155
!
 controller T1 2/0
  framing esf
  linecode b8zs
  pri-group timeslots 1-24
!
! Apply ACL 102 inbound at interface Ethernet0/1 and configure NAT.
interface Ethernet0/1
 ip address 192.168.150.2 255.255.255.0
 ip access-group 102 in
 no ip directed-broadcast
 ip nat inside
 no ip mroute-cache
!
! Apply the authentication proxy rule PXY, CBAC inspection rule HTTP_TEST, NAT, and
! and ACL 105 at interface Serial2/0:23.
interface Serial2/0:23
 ip address 16.0.0.2 255.0.0.0
 ip access-group 105 in
 no ip directed-broadcast
 ip nat outside
 ip inspect rule44 in
 ip auth-proxy pxy
 encapsulation ppp
 ip mroute-cache
 dialer idle-timeout 5000
 dialer map ip 16.0.0.1 name router1 broadcast 71011
 dialer-group 1
 isdn switch-type primary-5ess
 fair-queue 64 256 0
 crypto map testtag
!
! Use NAT to translate the Web server address.
ip nat inside source static 192.168.150.14 192.168.150.100
ip classless
ip route 192.168.50.0 255.255.255.0 16.0.0.1
```

```
! Configure the HTTP server.
ip http server
ip http access-class 15
ip http authentication aaa
!
! Create standard ACL 5 to specify the list of hosts from which to accept java applets.
! ACL 5 is used to block Java applets in the CBAC inspection rule named “rule44,” which
! is applied at interface Serial2/0:23.
access-list 5 permit any
! Create standard ACL 10 to specify the hosts using the authentication proxy. This ACL
! used in the authentication proxy rule named “PXY”, which is applied at interface
! Serial2/0:23.
access-list 10 permit any
! Create ACL 15 to block all traffic for the http server.
access-list 15 deny any
! Create extended ACL 102 to block all traffic inbound on interface Ethernet0/1
! except for traffic from the AAA server.
access-list 102 permit tcp host 192.168.150.20 eq tacacs 192.168.150.2
access-list 102 deny   tcp any any
access-list 102 deny   udp any any
access-list 102 permit ip any any
! Create extended ACL 105 to block all TCP and UDP traffic inbound on interface
! Serial2/0:23.
access-list 105 deny   tcp any any
access-list 105 deny   udp any any
access-list 105 permit ip any any
! Identify the IPSec specific traffic.
access-list 155 permit tcp host 192.168.150.100 host 192.168.50.13 eq www
access-list 155 permit tcp host 192.168.150.100 eq www host 192.168.50.13
dialer-list 1 protocol ip permit
! Define the AAA server host and encryption key.
tacacs-server host 192.168.126.14
tacacs-server key cisco
!
line con 0
 exec-timeout 0 0
! Define the AAA server host and encryption key.
 login authentication console_line
 transport input none
 line aux 0
 line vty 0 4
 password lab
!
!
 end
```

AAA サーバのユーザ プロファイル例

ここでは、AAA サーバでの認証プロキシのユーザ プロファイル エントリの例を示します。「proxyacl」エントリを使用して、ユーザのアクセス権限を定義します。ユーザが認証プロキシを使用してログインに成功すると、これらのエントリはファイアウォール ルータに転送されます。プロファイル内の各エントリにはサービスまたはアプリケーションの「permit」アクセスを指定する必要があります。各エントリの送信元アドレスは、「any」に設定します。アドレスは、プロファイルがファイアウォールにダウンロードされるときに認証ホストの IP アドレスに置換されます。すべての AAA ユーザの特権レベルは 15 に設定する必要があります。

ここでは、次の各手順について説明します。

- 「CiscoSecure ACS 2.3 for Windows NT」

- 「CiscoSecure ACS 2.3 for UNIX」
- 「TACACS+ Server」
- 「Livingston Radius Server」
- 「Ascend Radius Server」

CiscoSecure ACS 2.3 for Windows NT

ここでは、CiscoSecure ACS 2.3 for Windows NT 上で認証プロキシを設定する方法について説明します。CiscoSecure ACS の詳細については、該当する製品のマニュアルを参照してください。

次の設定例は、CiscoSecure ACS for Windows NT の TACACS+ サービス用の設定です。

ステップ 1 [Interface Configuration] アイコンをクリックし、[TACACS+ (Cisco)] をクリックします。

a. 下にスクロールして [New Services] を表示します。

b. 新しいサービス「auth-proxy」を [Service] フィールドに追加します。[Protocol] フィールドは空のままにします。

c. 新しいサービスに対して [User] チェックボックスと [Group] チェックボックスをオンにします。

d. 下にスクロールして [Advance Configuration Options] を表示し、[Per-user Advance TACACS+] 機能をオンにします。

e. [Submit] をクリックします。

ステップ 2 [Network Configuration] アイコンをクリックします。

a. [Network Access Servers] の [Add Entry] アイコンをクリックし、[Network Access Server Hostname]、IP アドレス、キー（ルータで設定したキー）のフィールドに情報を入力します。

b. [Authenticate Using] オプションに対して [TACACS+ (Cisco)] を選択します。

c. [Submit + Restart] アイコンをクリックします。

ステップ 3 [Group Setup] アイコンをクリックします。

a. ドロップダウン メニューからユーザ グループを選択します。

b. [Users in Group] チェックボックスをオンにします。

c. ユーザ リストからユーザを選択します。

d. [User Setup] リストで下にスクロールし、[TACACS+ Settings] を表示して、「auth-proxy」チェックボックスをオンにします。

e. [Custom Attributes] チェックボックスをオンにします。

f. プロファイル エントリを追加し（エントリは単一引用符または二重引用符で囲みません）、特権レベルを 15 に設定します。

