ASA 8.0： WebVPN ユーザのための RADIUS 認証の設定

目次

概要

このドキュメントでは、WebVPN ユーザの認証に Remote Authentication Dial-In User Service（RADIUS）を使用するための Cisco Adaptive Security Appliance（ASA）の設定方法を説明しています。 この例での RADIUS サーバは Cisco Access Control Server（ACS）バージョン 4.1 です。この設定は、ソフトウェア バージョン 8.0(2) が稼働する ASA 上の Adaptive Security Device Manager（ASDM）6.0(2) で実行されます。

注: この RADIUS 認証例は WebVPN ユーザ用に設定されていますが、この設定を他のタイプのリモート アクセス VPN にも使用できます。 示されているように、必要な接続プロファイル（トンネル グループ）に AAA を割り当てるだけです。

前提条件

- 基本的な WebVPN 設定が必要です。
- ユーザ認証のためには、Cisco ACS にユーザが設定されている必要があります。 詳細は、『ユーザ管理』の「基本ユーザ アカウントの追加」セクションを参照してください。

ACS サーバの設定

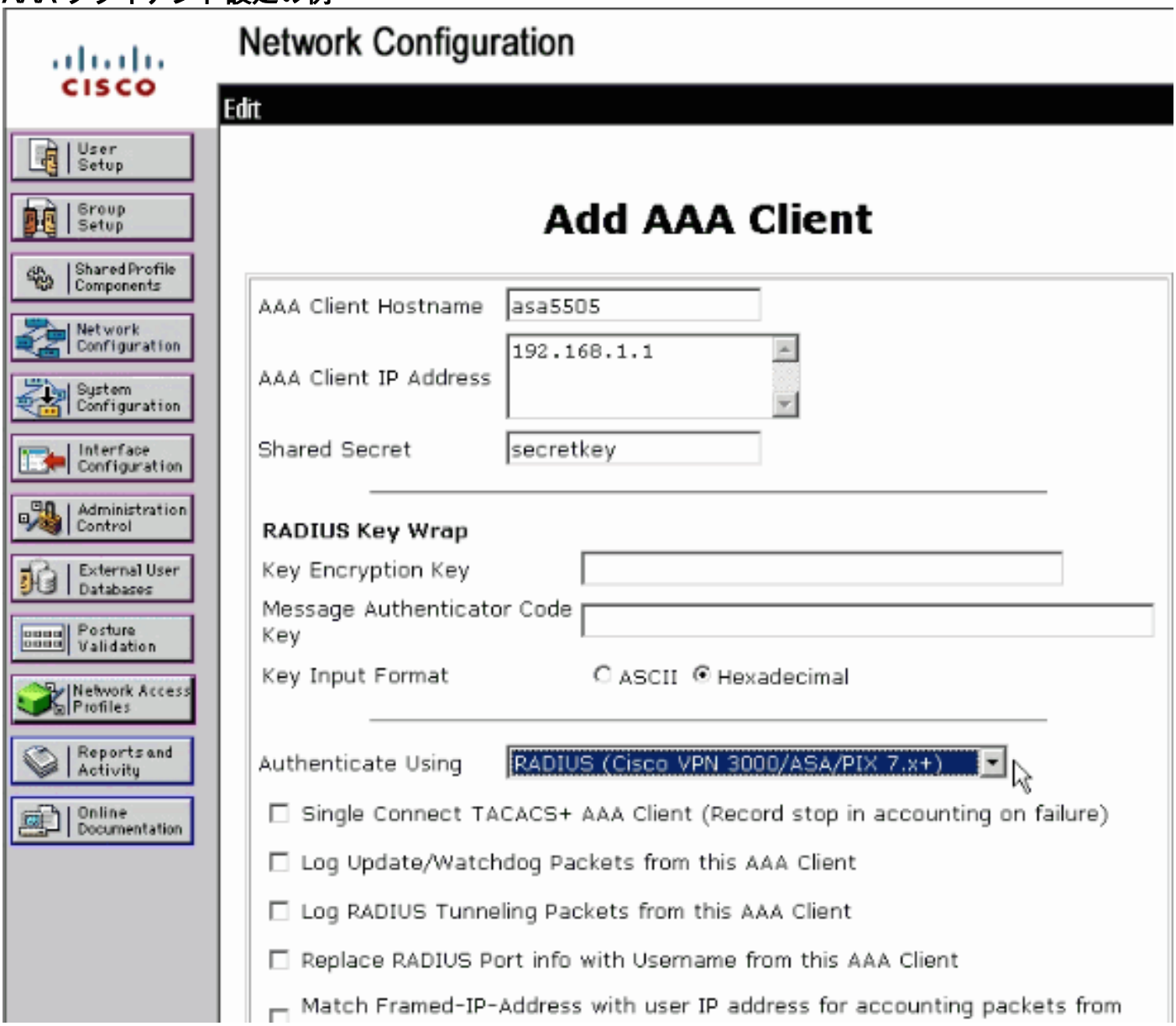
このセクションでは、ACS と ASA で RADIUS 認証を設定するための情報を提供しています。

