ネットワーク アクセスに対する AAA 規則の設定

この章では、ネットワーク アクセスに対して AAA（「トリプル エー」と発音）をイネーブルにする方法について説明します。

管理アクセスの AAA については、一般的な操作のコンフィギュレーション ガイドの“Configuring AAA for System Administrators” section on page 41-15 を参照してください。

この章は、次の項で構成されています。

AAA のパフォーマンス

ASA は「カットスルー プロキシ」を使用します。これにより、従来のプロキシ サーバと比較して、パフォーマンスが大幅に向上します。従来のプロキシ サーバは、OSI モデルのアプリケーション レイヤですべてのパケットを分析するため、プロキシ サーバのパフォーマンスに負担がかかります。ASA カットスルー プロキシは、アプリケーション層で最初にユーザ確認を行い、続いて標準 AAA サーバまたはローカル データベースで認証します。ASA はユーザを認証した後、セッション フローをシフトするため、セッション ステート情報を維持したまま、すべてのトラフィックが送信元と宛先の間で直接かつ迅速に流れます。

AAA ルールのライセンス要件

次の表に、この機能のライセンス要件を示します。

モデル	ライセンス要件
すべてのモデル	基本ライセンス

ガイドラインと制限事項

この項では、この機能のガイドラインと制限事項について説明します。

コンテキスト モードのガイドライン

シングル コンテキスト モードとマルチ コンテキスト モードでサポートされています。

ファイアウォール モードのガイドライン

ルーテッド ファイアウォール モードとトランスペアレント ファイアウォール モードでサポートされています。

IPv6 のガイドライン

IPv6 をサポートします。

その他のガイドライン

クラスタリングでは、この機能は、マスター ユニットでのみサポートされます。

ネットワーク アクセス認証の設定

この項は、次の内容で構成されています。

認証について

ASA では、AAA サーバを使用するネットワーク アクセス認証を設定できます。この項は、次の内容で構成されています。

一度だけの認証

所定の IP アドレスのユーザは、認証セッションが期限切れになるまで、すべてのルールおよびタイプに対して一度だけ認証を受ける必要があります（タイムアウトの値については、コマンド リファレンスで **timeout uauth** コマンドを参照してください）。たとえば、Telnet および FTP を認証するように ASA が設定されていて、ユーザが正常に Telnet 認証を受けた場合、認証セッションが継続している限り、ユーザは FTP 認証を受ける必要はありません。

認証確認を受けるために必要なアプリケーション

プロトコルまたはサービスへのネットワーク アクセス認証を要求するように ASA を設定することは、すべてのプロトコルまたはサービスについて可能ですが、ユーザが直接認証を受けることができるのは、HTTP、HTTPS、Telnet、または FTP だけです。ユーザがこれらのサービスのいずれかの認証を受けないと、ASA は認証が必要な他のトラフィックを許可しません。

次のように、ASA が AAA 用にサポートしている認証ポートは固定値です。

- ポート 21 は FTP 用
- ポート 23 は Telnet 用
- ポート 80 は HTTP 用
- ポート 443 は HTTPS 用

ASA の認証プロンプト

Telnet および FTP の場合、ASA は認証プロンプトを生成します。

HTTP の場合、ASA はデフォルトで基本 HTTP 認証を使用し、認証プロンプトを提供します。ユーザがユーザ名とパスワードを入力できる内部 Web ページにユーザをリダイレクトするように ASA を設定することもできます（**aaa authentication listener** コマンドで設定します）。

HTTPS の場合、ASA はカスタム ログイン画面を生成します。ユーザがユーザ名とパスワードを入力できる内部 Web ページにユーザをリダイレクトするように ASA を設定することもできます（**aaa authentication listener** コマンドで設定します）。

リダイレクションは、基本方式を強化したものです。これは、認証時に向上したユーザ エクスペリエンスが提供されると同時に、Easy VPN でもファイアウォール モードでも、HTTP および HTTPS と同じユーザ エクスペリエンスが提供されるためです。また、ASA での直接認証もサポートしています。

次の理由で、基本 HTTP 認証を引き続き使用しなければならない場合があります。

- ASA でリスニング ポートを開きたくない。
- ルータ上で NAT を使用しており、ASA によって提供される Web ページのトランスレーション ルールを作成したくない。
- 基本 HTTP 認証の方がネットワークで有効に機能する見込みがある。

たとえば、電子メールに URL が埋め込まれている場合などのように、ブラウザ以外のアプリケーションでは基本認証の方が適していることがあります。

正常に認証されると、ASA により元の宛先にリダイレクトされます。宛先サーバにも独自の認証がある場合、ユーザは別のユーザ名とパスワードを入力します。基本 HTTP 認証を使用していて、宛先サーバ用に別のユーザ名とパスワードを入力する必要がある場合は、**virtual http** コマンドを設定する必要があります。

（注）　HTTP 認証を使用する場合、デフォルトでクリア テキストのユーザ名とパスワードがクライアントから ASA に送信されます。さらにこのユーザ名とパスワードは宛先 Web サーバにも送信されます。クレデンシャルの保護の詳細については、「Web クライアントのセキュアな認証のイネーブル化」（P.7-9）を参照してください。

FTP の場合、ASA ユーザ名、アット マーク（@）、FTP ユーザ名（name1@name2）を入力するオプションがあります。パスワードには、ASA パスワード、アット マーク（@）、FTP パスワード（password1@password2）を入力します。たとえば、次のテキストを入力します。

```
name> name1@name2
password> password1@password2
```

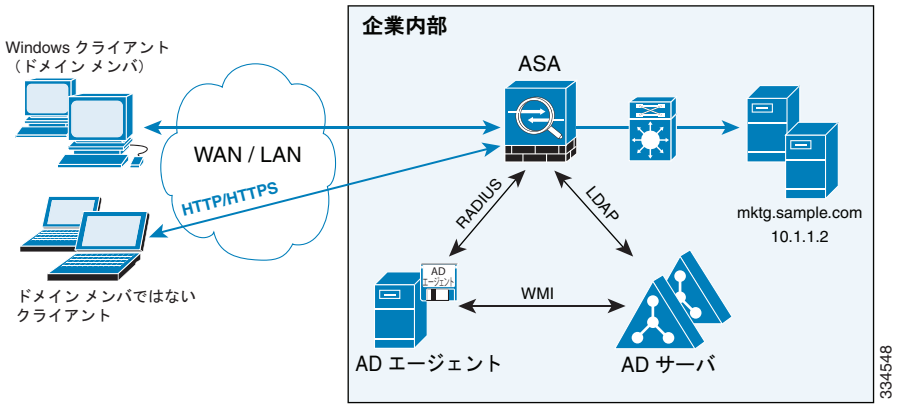
この機能は、複数のログインを必要とするファイアウォールをカスケード接続している場合に有用です。複数の名前およびパスワードは、複数のアット マーク（@）で区切ることができます。

AAA プロンプトとアイデンティティ ファイアウォール

企業では、ユーザによっては、Web ポータル（カットスルー プロキシ）を使用した認証など、通常とは異なる認証メカニズムを使用してネットワークにログインする場合があります。たとえば、クライアントとして Mac や Linux を使用しているユーザは、Web ポータル（カットスルー プロキシ）にログインすることがあります。そのため、これらの認証方式がアイデンティティに基づくアクセス ポリシーと連携できるようにアイデンティティ ファイアウォールを設定する必要があります。

図 7-1 は、カットスルー プロキシ認証キャプティブ ポータルをサポートする展開方法を示しています。Active Directory サーバと AD エージェントはメイン サイトの LAN 上に配置されています。ただし、アイデンティティ ファイアウォールは、Active Directory ドメインに含まれないクライアントの認証をサポートするよう設定されています。