```
priv-lvl=15
proxyacl#1=permit tcp any any eq 26
proxyacl#2=permit icmp any host 60.0.0.2
proxyacl#3=permit tcp any any eq ftp
proxyacl#4=permit tcp any any eq ftp-data
proxyacl#5=permit tcp any any eq smtp
proxyacl#6=permit tcp any any eq telnet
```

g. [Submit] をクリックします。

ステップ 4 [User Setup] アイコンをクリックします。

a. [List All Users] をクリックします。

b. ユーザ名を追加します。

c. 下にスクロールして [User Setup Password Authentication] を表示します。

d. [Password Authentication] ドロップダウン メニューから [Select SDI SecurID Token Card] を選択します。

e. 以前設定したユーザ グループ 1 を選択します。

f. [Submit] をクリックします。

ステップ 5 再度 [Group Setup] アイコンをクリックします。

a. ユーザ グループ 1 を選択します。

b. [Users in Group] をクリックします。

c. [Edit Settings] をクリックします。

d. [Submit + Restart] アイコンをクリックして、最新の設定を更新し、AAA サーバに送信します。

CiscoSecure ACS 2.3 for UNIX

ここでは、CiscoSecure ACS 2.3 for UNIX 上で認証プロキシを設定する方法について説明します。CiscoSecure ACS の詳細については、該当する製品のマニュアルを参照してください。

Administrator プログラムを使用して CiscoSecure ACS を管理するには、Java と JavaScript をサポートする Web ブラウザが必要です。ブラウザ アプリケーションで Java をイネーブルにする必要があります。Java ベースの CiscoSecure Administrator の詳細設定プログラムは、CiscoSecure ACS Administrator のどの Web ページからでも起動できます。

次に、CiscoSecure ACS 2.3 for UNIX の TACACS+ サービスの設定手順の例を示します。

ステップ 1 CiscoSecure ACS Web インターフェイスの CiscoSecure ACS Web メニュー バーで、[Advanced] をクリックし、再度 [Advanced] をクリックします。

Java ベースの CiscoSecure Administrator 詳細設定プログラムが表示されます。ロードに数分かかることがあります。

ステップ 2 CiscoSecure Administrator 詳細設定プログラムで、タブ化された [Members] ページの [Navigator] ペインで [Browse] をオフにします。

これにより [Create New Profile] アイコンが表示されます。

ステップ 3 [Navigator] ペインで、次のいずれかを実行します。

- ユーザを追加するグループを探してクリックします。
- ユーザをグループに追加しない場合は、[Root] フォルダ アイコンをクリックします。

ステップ 4 [Create Profile] をクリックして、[New Profile] ダイアログ ボックスを表示します。

ステップ 5 [Group] チェックボックスがオフになっていることを確認します。

ステップ 6 作成するユーザの名前を入力し、[OK] をクリックします。新しいユーザがツリーに表示されます。

ステップ 7 タブ化された [Members] ページの [Navigator] ペインに表示されるツリー内の、グループ プロファイルまたはユーザ プロファイルのアイコンをクリックします。

ステップ 8 必要に応じて、[Profile] ペインで [Profile] アイコンをクリックしてペインを展開します。

選択したプロファイルまたはサービスに該当するアトリビュートが含まれるリストまたはダイアログ ボックスが、画面右下のウィンドウに表示されます。このウィンドウの情報は、[Profile] ペインで選択した内容に応じて変化します。

ステップ 9 [Service-String] をクリックします。

ステップ 10 [string] をクリックし、テキスト フィールドに「**auth-proxy**」と入力し、[Apply] をクリックします。

ステップ 11 [Option] メニューを選択します。

ステップ 12 [Option] メニューで、[Default Attributes] をクリックします。

ステップ 13 アトリビュートを [Deny] から [Permit] に変更します。

ステップ 14 [Apply] をクリックします。

ステップ 15 [Option] メニューで、[Attribute] をクリックし、テキスト フィールドに特権レベルを入力します。