ASA と通信するように ACS サーバを設定するには、次の手順を実行します。

1. ACS 画面の左のメニューから **Network Configuration** を選択します。

2. **AAA Clients** にある **Add Entry** をクリックします。
3. 次のようにクライアント情報を入力します。**hostname — AAA クライアント**選択の名前**AAA クライアントIPアドレス**— ACS がセキュリティ アプライアンス モデルによってが接触するアドレス**共有秘密**— ACS とセキュリティ アプライアンス モデルで設定される秘密鍵
4. **Authenticate Using** ドロップダウン メニューで **RADIUS (Cisco VPN 3000/ASA/PIX 7.x+)** を選択します。
5. [Submit+Apply] をクリックします。

AAA クライアント設定の例



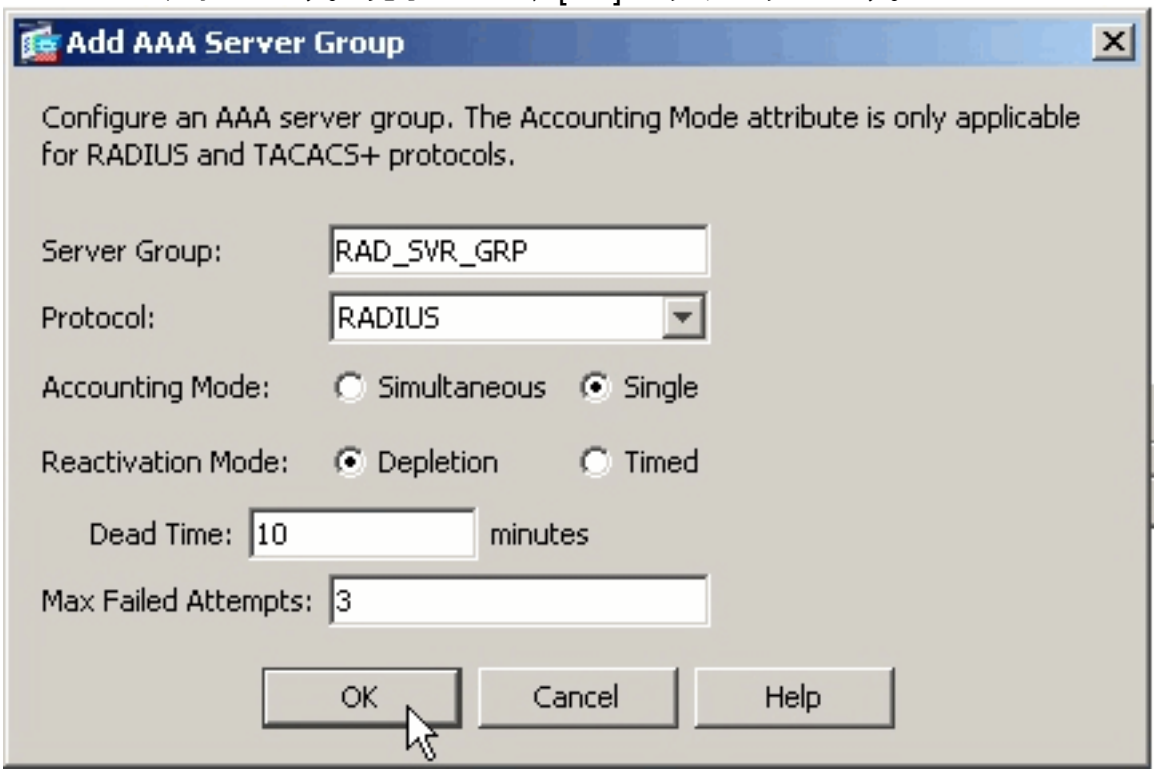
セキュリティ アプライアンスの設定

ASDM

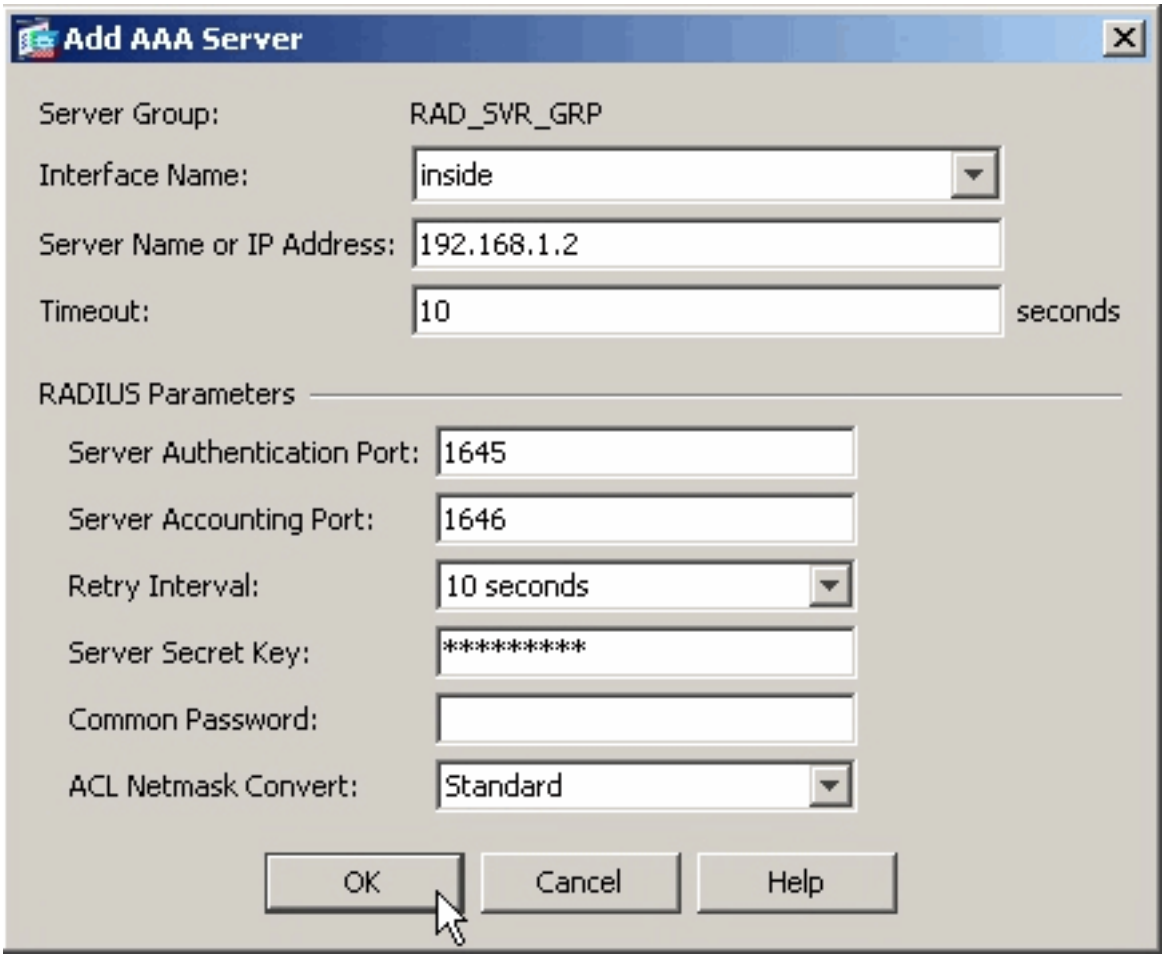
ACS サーバと通信して WebVPN クライアントを認証するように ASA を設定するには、ASDM で次の手順を実行します。