図 7-1　　カットスルー プロキシ認証をサポートする展開



ASA は、Web ポータル（カットスルー プロキシ）経由でログインするユーザを認証が行われる Active Directory ドメインに属するユーザと見なします。

ASA は、Web ポータル（カットスルー プロキシ）によってログインしたユーザを AD エージェントに報告し、AD エージェントがユーザ情報を登録されているすべての ASA デバイスに配布します。この場合は、アイデンティティ ファイアウォールは、Active Directory ドメインとユーザを関連付けることができます。具体的には、認証されたユーザのユーザ アイデンティティと IP アドレスのマッピングが、パケットを受信して認証する入力インターフェイスを含むすべての ASA コンテキストに転送されます。

ユーザは、HTTP/HTTPS、FTP、Telnet、または SSH を使用してログインできます。ユーザがこれらの認証方式でログインする場合は、次のガイドラインが適用されます。

- HTTP/HTTPS トラフィックの場合、認証されていないユーザには認証ウィンドウが表示されます。
- Telnet および FTP トラフィックの場合、ユーザはカットスルー プロキシ サーバ経由でログインし、さらに Telnet および FTP サーバにログインする必要があります。
- ユーザは、ログイン クレデンシャル（形式は domain\username）を入力するときに、Active Directory ドメインを指定できます。ASA は、指定されたドメインに関連付けられた AAA サーバ グループを自動的に選択します。
- ユーザがログイン クレデンシャル（形式は domain\username）を入力するときに Active Directory ドメインを指定すると、ASA はドメインを解析し、それを使用して、アイデンティティ ファイアウォール用に設定された AAA サーバから認証サーバを選択します。AAA サーバには username だけが渡されます。
- ログイン クレデンシャルにバックスラッシュ（\）デリミタが含まれていない場合、ASA はドメインを解析せず、アイデンティティ ファイアウォールに設定されたデフォルト ドメインに対応する AAA サーバで認証が実行されます。
- デフォルト ドメインが設定されていない場合、またはそのデフォルト ドメインにサーバ グループが設定されていない場合、ASA は認証を拒否します。
- ドメインが指定されない場合、ASA はアイデンティティ ファイアウォールに設定されたデフォルト ドメインの AAA サーバ グループを選択します。

バックアップ認証方式としての AAA ルール

認証ルール（「カットスルー プロキシ」とも呼ばれます）は、ユーザに基づいてネットワーク アクセスを制御します。この機能がアクセス ルールとアイデンティティ ファイアウォールに非常に似ているため、AAA ルールは、ユーザの AD ログインの期限が切れるか、有効なユーザが AD にまだログインしていない場合、認証のバックアップ方式として使用できます。たとえば、有効なログインのないユーザの場合、AAA ルールをトリガーできます。AAA ルールが有効なログインがないユーザに対してだけトリガーされるようにするには、拡張 ACL でアクセス ルールと AAA ルールに使用される特別なユーザ名 None（有効なログインのないユーザ）および Any（有効なログインを持つユーザ）を指定します。アクセス ルールでは、ユーザおよびグループのポリシーを通常どおりに設定しますが、any any を拒否する前にすべての None ユーザを許可するルールを含めます。これらのユーザが後で AAA ルールをトリガーできるように、これらのユーザを許可する必要があります。次に、Any ユーザ（これらのユーザは、AAA ルールの対象ではなく、アクセス ルールによってすでに処理されています）を照合せず、すべての None ユーザのみを照合する AAA ルールを設定して、None ユーザに対する AAA 認証をトリガーします。ユーザがカットスルー プロキシによって正常にログインした後、トラフィックは再び正常に流れます。

スタティック PAT および HTTP

HTTP 認証では、スタティック PAT が設定されている場合、ASA は実際のポートをチェックします。ASA は、マッピング ポートにかかわらず、実際のポート 80 を宛先とするトラフィックを検出した場合、HTTP 接続を代行受信し、認証を実行します。

たとえば、外部 TCP ポート 889 が次のようにポート 80 に変換されていて、関係するすべての ACL でこのトラフィックが許可されているとします。

```
object network obj-192.168.123.10-01
   host 192.168.123.10
   nat (inside,outside) static 10.48.66.155 service tcp 80 889
```

この場合、ユーザはポート 889 で 10.48.66.155 にアクセスを試み、ASA はそのトラフィックを代行受信して、HTTP 認証を実行します。ASA が HTTP 接続の完了を許可する前に、ユーザの Web ブラウザには HTTP 認証ページが表示されます。

次の例のように、ローカル ポートがポート 80 ではないとします。

```
object network obj-192.168.123.10-02
    host 192.168.123.10
    nat (inside,outside) static 10.48.66.155 service tcp 111 889
```

この場合、ユーザには認証ページは表示されません。代わりに、ASA は Web ブラウザにエラー メッセージを送信して、要求されたサービスを使用する前にユーザが認証を受ける必要があることを通知します。

ネットワーク アクセス認証の設定

ネットワーク アクセス認証を設定するには、次の手順を実行します。

	コマンド	目的
ステップ 1	**aaa-server** **例：** `hostname(config)# aaa-server AuthOutbound protocol tacacs+`	AAA サーバを指定します。すでに指定済みの場合は、次の手順に進みます。AAA サーバの指定方法の詳細については、一般的な操作のコンフィギュレーション ガイドの “Configuring AAA Server Groups” section on page 33-11 を参照してください。
ステップ 2	**access-list** *access_list_name* **extended** {**deny** | **permit**} {**tcp** | **udp**} [*user_argument*] [*security_group_argument*] *source_address_argument* [*port_argument*] [*security_group_argument*] *dest_address_argument* [*port_argument*] **例：** `hostname(config)# access-list MAIL_AUTH extended permit tcp any any eq smtp`	認証するトラフィックの送信元アドレスと宛先アドレスを指定する ACL を作成します。ここに示される構文は単なる例です。詳細については、一般的な操作のコンフィギュレーション ガイドの Chapter 19, “Adding an Extended Access Control List,” を参照してください。 ACL でアイデンティティ ファイアウォールの引数を指定した場合、ACL の次のキーワードは、AAA ルールにのみ関連します。キーワード **user-group any** および **user-group none** を指定することにより、カットスルー プロキシ認証をサポートできます。 • **any**：ACL は、すべてのユーザに関連付けられている任意の IP アドレスと一致します。 • **none**：ACL は、どの IP アドレスにも関連付けられていない任意の IP アドレスと一致します。

	コマンド	目的
ステップ 3	**aaa authentication match** *acl_name interface_name server_group* [**user-identity**] **例：** hostname(config)# aaa authentication match MAIL_AUTH inside AuthOutbound	認証を設定します。 *acl_name* 引数は、Step 2 で作成した ACL の名前です。*interface_name* 引数は、**nameif** コマンドで指定したインターフェイスの名前です。*server_group* 引数は、Step 1 で作成した AAA サーバ グループです。 **（注）** もう 1 つの方法として、**aaa authentication include** コマンド（コマンド内でトラフィックを指定するコマンド）を使用することもできます。ただし、同一コンフィギュレーション内で両方の方法を使用することはできません。詳細については、コマンド リファレンスを参照してください。 **user-identity** キーワードは、アイデンティティ ファイアウォールに対して認証を照合します。
ステップ 4	**aaa authentication listener http**[**s**] *interface_name* [**port** *portnum*] **redirect** **例：** hostname(config)# aaa authentication listener http inside redirect	（任意）HTTP または HTTPS 接続の認証のリダイレクション方式をイネーブルにします。 引数 *interface_name* は、リスニング ポートをイネーブルにするインターフェイスです。**port** *portnum* 引数で、ASA がリッスンするポート番号を指定します。デフォルトは 80（HTTP）と 443（HTTPS）です。 任意のポート番号を使用して同じ機能を保持できますが、直接認証ユーザがそのポート番号を認識している必要があります。これは、リダイレクトされたトラフィックは正しいポート番号に自動的に送信されますが、直接認証するユーザは、ポート番号を手動で指定する必要があるためです。 このコマンドを HTTP と HTTPS について別々に入力します。
ステップ 5	**aaa local authentication attempts max-fail** *number* **例：** hostname(config)# aaa local authentication attempts max-fail 7	（任意）ネットワーク アクセス認証にローカル データベースを使用し、ASA が所定のユーザ アカウントを許可するローカル ログイン連続失敗試行回数を制限します（特権レベル 15 のユーザは除きます。この機能はレベル 15 のユーザには影響はありません）。*number* 引数の値は 1 ～ 16 です。 **ヒント** 特定のユーザまたはすべてのユーザのロックアウト ステータスを解除するには、**clear aaa local user lockout** コマンドを使用します。