```
priv-lvl=15
```

ステップ 16 [Option] メニューで、[Attribute] をクリックし、テキスト フィールドに [proxyacl] エントリを入力します。

```
proxyacl#1=”permit tcp any any eq 26”
```

追加する各サービスまたはプロトコルに対してこのステップを繰り返します。

```
proxyacl#2=”permit icmp any host 60.0.0.2”
proxyacl#3=”permit tcp any any eq ftp”
proxyacl#4=”permit tcp any any eq ftp-data”
proxyacl#5=”permit tcp any any eq smtp”
proxyacl#6=”permit tcp any any eq telnet”
```

ステップ 17 すべての変更を終えたら、[Submit] をクリックします。

TACACS+ Server

```
default authorization = permit
key = cisco
user = Brian {
login = cleartext cisco
service = auth-proxy
 {
  priv-lvl=15
  proxyacl#1="permit tcp any any eq 26"
  proxyacl#2="permit icmp any host 60.0.0.2"
  proxyacl#3="permit tcp any any eq ftp"
  proxyacl#4="permit tcp any any eq ftp-data"
  proxyacl#5="permit tcp any any eq smtp"
  proxyacl#6="permit tcp any any eq telnet"
 }
}
```

Livingston Radius Server

```
Bob Password = "cisco" User-Service-Type=Outbound-User
cisco-avpair = "auth-proxy:priv-lvl=15",
cisco-avpair = "auth-proxy:proxyacl#1=permit tcp any any eq 26",
cisco-avpair = "auth-proxy:proxyacl#2=permit icmp any host 60.0.0.2",
cisco-avpair = "auth-proxy:proxyacl#3=permit tcp any any eq ftp",
cisco-avpair = "auth-proxy:proxyacl#4=permit tcp any any eq ftp-data",
cisco-avpair = "auth-proxy:proxyacl#5=permit tcp any any eq smtp",
cisco-avpair = "auth-proxy:proxyacl#6=permit tcp any any eq telnet"
```

Ascend Radius Server