1. **> 設定されるリモートアクセス VPN > AAA > AAA サーバ グループ** 『Configuration』 を選択 して下さい。

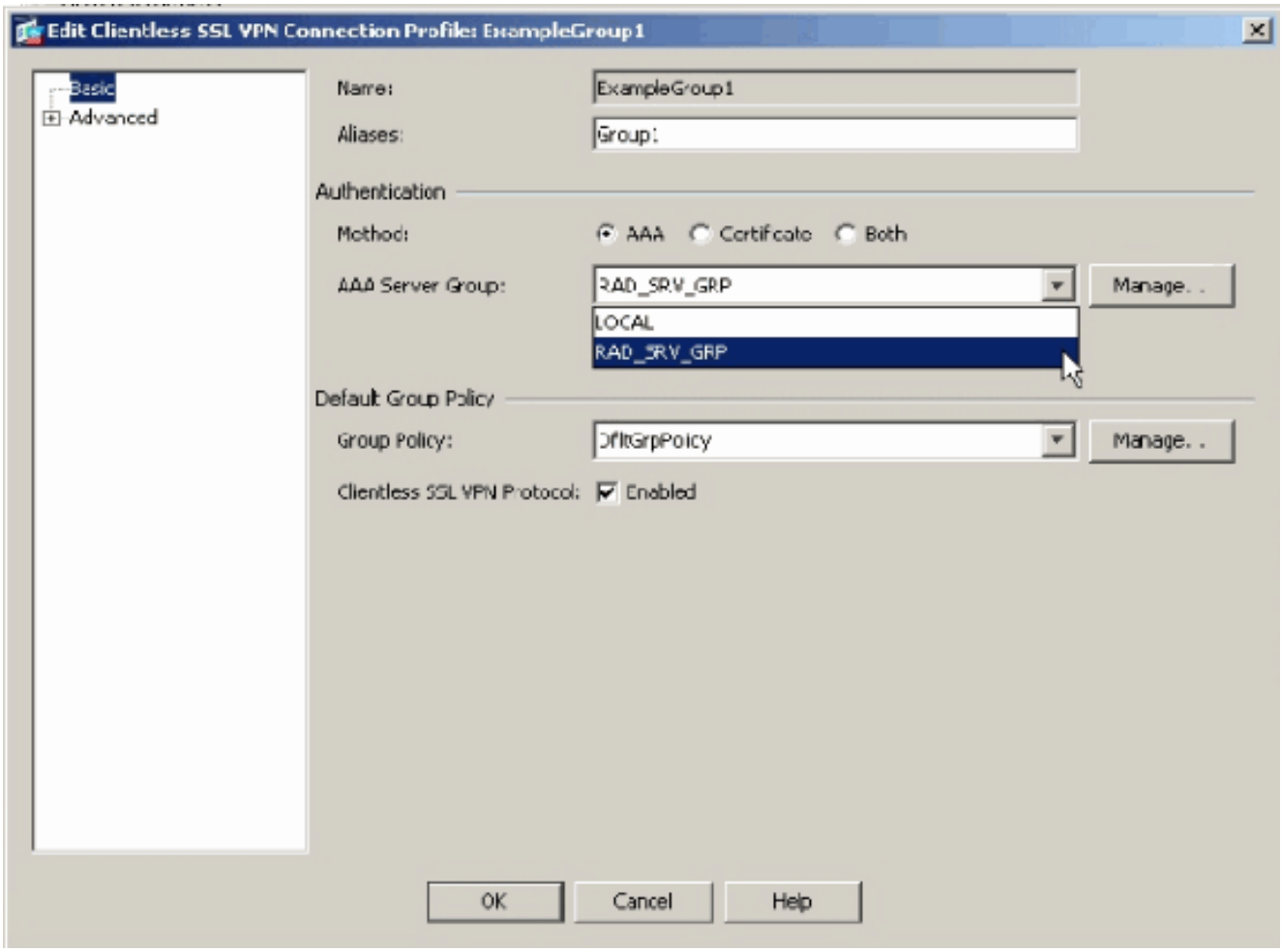
2. AAA Server Groups の横にある **Add** をクリックします。
3. 表示されたウィンドウで、新しい AAA サーバ グループの名前を指定して、プロトコルに **RADIUS** を選択します。 完了したら、[OK] をクリックします。