例

次の例では、すべての内部 HTTP トラフィックおよび SMTP トラフィックを認証します。

```
hostname(config)# aaa-server AuthOutbound protocol tacacs+
hostname(config-aaa-server-group)# exit
hostname(config)# aaa-server AuthOutbound (inside) host 10.1.1.1
hostname(config-aaa-server-host)# key TACPlusUauthKey
```

```
hostname(config-aaa-server-host)# exit
hostname(config)# access-list MAIL_AUTH extended permit tcp any any eq smtp
hostname(config)# access-list MAIL_AUTH extended permit tcp any any eq www
hostname(config)# aaa authentication match MAIL_AUTH inside AuthOutbound
hostname(config)# aaa authentication listener http inside redirect
```

次の例では、外部インターフェイスから特定のサーバ（209.165.201.5）への Telnet トラフィックを認証します。

```
hostname(config)# aaa-server AuthInbound protocol tacacs+
hostname(config-aaa-server-group)# exit
hostname(config)# aaa-server AuthInbound (inside) host 10.1.1.1
hostname(config-aaa-server-host)# key TACPlusUauthKey
hostname(config-aaa-server-host)# exit
hostname(config)# access-list TELNET_AUTH extended permit tcp any host 209.165.201.5 eq
telnet
hostname(config)# aaa authentication match TELNET_AUTH outside AuthInbound
```

次に、ユーザが ASA を介してログインすることを可能にする典型的なカットスルー プロキシ設定の例を示します。この例は次の条件に基づいています。

- ASA の IP アドレスは 192.168.123.10 です。
- Active Directory ドメイン コントローラの IP アドレスは 10.1.2.10 です。
- エンド ユーザ クライアントは、IP アドレスが 192.168.123.10 であり、HTTPS を使用して Web ポータル経由でログインします。
- ユーザは、LDAP を介して Active Directory ドメイン コントローラにより認証されます。
- ASA は、Inside インターフェイスを使用して企業ネットワーク上の Active Directory ドメイン コントローラに接続します。

```
hostname(config)# access-list AUTH extended permit tcp any 192.168.123.10 255.255.255.0 eq http
hostname(config)# access-list AUTH extended permit tcp any 192.168.123.10 255.255.255.0 eq https
hostname(config)# aaa-server LDAP protocol ldap
hostname(config-aaa-server-group)# aaa-server LDAP (inside) host 10.1.2.10
hostname(config-aaa-server-host)#  ldap-base-dn DC=cisco,DC=com
hostname(config-aaa-server-host)#  ldap-group-base-dn DC=cisco,DC=com
hostname(config-aaa-server-host)#  ldap-scope subtree
hostname(config-aaa-server-host)#  ldap-login-dn cn=kao,OU=Employees,OU=Cisco  Users,DC=cisco,DC=com
hostname(config-aaa-server-host)#  ldap-login-password *****
hostname(config-aaa-server-host)#  ldap-over-ssl enable
hostname(config-aaa-server-host)#  server-type microsoft
hostname(config-aaa-server-host)# aaa authentication match AUTH inside LDAP
hostname(config)#
hostname(config)# http server enable
hostname(config)# http 0.0.0.0 0.0.0.0 inside
hostname(config)#
hostname(config)# auth-prompt prompt Enter Your Authentication
hostname(config)# auth-prompt accept You are Good
hostname(config)# auth-prompt reject Goodbye
```

この例には、次のガイドラインが適用されます。

- **access-list** コマンドでは、未認証の着信ユーザが AAA カットスルー プロキシをトリガーできるように、**access-list 100 ex deny any any** コマンドを入力する前に permit user NONE ルールを設定する必要があります。
- **access-list AUTH** コマンドでは、permit user NONE ルールにより、未認証のユーザだけが AAA カットスルー プロキシをトリガーできるよう指定されます。

```
hostname(config)# access-list listenerAuth extended permit tcp any any
hostname(config)# aaa authentication match listenerAuth inside ldap
hostname(config)# aaa authentication listener http inside port 8888
```

```
hostname(config)# access-list 100 ex permit ip user SAMPLE\user1 any any
hostname(config)# access-list 100 ex deny ip user SAMPLE\user2 any any
hostname(config)# access-list 100 ex permit ip user NONE any any
hostname(config)# access-list 100 ex deny any any
hostname(config)# access-group 100 in interface inside
hostname(config)# aaa authenticate match 100 inside user-identity
```

次に、AAA ルールとアイデンティティ ファイアウォール（カットスルー プロキシ）を使用して、正常に認証する例を示します。

```
hostname(config)# access-list 100 ex permit ip user CISCO\xyz any any
hostname(config)# access-list 100 ex deny ip user CISCO\abc any any
hostname(config)# access-list 100 ex permit ip user NONE any any
hostname(config)# access-list 100 ex deny any any
hostname(config)# access-group 100 in interface inside
hostname(config)# access-list 200 ex permit user NONE any any
hostname(config)# aaa authenticate match 200 inside user-identity
```

認証の詳細については、「認証について」（P.7-2）を参照してください。

Web クライアントのセキュアな認証のイネーブル化

HTTP 認証を使用する場合、デフォルトでユーザ名とパスワードがクリア テキストでクライアントから ASA に送信されます。さらにこのユーザ名とパスワードは宛先 Web サーバにも送信されます。

ASA では、HTTP 認証の保護のために次のような方式を用意しています。

- HTTP について認証のリダイレクション方式をイネーブルにする：**aaa authentication listener** コマンドに **redirect** キーワードを指定して使用します。この方式では、認証クレデンシャルがその後続けて宛先サーバに送信されないようにします。リダイレクション方式と基本方式を比較した詳細については、「ASA の認証プロンプト」（P.7-3）を参照してください。
- 仮想 HTTP をイネーブルにする：**virtual http** コマンドを使用して、ASA による認証と HTTP サーバによる認証を別々に受けることができます。HTTP サーバが 2 次認証を必要としない場合でも、このコマンドにより基本認証クレデンシャルは HTTP GET 要求から除去されます。詳細については、「仮想サーバによる HTTP 接続および HTTPS 接続の認証」（P.7-11）を参照してください。

  Web クライアントと ASA の間の HTTPS によるユーザ名とパスワードの交換をイネーブルにする：**aaa authentication secure-http-client** コマンドを使用して、Web クライアントと ASA との間の HTTPS によるユーザ名とパスワードの交換をイネーブルにします。これは ASA と宛先サーバの間だけでなく、クライアントと ASA の間のクレデンシャルを保護する唯一の方式です。この方式だけを使用することも、または他の方式のいずれかと組み合わせてセキュリティを最大限にすることもできます。

  この機能をイネーブルにすると、ユーザが HTTP の使用時に認証を必要とする場合は、ASA が HTTP ユーザを HTTPS プロンプトにリダイレクトします。正常に認証されると、ユーザは ASA により元の HTTP URL にリダイレクトされます。