```
Alice Password = "cisco" User-Service = Dialout-Framed-User
cisco-avpair = "auth-proxy:priv-lvl=15",
cisco-avpair = "auth-proxy:proxyacl#1=permit tcp any any eq 26",
cisco-avpair = "auth-proxy:proxyacl#2=permit icmp any host 60.0.0.2",
cisco-avpair = "auth-proxy:proxyacl#3=permit tcp any any eq ftp",
cisco-avpair = "auth-proxy:proxyacl#4=permit tcp any any eq ftp-data",
cisco-avpair = "auth-proxy:proxyacl#5=permit tcp any any eq smtp",
cisco-avpair = "auth-proxy:proxyacl#6=permit tcp any any eq telnet"
```

その他の参考資料

ここでは、認証プロキシ機能に関する関連資料について説明します。

関連資料

内容	参照先
認可	「Configuring Authorization」
認証	「Configuring Authentication」
アカウンティング	「Configuring Accounting」
RADIUS	「Configuring RADIUS」
TACACS+	「Configuring TACACS+」

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能による既存規格のサポートに変更はありません。	—

MIB

MIB	MIB リンク
この機能によってサポートされる新しい MIB または変更された MIB はありません。またこの機能による既存 MIB のサポートに変更はありません。	選択したプラットフォーム、Cisco IOS リリース、および機能セットの MIB を検索してダウンロードする場合は、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

RFC

RFC	タイトル
この機能によってサポートされる新しい RFC や変更された RFC はありません。	—

シスコのテクニカル サポート

説明	リンク
右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。 以下を含むさまざまな作業にこの Web サイトが役立ちます。 ・テクニカル サポートを受ける ・ソフトウェアをダウンロードする ・セキュリティの脆弱性を報告する、またはシスコ製品のセキュリティ問題に対する支援を受ける ・ツールおよびリソースへアクセスする - Product Alert の受信登録 - Field Notice の受信登録 - Bug Toolkit を使用した既知の問題の検索 ・Networking Professionals（NetPro）コミュニティで、技術関連のディスカッションに参加する ・トレーニング リソースへアクセスする ・TAC Case Collection ツールを使用して、ハードウェアや設定、パフォーマンスに関する一般的な問題をインタラクティブに特定および解決する この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/techsupport

認証プロキシの機能情報

表 3 に、この機能のリリース履歴を示します。

ご使用の Cisco IOS ソフトウェア リリースによっては、コマンドの中に一部使用できないものがあります。特定のコマンドに関するリリース情報については、コマンド リファレンス マニュアルを参照してください。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェア イメージのサポート情報を検索できます。Cisco Feature Navigator を使用すると、特定のソフトウェア リリース、機能セット、またはプラットフォームをサポートする Cisco IOS ソフトウェア イメージおよび Catalyst OS ソフトウェア イメージを確認できます。Cisco Feature Navigator には、http://tools.cisco.com/ITDIT/CFN/jsp/index.jsp からアクセスできます。Cisco.com のアカウントは必要ありません。

（注） 表 3 には、一連の Cisco IOS ソフトウェア リリースのうち、特定の機能が初めて導入された Cisco IOS ソフトウェア リリースだけが記載されています。特に明記していないかぎり、その機能は、一連の Cisco IOS ソフトウェア リリースの以降のリリースでもサポートされます。

表 3　認証を設定するための機能情報

機能名	リリース	機能情報
認証プロキシ	12.1(5)T	Cisco IOS Firewall 認証プロキシ機能では、動的かつユーザごとの認証と認可、業界標準の TACACS+ および RADIUS 認証プロトコルを使用したユーザの認証が可能です。ユーザによる接続の認証と認可により、ネットワーク攻撃に対するより強力な保護が可能になります。 この機能は、12.1(5)T で Cisco IOS に導入されました。