4. トップ ペインで自分の新しいグループが選択されているのを確認して、下方のペインの右で **Add** をクリックします。
5. 次のようにサーバ情報を入力します。**インターフェイス名**— ACS サーバに達するのに使用する ASA が必要があるインターフェイスACS サーバに達するのに使用する ASA が必要がある **ip address** — アドレス**かサーバ名サーバシークレット キー**— ACS サーバの ASA のために設定される共有秘密 キー**ASA での AAA サーバ設定例**



6. AAAサーバグループおよびサーバを設定したら、設定 > リモートアクセス VPN > Clientless SSL VPN アクセス > 接続プロファイルへのナビゲート WebVPN を新しい AAA設定を使用するために設定するため。**注:** WebVPN AAA
7. AAA を設定するプロファイルを選択して、**Edit** をクリックします。
8. **Authentication** の下で、事前に作成してある RADIUS サーバ グループを選択します。 完了したら、[OK] をクリックします。



コマンド行インターフェイス

ACS サーバと通信して WebVPN クライアントを認証するように ASA を設定するには、CLI で次の手順を実行します。

```
ciscoasa#configure terminal !--- Configure the AAA Server group. ciscoasa(config)# aaa-server
RAD_SRV_GRP protocol RADIUS ciscoasa(config-aaa-server-group)# exit !--- Configure the AAA
Server. ciscoasa(config)# aaa-server RAD_SRV_GRP (inside) host 192.168.1.2 ciscoasa(config-aaa-
server-host)# key secretkey ciscoasa(config-aaa-server-host)# exit !--- Configure the tunnel
group to use the new AAA setup. ciscoasa(config)# tunnel-group ExampleGroup1 general-attributes
ciscoasa(config-tunnel-general)# authentication-server-group RAD_SRV_GRP
```