  セキュアな Web クライアント認証では、次の制限事項があります。

  - 同時に行うことができる HTTPS 認証セッションは、最大 16 個です。16 個の HTTPS 認証プロセスがすべて実行されている場合、認証を必要とする新しい接続は失敗します。
  - **uauth timeout 0** が設定されると（**uauth timeout** が 0 に設定される）、HTTPS 認証は機能しない場合があります。HTTPS 認証を受けた後、ブラウザが複数の TCP 接続を開始して Web ページのロードを試みると、最初の接続はそのまま許可されますが、後続の接続に対しては認証が発生します。その結果、ユーザが認証ページに正しいユーザ名とパスワードを毎回入力しても、繰り返し認証ページが表示されます。この状況を回避するには、**timeout uauth 0:0:1**

コマンドで **uauth timeout** を 1 秒に設定します。ただし、この回避策では、同じ送信元 IP アドレスからアクセスした認証されていないユーザがファイアウォールを通過できる期間が 1 秒間発生します。

HTTPS 認証は SSL ポート 443 で行われるため、HTTP クライアントから HTTP サーバ ポート 443 へのトラフィックをブロックするように、**access-list** コマンド ステートメントを設定しないでください。また、ポート 80 上の Web トラフィックに対してスタティック PAT を設定する場合は、SSL ポートに対してもスタティック PAT を設定する必要があります。

- 次の例では、最初のコマンド群で Web トラフィックに対してスタティック PAT が設定されるため、HTTPS 認証コンフィギュレーションをサポートするために 2 番目のコマンド群を追加する必要があります。

```
object network obj-10.130.16.10-01
  host 10.130.16.10
  nat (inside,outside) static 10.132.16.200 service tcp 80 80
object network obj-10.130.16.10-02
  host 10.130.16.10
  nat (inside,outside) static 10.132.16.200 service tcp 443 443
```

ASA での直接認証

HTTP、HTTPS、Telnet、または FTP が ASA を通過することを許可せずに、他のタイプのトラフィックを認証する場合は、HTTP、HTTPS、または Telnet を使用して ASA の認証を直接受けることができます。

この項は、次の内容で構成されています。

仮想サーバによる HTTP 接続および HTTPS 接続の認証

「ネットワーク アクセス認証の設定」（P.7-6）に示す HTTP および HTTPS 認証のリダイレクト方式をイネーブルにした場合は、直接認証も自動的にイネーブルになります。

ASA で HTTP 認証を使用する場合（「ネットワーク アクセス認証の設定」（P.7-6）を参照）、ASA では、基本 HTTP 認証がデフォルトで使用されます。

基本 HTTP 認証を引き続き使用しながら、HTTP および HTTPS に対する直接認証をイネーブルにするには、次のコマンドを入力します。

コマンド	目的
aaa authentication listener http[**s**] *interface_name* [**port** *portnum*] **redirect** **例：** `hostname(config)# aaa authentication listener http inside redirect`	（任意）HTTP または HTTPS 接続の認証のリダイレクション方式をイネーブルにします。 引数 *interface_name* は、リスニング ポートをイネーブルにするインターフェイスです。**port** *portnum* 引数で、ASA がリッスンするポート番号を指定します。デフォルトは 80（HTTP）と 443（HTTPS）です。 任意のポート番号を使用して同じ機能を保持できますが、直接認証ユーザがそのポート番号を認識している必要があります。これは、リダイレクトされたトラフィックは正しいポート番号に自動的に送信されますが、直接認証するユーザは、ポート番号を手動で指定する必要があるためです。 このコマンドを HTTP と HTTPS について別々に入力します。

ASA だけでなく宛先 HTTP サーバでも認証が必要な場合は、次のコマンドを入力して、ASA（AAA サーバ経由）での認証と HTTP サーバでの認証を個別に行います。

コマンド	目的
virtual http **例：** hostname(config)# virtual http	AAA 認証を必要とするすべての HTTP 接続を ASA 上の仮想 HTTP サーバにリダイレクトします。ASA により、AAA サーバのユーザ名とパスワードの入力を求めるプロンプトが表示されます。AAA サーバがユーザを認証すると、ASA は HTTP 接続を元のサーバにリダイレクトして戻しますが、AAA サーバのユーザ名とパスワードは含めません。HTTP パケットにユーザ名とパスワードが含まれていないため、HTTP サーバによりユーザに HTTP サーバのユーザ名とパスワードの入力を求めるプロンプトが別途表示されます。 着信ユーザ（セキュリティの低い方から高い方へ向かう）については、送信元インターフェイスに適用される ACL に、宛先インターフェイスとして仮想 HTTP アドレスを追加する必要もあります。さらに、NAT が必要ない場合であっても、仮想 HTTP IP アドレスに対するスタティック NAT コマンドを追加する必要があります。通常、アイデンティティ NAT コマンドが使用されます（アドレスを同一アドレスに変換）。 発信ユーザについては、トラフィックに明示的な許可がありますが、内部インターフェイスに ACL を適用する場合は、必ず仮想 HTTP アドレスへのアクセスを許可する必要があります。static ステートメントは不要です。 **（注）** **virtual http** コマンドを使用する場合は、**timeout uauth** コマンドの期間を 0 秒に設定しないでください。設定すると、実際の Web サーバへの HTTP 接続ができなくなります。 インターフェイスの AAA をイネーブルにすると、次の URL で ASA の直接認証を受けることができます。 **http://***interface_ip*[**:***port*]**/netaccess/connstatus.html** **https://***interface_ip*[**:***port*]**/netaccess/connstatus.html** 仮想 HTTP を使用しない場合は、ASA による認証で使用したものと同じユーザ名とパスワードが HTTP サーバに送信されます。HTTP サーバのユーザ名とパスワードを別に入力するように求められることはありません。AAA サーバと HTTP サーバでユーザ名とパスワードが異なる場合、HTTP 認証は失敗します。

仮想サーバによる Telnet 接続の認証

任意のプロトコルやサービスにネットワーク アクセス認証を設定できますが（**aaa authentication match** または **aaa authentication include** コマンドを参照）、直接に認証できるのは、HTTP、Telnet、または FTP の場合だけです。ユーザがまずこれらのサービスのいずれかで認証を受けておかないと、他のサービスは通過を許可されません。HTTP、Telnet、または FTP トラフィックの ASA の通過を許可せず、その他のタイプのトラフィックを認証する場合は、ASA 上で設定された所定の IP アドレスにユーザが Telnet で接続し、ASA によって Telnet プロンプトが表示されるように、仮想 Telnet を設定できます。

仮想 Telnet サーバを設定するには、次のコマンドを入力します。

コマンド	目的
virtual telnet *ip_address* **例：** hostname(config)# virtual telnet 209.165.202.129	仮想 Telnet サーバを設定します。 *ip_address* 引数によって、仮想 Telnet サーバの IP アドレスを設定します。このアドレスは必ず、ASA にルーティングされる未使用のアドレスにしてください。 **authentication match** コマンドか **aaa authentication include** コマンドを使用して、仮想 Telnet アドレスへの Telnet アクセスの認証、さらには認証する他のサービスを設定する必要があります。 認証が済んでいないユーザが仮想 Telnet IP アドレスに接続すると、ユーザはユーザ名とパスワードを求められ、その後 AAA サーバにより認証されます。認証されると、ユーザには「Authentication Successful.」というメッセージが表示されます。それ以降、ユーザは認証を必要とする他のサービスに正常にアクセスできます。 着信ユーザ（セキュリティの低い方から高い方へ向かう）については、送信元インターフェイスに適用される ACL に、宛先インターフェイスとして仮想 Telnet アドレスを追加する必要もあります。さらに、NAT が必要ない場合であっても、仮想 Telnet IP アドレスに対するスタティック NAT コマンドを追加する必要があります。通常、アイデンティティ NAT コマンドが使用されます（アドレスを同一アドレスに変換）。 発信ユーザについては、トラフィックに明示的な許可がありますが、内部インターフェイスに ACL を適用する場合は、必ず仮想 Telnet アドレスへのアクセスを許可する必要があります。static ステートメントは不要です。 ASA からログアウトするには、仮想 Telnet IP アドレスに再接続します。その後、ログアウトするように求められます。

例

次に、その他のサービスに対して AAA 認証とともに仮想 Telnet をイネーブルにする例を示します。

```
hostname(config)# virtual telnet 209.165.202.129
hostname(config)# access-list ACL-IN extended permit tcp any host 209.165.200.225 eq smtp
hostname(config)# access-list ACL-IN remark This is the SMTP server on the inside
hostname(config)# access-list ACL-IN extended permit tcp any host 209.165.202.129 eq
telnet
hostname(config)# access-list ACL-IN remark This is the virtual Telnet address
hostname(config)# access-group ACL-IN in interface outside
hostname(config)# network object obj-209.165.202.129-01
hostname(config-network-object)# host 209.165.202.129
hostname(config-network-object)# nat (inside,outside) static 209.165.202.129
hostname(config)# access-list AUTH extended permit tcp any host 209.165.200.225 eq smtp
hostname(config)# access-list AUTH remark This is the SMTP server on the inside
hostname(config)# access-list AUTH extended permit tcp any host 209.165.202.129 eq telnet
hostname(config)# access-list AUTH remark This is the virtual Telnet address
hostname(config)# aaa authentication match AUTH outside tacacs+
```

ネットワーク アクセス許可の設定

ユーザが所定の接続のための認証を受けると、ASA は許可を使用して、ユーザからのトラフィックをさらに制御できます。

この項は、次の内容で構成されています。

TACACS+ 許可の設定

TACACS+ でネットワーク アクセス許可を実行するように、ASA を設定できます。許可ルールが一致する必要のある ACL を指定することにより、許可するトラフィックを指定します。または、許可ルール自体で直接、トラフィックを指定することもできます。

ヒント　ACL を使用して許可するトラフィックを指定すると、入力する必要のある許可コマンドの数を大幅に少なくすることができます。これは、入力した各許可規則では、送信元と宛先のサブネットとサービスを 1 つだけ指定できるのに対して、ACL には多数のエントリを含めることができるためです。

認証ステートメントと許可ステートメントは互いに独立しています。ただし、認証されていないトラフィックは、許可ルールに一致した場合でも拒否されます。許可は次のように実行されます。

1. ユーザは最初に ASA で認証を受ける必要があります。

   所定の IP アドレスのユーザは、すべてのルールおよびタイプに対して一度だけ認証を受ければよいため、認証セッションが期限切れになっていなければ、トラフィックが認証ルールで一致した場合でも、許可が発生することがあります。

2. ユーザの認証が完了すると、ASA は、一致するトラフィックの許可ルールをチェックします。
3. トラフィックが許可ルールに一致した場合は、ASA によりユーザ名が TACACS+ サーバに送信されます。
4. TACACS+ サーバは ASA に応答し、ユーザ プロファイルに基づいてそのトラフィックの許可または拒否を示します。
5. ASA は、その応答内の許可ルールを実施します。

ユーザに対するネットワーク アクセス許可の設定については、ご使用の TACACS+ サーバのマニュアルを参照してください。

TACACS+ 許可を設定するには、次の手順を実行します。

	コマンド	目的
ステップ 1	**aaa-server** **例：** `hostname(config)# aaa-server AuthOutbound protocol tacacs+`	AAA サーバを指定します。すでに指定済みの場合は、次の手順に進みます。AAA サーバの指定方法の詳細については、一般的な操作のコンフィギュレーション ガイドの“Configuring AAA Server Groups” section on page 33-11 を参照してください。
ステップ 2	**access-list** **例：** `hostname(config)# access-list MAIL_AUTH extended permit tcp any any eq smtp`	認証するトラフィックの送信元アドレスと宛先アドレスを指定する ACL を作成します。詳細については、一般的な操作のコンフィギュレーション ガイドのChapter 19, “Adding an Extended Access Control List,” を参照してください。 **許可** ACE は、一致したトラフィックを認証するようにマークします。一方、**拒否**エントリは、一致したトラフィックを認証から除外します。HTTP、HTTPS、Telnet、または FTP のいずれかの宛先ポートを ACL に必ず含めます。これは、ユーザがこれらのサービスのいずれかの認証を受けないと、他のサービスが ASA の通過を許可されないためです。
ステップ 3	**aaa authentication match** *acl_name interface_name server_group* **例：** `hostname(config)# aaa authentication match MAIL_AUTH inside AuthOutbound`	認証を設定します。*acl_name* 引数は、ステップ 2 で作成した ACL の名前です。*interface_name* 引数は、**nameif** コマンドで指定されるインターフェイスの名前、また *server_group 引数*は、ステップ 1 で作成した AAA サーバ グループです。 **（注）** もう 1 つの方法として、**aaa authentication include** コマンド（コマンド内でトラフィックを指定するコマンド）を使用することもできます。ただし、同一コンフィギュレーション内で両方の方法を使用することはできません。詳細については、コマンド リファレンスを参照してください。
ステップ 4	**aaa authentication listener http**[**s**] *interface_name* [**port** *portnum*] **redirect** **例：** `hostname(config)# aaa authentication listener http inside redirect`	（任意）HTTP または HTTPS 接続の認証のリダイレクション方式をイネーブルにします。 引数 *interface_name* は、リスニング ポートをイネーブルにするインターフェイスです。**port** *portnum* 引数で、ASA がリッスンするポート番号を指定します。デフォルトは 80（HTTP）と 443（HTTPS）です。 任意のポート番号を使用して同じ機能を保持できますが、直接認証ユーザがそのポート番号を認識している必要があります。これは、リダイレクトされたトラフィックは正しいポート番号に自動的に送信されますが、直接認証するユーザは、ポート番号を手動で指定する必要があるためです。 このコマンドを HTTP と HTTPS について別々に入力します。

	コマンド	目的
ステップ 5	**aaa local authentication attempts max-fail** *number* **例：** `hostname(config)# aaa local authentication attempts max-fail 7`	（任意）ネットワーク アクセス認証にローカル データベースを使用し、ASA が所定のユーザ アカウントを許可するローカル ログイン連続失敗試行回数を制限します（特権レベル 15 のユーザは除きます。この機能はレベル 15 のユーザには影響はありません）。*number* 引数の値は 1 ～ 16 です。 **ヒント** 特定のユーザまたはすべてのユーザのロックアウト ステータスを解除するには、**clear aaa local user lockout** コマンドを使用します。
ステップ 6	**access-list** **例：** `hostname(config)# access-list TELNET_AUTH extended permit tcp any any eq telnet`	許可するトラフィックの送信元アドレスと宛先アドレスを指定する ACL を作成します。手順については、一般的な操作のコンフィギュレーション ガイドの Chapter 19, “Adding an Extended Access Control List,” を参照してください。 **許可** ACE は、一致したトラフィックを許可するようにマークします。一方、**拒否**エントリは、一致したトラフィックを許可から除外します。許可の照合に使用する ACL には、認証の照合に使用される ACL 内のルールと同じか、その一部のルールが含まれている必要があります。 **（注）** 認証を設定してあり、認証されているすべてのトラフィックを許可する場合は、**aaa authentication match** コマンドで作成した同じ ACL を使用できます。
ステップ 7	**aaa authorization match** *acl_name interface_name server_group* **例：** `hostname(config)# aaa authentication match TELNET_AUTH inside AuthOutbound`	許可をイネーブルにします。 *acl_name* 引数はステップ 6 で作成した ACL の名前、*interface_name* 引数は **nameif** コマンドで指定したインターフェイスまたはデフォルトで指定されているインターフェイスの名前、*server_group* 引数は認証をイネーブルにしたときに作成した AAA サーバ グループです。 **（注）** もう 1 つの方法として、**aaa authorization include** コマンド（コマンド内でトラフィックを指定するコマンド）を使用することもできます。ただし、同一コンフィギュレーション内で両方の方法を使用することはできません。詳細については、コマンド リファレンスを参照してください。