確認

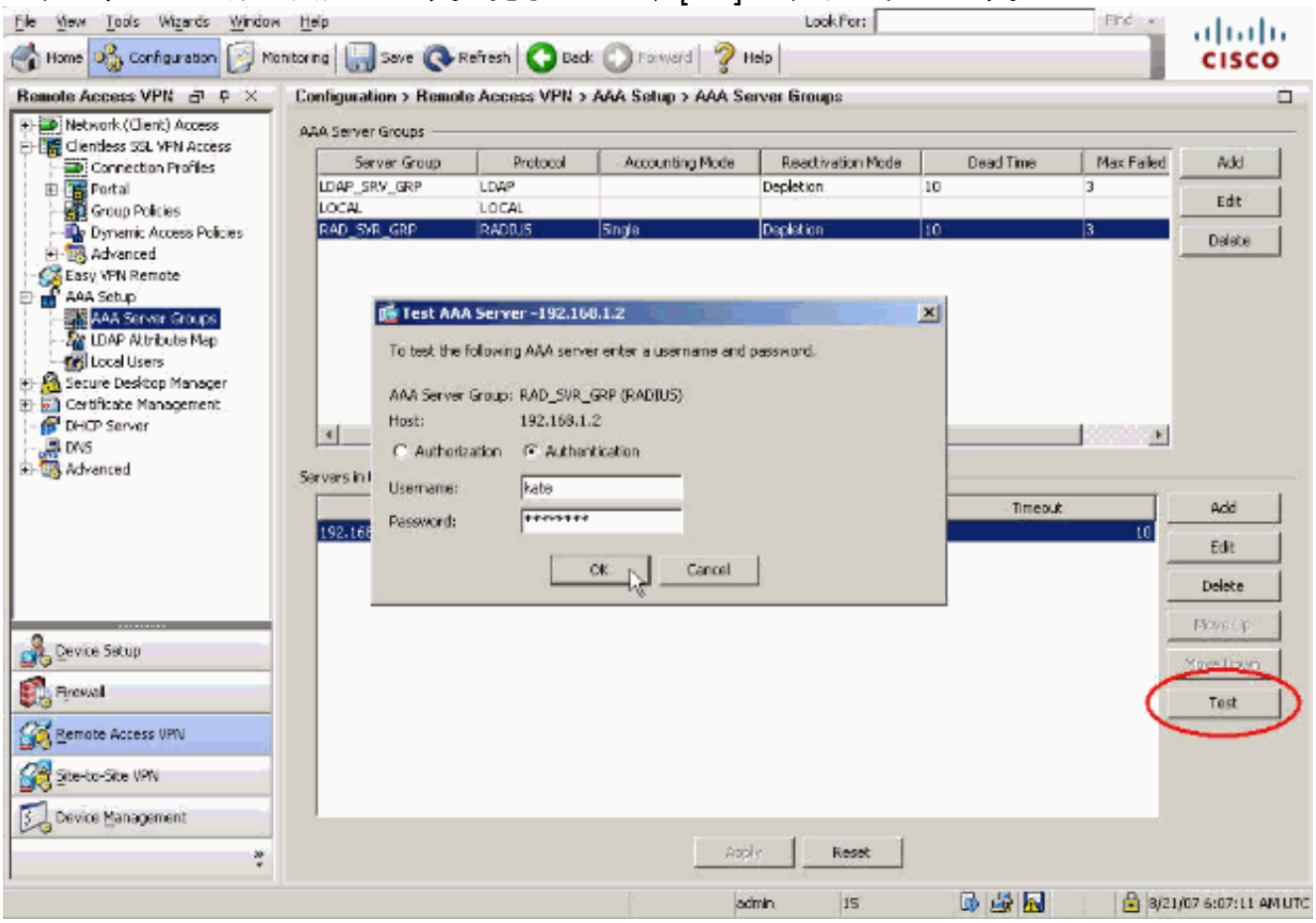
このセクションでは、設定が正常に機能していることを確認します。

ASDM でのテスト

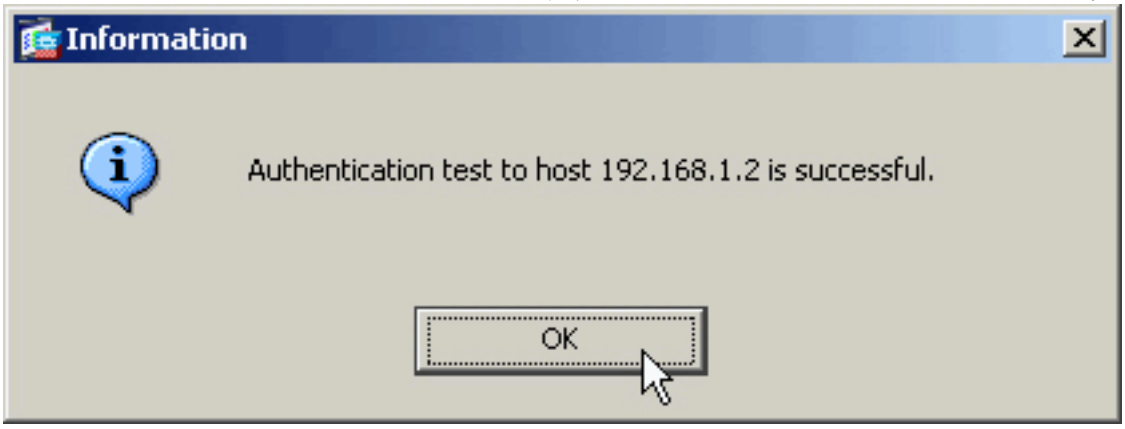
AAA Server Groups 設定画面の **Test** ボタンで、RADIUS 設定を確認します。 ユーザ名とパスワードを入力したら、このボタンにより、テスト認証要求を ACS サーバに送信できます。

1. **> 設定されるリモートアクセス VPN > AAA > AAA サーバ グループ** 『Configuration』 を選択 して下さい。
2. 最上部のペインで対象の AAA サーバ グループを選択します。
3. 下部のペインでテストする AAA サーバを選択します。
4. 下部のペインの右側にある **Test** ボタンをクリックします。

5. 表示されるウィンドウで、[Authentication] オプション ボタンをクリックして、テスト対象のクレデンシャルを入力します。 完了したら、[OK] をクリックします。



6. ASA から AAA サーバへのコンタクトの後で、成功メッセージか失敗メッセージが表示されます。



CLI でのテスト

AAA 設定をテストするためにコマンドラインで **test** コマンドを使用できます。 テスト要求が AAA サーバに送信され、コマンド ラインに結果が表示されます。

```
ciscoasa#test aaa-server authentication RAD_SVR_GRP host 192.168.1.2 username kate password
cisco123 INFO: Attempting Authentication test to IP address <192.168.1.2> (timeout: 12 seconds)
INFO: Authentication Successful
```