例

次の例では、内部 Telnet トラフィックを認証および許可します。209.165.201.5 以外のサーバに向かう Telnet トラフィックは認証だけを受けますが、209.165.201.5 に向かうトラフィックには許可が必要です。

```
hostname(config)# access-list TELNET_AUTH extended permit tcp any any eq telnet
hostname(config)# access-list SERVER_AUTH extended permit tcp any host 209.165.201.5 eq telnet
hostname(config)# aaa-server AuthOutbound protocol tacacs+
```

```
hostname(config-aaa-server-group)# exit
hostname(config)# aaa-server AuthOutbound (inside) host 10.1.1.1
hostname(config-aaa-server-host)# key TACPlusUauthKey
hostname(config-aaa-server-host)# exit
hostname(config)# aaa authentication match TELNET_AUTH inside AuthOutbound
hostname(config)# aaa authorization match SERVER_AUTH inside AuthOutbound
```

RADIUS 許可の設定

認証が成功すると、RADIUS プロトコルは RADIUS サーバによって送信される access-accept メッセージでユーザ許可を返します。認証の設定の詳細については、「ネットワーク アクセス認証の設定」（P.7-6）を参照してください。

ネットワーク アクセスについてユーザを認証するように ASA を設定すると、RADIUS 許可も暗黙的にイネーブルになっています。したがって、この項では、ASA 上の RADIUS 許可の設定については取り上げません。ここでは、ASA が RADIUS サーバから受信した ACL 情報をどのように処理するかについて説明します。

ACL を ASA にダウンロードするように RADIUS サーバを設定できます。または、認証時に ACL 名をダウンロードするようにも設定できます。ユーザは、ユーザ固有の ACL で許可された操作だけを許可されます。

（注）　**access-group** コマンドを使用して ACL をインターフェイスに適用した場合は、**per-user-override** キーワードが、ユーザ固有の ACL による許可に対して次のように影響を与えることに注意してください。

- **per-user-override** キーワードを使用しない場合、ユーザ セッションのトラフィックは、インターフェイス ACL とユーザ固有の ACL の両方によって許可される必要があります。
- **per-user-override** キーワードを使用した場合、ユーザ固有の ACL によって許可される内容が決定されます。

詳細については、コマンド リファレンスの **access-group** コマンドの項を参照してください。

この項は、次の内容で構成されています。

ダウンロード可能なアクセス コントロール リスト（ACL）を送信するための RADIUS サーバの設定

この項では、Cisco Secure Access Control Server（ACS）およびサードパーティ RADIUS サーバを設定する方法について説明します。次の項目を取り上げます。

ダウンロード可能な ACL の機能と Cisco Secure ACS について

ダウンロード可能な ACL は、Cisco Secure ACS を使用して各サーバに適切な ACL を提供する場合に最もスケーラブルな方法です。次の機能があります。

- 無制限 ACL のサイズ: ダウンロード可能な ACL は、完全な ACL を Cisco Secure ACS から ASA に転送するために必要な数の RADIUS パケットを使用して送信されます。
- ACL 管理の簡素化および集中化：ダウンロード可能な ACL により、一度記述した ACL セットを多数のユーザ プロファイルまたはグループ プロファイルに適用することや、多数の ASA に配布することができます。

この方法は、複数の Cisco Secure ACS ユーザまたはグループに適用する非常に大きい ACL セットがある場合に最適ですが、Cisco Secure ACS ユーザおよびグループの管理を簡素化できることから、ACL のサイズを問わず有用です。

ASA は、ダウンロード可能な ACL を Cisco Secure ACS から次のプロセスで受信します。

1. ASA がユーザ セッションのための RADIUS 認証要求パケットを送信します。
2. Cisco Secure ACS がそのユーザを正常に認証した場合、Cisco Secure ACS は、該当するダウンロード可能な ACL の内部名が含まれた RADIUS access-accept メッセージを返します。Cisco IOS cisco-av-pair RADIUS VSA（ベンダー 9、属性 1）には、ダウンロード可能な ACL セットを特定する次の AV のペアが含まれています。

   ```
   ACS:CiscoSecure-Defined-ACL=acl-set-name
   ```

   acl-set-name はダウンロード可能な ACL の内部名です。この名前は、Cisco Secure ACS 管理者が ACL に割り当てた名前と ACL が最後に変更された日時の組み合わせです。
3. ASA はダウンロード可能な ACL の名前を検査し、以前にその名前のダウンロード可能な ACL を受信したことがあるかどうかを判別します。
   - ASA が以前にその名前のダウンロード可能な ACL を受信したことがある場合は、Cisco Secure ACS との通信は完了し、ASA は ACL をユーザ セッションに適用します。ダウンロード可能な ACL の名前には最後に変更された日時が含まれているため、Cisco Secure ACS から送信された名前と、以前にダウンロードした ACL の名前が一致するということは、ASA はダウンロード可能な ACL の最新バージョンを持っていることになります。
   - ASA が以前にその名前のダウンロード可能な ACL を受信したことがない場合は、その ACL の古いバージョンを持っているか、その ACL のどのバージョンもダウンロードしたことがないことになります。いずれの場合でも、ASA は、ダウンロード可能な ACL 名を RADIUS 要求内のユーザ名として使用し、ヌル パスワード属性とともに RADIUS 認証要求を発行します。cisco-av-pair RADIUS VSA では、この要求に次の属性と値のペアも含まれます。

     ```
     AAA:service=ip-admission
     AAA:event=acl-download
     ```

     これに加えて、ASA は Message-Authenticator 属性（IETF RADIUS 属性 80）で要求に署名します。
4. ダウンロード可能な ACL の名前が含まれているユーザ名属性を持つ RADIUS 認証要求を受信すると、Cisco Secure ACS は Message-Authenticator 属性をチェックして要求を認証します。Message-Authenticator 属性がない場合、または正しくない場合、Cisco Secure ACS はその要求を無視します。Message-Authenticator 属性の存在により、ダウンロード可能な ACL 名がネットワーク アクセスの不正取得に悪用されることが防止されます。Message-Authenticator 属性とその使用方法は、RFC 2869「RADIUS Extensions」で定義されています。この文書は、http://www.ietf.org で入手できます。

5. 要求された ACL の長さが約 4 KB 未満の場合、Cisco Secure ACS はその ACL を含めた access-accept メッセージで応答します。メッセージの一部に他の必須属性を含める必要があるので、1 つの access-accept メッセージに収まる ACL の最大サイズは 4 KB よりわずかに小さくなります。

   Cisco Secure ACS はダウンロード可能な ACL を cisco-av-pair RADIUS VSA で送信します。ACL は、一連の属性と値のペアという形式をとります。各ペアには ACE が 1 つ含まれ、シリアル番号が付けられます。

   ```
   ip:inacl#1=ACE-1
   ip:inacl#2=ACE-2
   .
   .
   .
   ip:inacl#n=ACE-n

   ip:inacl#1=permit tcp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
   ```

6. 要求された ACL の長さが約 4 KB を超える場合、Cisco Secure ACS は、上記の形式の ACL の一部が含まれた access-challenge メッセージで応答します。メッセージには、State 属性（IETF RADIUS 属性 24）も含まれています。State 属性には、Cisco Secure ACS がダウンロードの進捗を追跡するために使用する制御データが含まれています。Cisco Secure ACS は、RADIUS メッセージの最大サイズ以内で可能な限り多数の完全な属性と値のペアを cisco-av-pair RADIUS VSA に含めます。

   ASA は ACL の一部を受信すると、それを保存し、新しい access-request メッセージで応答します。これには、ダウンロード可能な ACL を求める最初の要求と同じ属性と、access-challenge メッセージで受信した State 属性のコピーが含まれています。

   このプロセスは、Cisco Secure ACS が ACL の最後の部分を access-accept メッセージで送信するまで繰り返されます。

ダウンロード可能な ACL に関する Cisco Secure ACS の設定

Cisco Secure ACS 上のダウンロード可能な ACL を共有プロファイル コンポーネントとして設定し、その ACL をグループまたは個々のユーザに割り当てることができます。

ACL 定義は、次のプレフィックスがない点を除いて拡張 **access-list** コマンド（コマンド リファレンスを参照）に類似する、1 つまたは複数の ASA のコマンドで構成されます。

access-list *acl_name* **extended**

Cisco Secure ACS バージョン 3.3 上のダウンロード可能な ACL 定義の例を次に示します。

```
+---------------------------------------------+
| Shared profile Components                   |
|                                             |
|      Downloadable IP ACLs Content           |
|                                             |
| Name:    acs_ten_acl                        |
|                                             |
|      ACL Definitions                        |
|                                             |
| permit tcp any host 10.0.0.254              |
| permit udp any host 10.0.0.254              |
| permit icmp any host 10.0.0.254             |
| permit tcp any host 10.0.0.253              |
| permit udp any host 10.0.0.253              |
| permit icmp any host 10.0.0.253             |
| permit tcp any host 10.0.0.252              |
| permit udp any host 10.0.0.252              |
| permit icmp any host 10.0.0.252             |
```

```
| permit ip any any                          |
+--------------------------------------------+
```

ダウンロード可能な ACL を作成する方法、およびそれらをユーザと関連付ける方法の詳細については、ご使用のバージョンの Cisco Secure ACS のガイドを参照してください。

ASA 上では、ダウンロードされた ACL の名前は次のようになります。

```
#ACSACL#-ip-acl_name-number
```

acl_name 引数は Cisco Secure ACS で定義された名前（上記の例では *acs_ten_acl*）、*number* は Cisco Secure ACS が生成した固有のバージョン ID です。

ASA 上にダウンロードされた ACL は、次の行で構成されます。

```
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit tcp any host 10.0.0.254
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit udp any host 10.0.0.254
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit icmp any host 10.0.0.254
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit tcp any host 10.0.0.253
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit udp any host 10.0.0.253
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit icmp any host 10.0.0.253
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit tcp any host 10.0.0.252
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit udp any host 10.0.0.252
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit icmp any host 10.0.0.252
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit ip any any
```

ダウンロード可能な ACL に関する任意の RADIUS サーバの設定

ユーザ固有の ACL を Cisco IOS RADIUS cisco-av-pair VSA（ベンダー 9、属性 1）で ASA に送信するように、Cisco IOS RADIUS VSA をサポートする任意の RADIUS サーバを設定できます。

cisco-av-pair VSA で、**access-list extended** コマンド（コマンド リファレンスを参照）と類似する 1 つまたは複数の ACE を設定します。ただし、次のコマンド プレフィックスを置き換える必要があります。

access-list *acl_name* **extended**

次のテキストに置き換えます。

ip:inacl#*nnn***=**

nnn 引数は、0 ～ 999999999 の番号で、ASA 上に設定するコマンド文の順序を指定します。このパラメータを省略すると、順番は 0 となり、cisco-av-pair RADIUS VSA 内部の ACE の順序が使用されます。

RADIUS サーバ上の cisco-av-pair VSA に対して設定されている必要のある ACL 定義の例を次に示します。

```
ip:inacl#1=permit tcp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
ip:inacl#99=deny tcp any any
ip:inacl#2=permit udp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
ip:inacl#100=deny udp any any
ip:inacl#3=permit icmp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
```

cisco-av-pair 属性で送信される ACL をユーザごとに固有にする方法については、ご使用の RADIUS サーバのマニュアルを参照してください。

ASA 上では、ダウンロードされた ACL の名前は次のようになります。

```
AAA-user-username
```

username 引数は、認証を受けるユーザの名前です。

ASA 上にダウンロードされた ACL は、次の行で構成されます。RADIUS サーバ上で指定された番号に基づいた順序になっています。

```
access-list  AAA-user-bcham34-79AD4A08 permit tcp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
access-list  AAA-user-bcham34-79AD4A08 permit udp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
access-list  AAA-user-bcham34-79AD4A08 permit icmp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
access-list  AAA-user-bcham34-79AD4A08 deny tcp any any
access-list  AAA-user-bcham34-79AD4A08 deny udp any any
```

ダウンロードされた ACL の「access-list」という単語と名前の間には、2 個のスペースがあります。これらのスペースにより、ダウンロードされた ACL とローカルの ACL が区別されます。この例では、「79AD4A08」は ASA によって作成されたハッシュ値で、RADIUS サーバ上で ACL 定義がいつ変更されたかを判別するために役立ちます。

ダウンロード可能な ACL 内のワイルドカード ネットマスク表現の変換

RADIUS サーバを使用して、ダウンロード可能な ACL を Cisco VPN 3000 Series Concentrator および ASA に提供する場合は、ワイルドカード ネットマスク表現を標準のネットマスク表現に変換するように ASA を設定しなければならない場合があります。これは、Cisco VPN 3000 Series Concentrator はワイルドカード ネットマスク表現をサポートしますが、ASA は標準のネットマスク表現しかサポートしないためです。これらの違いは、RADIUS サーバ上のダウンロード可能な ACL を設定する方法に影響しますが、ワイルドカード ネットマスク表現を変換するように ASA を設定することで、その影響を最小限に抑えることができます。ワイルドカード ネットマスク表現の変換により、RADIUS サーバ上のダウンロード可能な ACL のコンフィギュレーションを変更することなく、Cisco VPN 3000 Series Concentrator 用に記述されたダウンロード可能な ACL を ASA で使用できます。

ACL ネットマスク変換は、**acl-netmask-convert** acl-netmask-convert コマンドを使用してサーバごとに設定できます。このコマンドは aaa サーバ コンフィギュレーション モードで使用できます。RADIUS サーバの設定の詳細については、一般的な操作のコンフィギュレーション ガイドの “Configuring AAA Server Groups” section on page 33-11 を参照してください。
acl-netmask-convert コマンドの詳細については、コマンド リファレンスを参照してください。

ユーザごとのアクセス コントロール リスト名をダウンロードするための RADIUS サーバの設定

ユーザ認証時に、ASA で作成済みの ACL の名前を RADIUS サーバからダウンロードするには、IETF RADIUS filter-id 属性（属性番号 11）を次のように設定します。

filter-id=*acl_name*

（注）　Cisco Secure ACS では、filter-id 属性の値は、HTML インターフェイスのボックスで、**filter-id=** を省略し、*acl_name* だけを入力して指定します。

filter-id 属性の値をユーザごとに固有にする方法については、ご使用の RADIUS サーバのマニュアルを参照してください。

ASA 上で ACL を作成するには、一般的な操作のコンフィギュレーション ガイドの Chapter 19, “Adding an Extended Access Control List,” を参照してください。

ネットワーク アクセスのアカウンティングの設定

ASA は、ASA を通過する任意の TCP トラフィックまたは UDP トラフィックについてのアカウンティング情報を RADIUS サーバまたは TACACS+ サーバに送信できます。そのトラフィックも認証されている場合、AAA サーバはユーザ名でアカウンティング情報を保持できます。トラフィックが認証済みでない場合、AAA サーバは IP アドレスによってアカウンティング情報を保持できます。ASA アカウンティング情報には、セッションの開始時刻と終了時刻、ユーザ名、そのセッションでを経由したバイト数、使用されたサービス、各セッションの継続時間が含まれます。