トラブルシューティング

このシナリオでの認証のトラブルシューティングには、**debug radius** コマンドが有効です。 このコマンドにより RADIUS セッションのデバッグがイネーブルになり、さらに RADIUS パケット

のデコードもイネーブルになります。 提示される各デバッグ出力では、デコードされた最初のパケットが、ASA から ACS サーバに送信されたパケットになっています。 2 番目のパケットは ACS サーバからの応答です。

注: debug コマンドを使用する前に、『**debug コマンドの重要な情報**』を参照してください。

認証が成功すると、RADIUS サーバから **access-accept** メッセージが送信されます。

```
ciscoasa#debug radius !--- First Packet. Authentication Request. ciscoassa#radius mkreq: 0x88
alloc_rip 0xd5627ae4 new request 0x88 --> 52 (0xd5627ae4) got user '' got password add_req
0xd5627ae4 session 0x88 id 52 RADIUS_REQUEST radius.c: rad_mkpkt RADIUS packet decode
(authentication request) ------------------------------------ Raw packet data (length =
62)..... 01 34 00 3e 18 71 56 d7 c4 ad e2 73 30 a9 2e cf | .4.>.qV....s0... 5c 65 3a eb 01 06 6b
61 74 65 02 12 0e c1 28 b7 | \e:...kate....(. 87 26 ed be 7b 2c 7a 06 7c a3 73 19 04 06 c0 a8 |
.&..{,z.|.s..... 01 01 05 06 00 00 00 34 3d 06 00 00 00 05 | .......4=..... Parsed packet
data..... Radius: Code = 1 (0x01) Radius: Identifier = 52 (0x34) Radius: Length = 62 (0x003E)
Radius: Vector: 187156D7C4ADE27330A92ECF5C653AEB Radius: Type = 1 (0x01) User-Name Radius:
Length = 6 (0x06) Radius: Value (String) = 6b 61 74 65 | kate Radius: Type = 2 (0x02) User-
Password Radius: Length = 18 (0x12) Radius: Value (String) = 0e c1 28 b7 87 26 ed be 7b 2c 7a 06
7c a3 73 19 | ..(..&..{,z.|.s. Radius: Type = 4 (0x04) NAS-IP-Address Radius: Length = 6 (0x06)
Radius: Value (IP Address) = 192.168.1.1 (0xC0A80101) Radius: Type = 5 (0x05) NAS-Port Radius:
Length = 6 (0x06) Radius: Value (Hex) = 0x34 Radius: Type = 61 (0x3D) NAS-Port-Type Radius:
Length = 6 (0x06) Radius: Value (Hex) = 0x5 send pkt 192.168.1.2/1645 rip 0xd5627ae4 state 7 id
52 rad_vrfy() : response message verified rip 0xd544d2e8 : chall_state '' : state 0x7 : timer
0x0 : reqauth: 18 71 56 d7 c4 ad e2 73 30 a9 2e cf 5c 65 3a eb : info 0x88 session_id 0x88
request_id 0x34 user 'kate' response '***' app 0 reason 0 skey 'secretkey' sip 192.168.1.2 type
1 !--- Second Packet. Authentication Response. RADIUS packet decode (response) ----------------
---------------------- Raw packet data (length = 50)..... 02 34 00 32 35 a1 88 2f 8a bf 2a 14 c5
31 78 59 | .4.25../..*..1xY 60 31 35 89 08 06 ff ff ff ff 19 18 43 41 43 53 | `15.........CACS
3a 30 2f 32 61 36 2f 63 30 61 38 30 31 30 31 2f | :0/2a6/c0a80101/ 35 32 | 52 Parsed packet
data..... Radius: Code = 2 (0x02) Radius: Identifier = 52 (0x34) Radius: Length = 50 (0x0032)
Radius: Vector: 35A1882F8ABF2A14C531785960313589 Radius: Type = 8 (0x08) Framed-IP-Address
Radius: Length = 6 (0x06) Radius: Value (IP Address) = 255.255.255.255 (0xFFFFFFFF) Radius: Type
= 25 (0x19) Class Radius: Length = 24 (0x18) Radius: Value (String) = 43 41 43 53 3a 30 2f 32 61
36 2f 63 30 61 38 30 | CACS:0/2a6/c0a80 31 30 31 2f 35 32 | 101/52 rad_procpkt: ACCEPT
RADIUS_ACCESS_ACCEPT: normal termination RADIUS_DELETE remove_req 0xd5627ae4 session 0x88 id 52
free_rip 0xd5627ae4 radius: send queue empty
```