アカウンティングを設定するには、次の手順を実行します。

	コマンド	目的
ステップ 1	**access-list** **例：** `hostname(config)# access-list TELNET_AUTH extended permit tcp any any eq telnet`	ユーザごとのアカウンティング データを提供するように ASA を設定する場合は、認証をイネーブルにする必要があります。詳細については、「ネットワーク アクセス認証の設定」（P.7-6）を参照してください。IP アドレスごとのアカウンティング データを提供するように ASA を設定する場合は、認証をイネーブルにする必要はありません。 アカウンティング データの送信元アドレスと宛先アドレスを指定する ACL を作成します。手順については、一般的な操作のコンフィギュレーション ガイドの Chapter 19, “Adding an Extended Access Control List,” を参照してください。 permit ACE は一致したトラフィックをアカウンティングのマークを付け、deny エントリは一致したトラフィックをアカウンティングから除外します。 **（注）** 認証を設定してあり、すべてのトラフィックのアカウンティング データを認証する場合は、**aaa authentication match** コマンドで作成した同じ ACL を使用できます。
ステップ 2	**aaa accounting match** *acl_name interface_name server_group* **例：** `hostname(config)# aaa accounting match SERVER_AUTH inside AuthOutbound`	アカウンティングをイネーブルにします。 *acl_name* 引数は、**access-list** コマンドで設定された ACL の名前です。 *interface_name* 引数は、**nameif** コマンドで設定されたインターフェイス名です。 *server_group* 引数は、**aaa-server** コマンドで設定されたサーバ グループ名です。 **（注）** もう 1 つの方法として、**aaa accounting include** コマンド（コマンド内でトラフィックを指定するコマンド）を使用することもできます。ただし、同一コンフィギュレーション内で両方の方法を使用することはできません。詳細については、コマンド リファレンスを参照してください。

例

次は、内部 Telnet トラフィックの認証、許可、アカウンティングの例です。209.165.201.5 以外のサーバに向かう Telnet トラフィックは認証だけを受けますが、209.165.201.5 に向かうトラフィックには許可およびアカウンティングが必要です。

```
hostname(config)# aaa-server AuthOutbound protocol tacacs+
hostname(config-aaa-server-group)# exit
hostname(config)# aaa-server AuthOutbound (inside) host 10.1.1.1
hostname(config-aaa-server-host)# key TACPlusUauthKey
hostname(config-aaa-server-host)# exit
hostname(config)# access-list TELNET_AUTH extended permit tcp any any eq telnet
hostname(config)# access-list SERVER_AUTH extended permit tcp any host 209.165.201.5 eq
telnet
hostname(config)# aaa authentication match TELNET_AUTH inside AuthOutbound
hostname(config)# aaa authorization match SERVER_AUTH inside AuthOutbound
hostname(config)# aaa accounting match SERVER_AUTH inside AuthOutbound
```

AAA には、ユーザ アクセスに対して、ACL だけを使用する場合よりもレベルの高い保護および制御機能が用意されています。たとえば、DMZ ネットワーク上のサーバの Telnet に対するすべての Outside ユーザのアクセスを許可する ACL を作成することができます。一部のユーザだけがサーバにアクセスできるようにする際に、そのユーザの IP アドレスを常に認識しているとは限らない場合、AAA を使用すると、認証済みまたは許可済みのユーザだけに ASA を介した接続を許可することができます （Telnet サーバもまた、認証を実行します。ASA は、許可されないユーザがサーバにアクセスできないようにします)。

MAC アドレスによるトラフィックの認証と許可の免除

ASA は、特定の MAC アドレスからのトラフィックの認証および許可を免除できます。たとえば、ASA が特定のネットワークから発信される TCP トラフィックを認証し、特定のサーバからの未認証の TCP 接続は許可する場合、MAC 免除規則を使用すると、この規則で指定したサーバからのすべてのトラフィックに対して認証および許可が免除されます。

この機能は、認証プロンプトに応答できない IP 電話などのデバイスを免除する場合に特に便利です。

MAC アドレスを使用してトラフィックの認証および許可を免除するには、次の手順を実行します。

	コマンド	目的
ステップ 1	**mac-list** *id* {**deny** | **permit**} *mac macmask* **例：** hostname(config)# mac-list abc permit 00a0.c95d.0282 ffff.ffff.ffff	MAC リストを設定します。 *id* 引数は、MAC リストに割り当てる 16 進数です。一連の MAC アドレスをグループ化するには、同じ ID 値で必要な回数の **mac-list** コマンドを入力します。AAA 免除に使用できる MAC リストは 1 つだけなので、MAC リストには免除するすべての MAC アドレスを含めてください。複数の MAC リストを作成できますが、一度に使用できるのは 1 つだけです。 パケットが最適に一致するエントリではなく最初に一致するエントリを使用するため、エントリの順序が重要になります。**permit** エントリがあり、その **permit** エントリで許可されているアドレスを拒否する場合は、**permit** エントリよりも前に **deny** エントリを入力してください。 *mac* 引数には、12 桁の 16 進数の形式（nnnn.nnnn.nnnn）で送信元の MAC アドレスを指定します。 *macmask* 引数には、照合に使用される MAC アドレスの一部を指定します。たとえば、ffff.ffff.ffff は完全に MAC アドレスと一致します。ffff.ffff.0000 は最初の 8 桁だけ一致します。
ステップ 2	**aaa mac-exempt match** *id* **例：** hostname(config)# aaa mac-exempt match 1	特定の MAC リストで指定されている MAC アドレスのトラフィックを免除します。 *id* 引数は、認証および許可を免除するトラフィックの MAC アドレスが記述されている MAC リストを指定する文字列です。 **aaa mac-exempt match** コマンドのインスタンスを 1 つだけ入力できます。

例

次の例では、1 個の MAC アドレスに対する認証をバイパスします。

```
hostname(config)# mac-list abc permit 00a0.c95d.0282 ffff.ffff.ffff
hostname(config)# aaa mac-exempt match abc
```

次の例では、ハードウェア ID が 0003.E3 であるすべての Cisco IP Phone について、認証をバイパスします。

```
hostname(config)# mac-list acd permit 0003.E300.0000 FFFF.FF00.0000
hostname(config)# aaa mac-exempt match acd
```

次の例では、00a0.c95d.02b2 以外の MAC アドレス グループの認証をバイパスします。00a0.c95d.02b2 は **permit** ステートメントとも一致するため、**permit** ステートメントよりも前に **deny** ステートメントを入力します。permit ステートメントが前にある場合、**deny** ステートメントとは一致しません。

```
hostname(config)# mac-list 1 deny 00a0.c95d.0282 ffff.ffff.ffff
```

```
hostname(config)# mac-list 1 permit 00a0.c95d.0000 ffff.ffff.0000
hostname(config)# aaa mac-exempt match 1
```

AAA ルールの機能履歴

表 7-1 に、各機能変更と、それが実装されたプラットフォーム リリースを示します。

表 7-1　AAA ルールの機能履歴

機能名	プラットフォーム リリース	機能情報
AAA ルール	7.0(1)	AAA ルールでは、ネットワーク アクセスで AAA をイネーブルにする方法について説明します。 次のコマンドを導入しました。 **aaa authentication match**、**aaa authentication include | exclude**、**aaa authentication listener http[s]**、**aaa local authentication attempts max-fail**、**virtual http**、**virtual telnet**、**aaa authentication secure-http-client**、**aaa authorization match**、**aaa accounting match**、**aaa mac-exempt match**。
カットスルー プロキシを使用した認証	9.0(1)	アイデンティティ ファイアウォール機能とともに AAA ルールを使用して認証できます。 次のコマンドが変更されました。 **aaa authentication match**