認証が失敗すると、ACS サーバから **access-reject** メッセージが送信されます。

```
ciscoasa#debug radius !--- First Packet. Authentication Request. ciscoasa# radius mkreq: 0x85
alloc_rip 0xd5627ae4 new request 0x85 --> 49 (0xd5627ae4) got user '' got password add_req
0xd5627ae4 session 0x85 id 49 RADIUS_REQUEST radius.c: rad_mkpkt RADIUS packet decode
(authentication request) ------------------------------------ Raw packet data (length =
62)..... 01 31 00 3e 88 21 46 07 34 5d d2 a3 a0 59 1e ff | .1.>.!F.4]...Y.. cc 15 2a 1b 01 06 6b
61 74 65 02 12 60 eb 05 32 | ..*...kate..`..2 87 69 78 a3 ce d3 80 d8 4b 0d c3 37 04 06 c0 a8 |
.ix.....K..7.... 01 01 05 06 00 00 00 31 3d 06 00 00 00 05 | .......1=..... Parsed packet
data..... Radius: Code = 1 (0x01) Radius: Identifier = 49 (0x31) Radius: Length = 62 (0x003E)
Radius: Vector: 88214607345DD2A3A0591EFFCC152A1B Radius: Type = 1 (0x01) User-Name Radius:
Length = 6 (0x06) Radius: Value (String) = 6b 61 74 65 | kate Radius: Type = 2 (0x02) User-
Password Radius: Length = 18 (0x12) Radius: Value (String) = 60 eb 05 32 87 69 78 a3 ce d3 80 d8
4b 0d c3 37 | `..2.ix.....K..7 Radius: Type = 4 (0x04) NAS-IP-Address Radius: Length = 6 (0x06)
Radius: Value (IP Address) = 192.168.1.1 (0xC0A80101) Radius: Type = 5 (0x05) NAS-Port Radius:
Length = 6 (0x06) Radius: Value (Hex) = 0x31 Radius: Type = 61 (0x3D) NAS-Port-Type Radius:
Length = 6 (0x06) Radius: Value (Hex) = 0x5 send pkt 192.168.1.2/1645 rip 0xd5627ae4 state 7 id
49 rad_vrfy() : response message verified rip 0xd544d2e8 : chall_state '' : state 0x7 : timer
0x0 : reqauth: 88 21 46 07 34 5d d2 a3 a0 59 1e ff cc 15 2a 1b : info 0x85 session_id 0x85
request_id 0x31 user 'kate' response '***' app 0 reason 0 skey 'secretkey' sip 192.168.1.2 type
1 !--- Second packet. Authentication Response. RADIUS packet decode (response) ----------------
---------------------- Raw packet data (length = 32)..... 03 31 00 20 70 98 50 af 39 cc b9 ba df
a7 bd ff | .1. p.P.9....... 06 af fb 02 12 0c 52 65 6a 65 63 74 65 64 0a 0d | ......Rejected..
Parsed packet data..... Radius: Code = 3 (0x03) Radius: Identifier = 49 (0x31) Radius: Length =
32 (0x0020) Radius: Vector: 709850AF39CCB9BADFA7BDFF06AFFB02 Radius: Type = 18 (0x12) Reply-
```

Message Radius: Length = 12 (0x0C) **Radius: Value (String) = 52 65 6a 65 63 74 65 64 0a 0d |** **Rejected.. rad_procpkt: REJECT** RADIUS_DELETE remove_req 0xd5627ae4 session 0x85 id 49 free_rip 0xd5627ae4 radius: send queue empty

関連情報

- Remote Authentication Dial-In User Service（RADIUS）
- Requests for Comments (RFC)
- テクニカルサポートとドキュメント - Cisco Systems