

Configure Remote Access VPN with RADIUS Authentication on ISE and Group-Policy Mapping

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components used](#)

[Configure](#)

[Configuration](#)

[ASA](#)

[ISE](#)

[Verify](#)

[Working Scenario](#)

[Troubleshoot](#)

[Non-working Scenario 1](#)

[Non-working Scenario 2](#)

[Non-working Scenario 3](#)

[Video](#)

Introduction

This document describes configuring Remote Access VPN for group-policy mapping with Cisco Identity Services Engine (ISE).

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Cisco Secure Client (AnyConnect)
- Cisco ISE
- Remote Access VPN on Cisco Adaptive Security Appliance (ASA)

Components used

The content of this document is based on these software and hardware versions.

- ASA 5506 with Software Version 9.8.1
- AnyConnect Version 4.8
- ISE Version 2.4.

The information in this document was created from the devices in a specific lab environment. All of the

devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Configure

In this configuration example, remote users connecting to the ASA via VPN using Cisco Secure Client (AnyConnect) are not allowed to select a connection profile (tunnel-group) from the drop-down menu, as Cisco ISE maps them to a specific Group-Policy based on the configured policies.

With this setup, you can assign a group-policy to each AnyConnect user through ISE. Since the users do not have the option to select the tunnel group, they are initially connected to the **DefaultWEBVPNGroup** tunnel-group and the **DfltGrpPolicy** group-policy. After authentication, if the RADIUS Class attribute (Group-policy) is sent by ISE within the authentication response, the user is assigned to the corresponding group-policy, thereby receiving the appropriate permissions. If ISE does not return any Class attribute or returns a group-policy label that is not configured on the ASA, the user remains assigned to the DfltGrpPolicy. To prevent users without an assigned group-policy from connecting through the VPN, you can configure the **vpn-simultaneous-logins 0** command under the **DfltGrpPolicy** group-policy.

Configuration

ASA

AAA-Server

```
aaa-server ISE_AAA protocol radius
aaa-server ISE_AAA (Outside) host 10.31.124.82
key cisco123
```

Remote Access VPN Configuration

```
webvpn
enable outside
anyconnect image disk0:/anyconnect-win-4.7.01076-webdeploy-k9.pkg 1
anyconnect enable

tunnel-group DefaultWEBVPNGroup general-attributes
address-pool Remote_users
authentication-server-group ISE_AAA

group-policy DfltGrpPolicy attributes
banner value ###YOU DON'T HAVE AUTHORIZATION TO ACCESS ANY INTERNAL RESOURCES###
vpn-simultaneous-logins 0
vpn-tunnel-protocol ssl-client

group-policy RADIUS-USERS internal
group-policy RADIUS-USERS attributes
banner value YOU ARE CONNECTED TO ### RADIUS USER AUTHENTICATION###
vpn-simultaneous-logins 3
vpn-tunnel-protocol ssl-client
split-tunnel-network-list value SPLIT_ACL
```

```
group-policy RADIUS-ADMIN internal
group-policy RADIUS-ADMIN attributes
banner value YOU ARE CONNECTED TO ###RADIUS ADMIN AUTHENTICATION ###
vpn-simultaneous-logins 3
vpn-tunnel-protocol ssl-client
split-tunnel-network-list none
```

ISE

Step1. Register the ASA as a valid network device on ISE and configure the shared secret key for RADIUS. For this, navigate to **Administration>Network Resources>Network Devices**.

Identity Services Engine

Home > Context Visibility > Operations > Policy > Administration > Work Centers

System > Identity Management > Network Resources > Device Portal Management > pxGrid Services > Feed Service > Threat Centric NAC

Network Devices > Network Device Groups > Network Device Profiles > External RADIUS Servers > RADIUS Server Sequences > NAC Managers > External MDM > Location Services

Network Devices

Default Device

Device Security Settings

Network Devices List > ASAv

Network Devices

* Name ASAv

Description

IP Address * IP : 10.31.124.85 / 32

* Device Profile Cisco

Model Name ASAv

Software Version 9.9

* Network Device Group

Location All Locations Set To Default

IPSEC No Set To Default

Device Type All Device Types Set To Default

RADIUS Authentication Settings

RADIUS UDP Settings

Protocol RADIUS

* Shared Secret cisco123 Hide

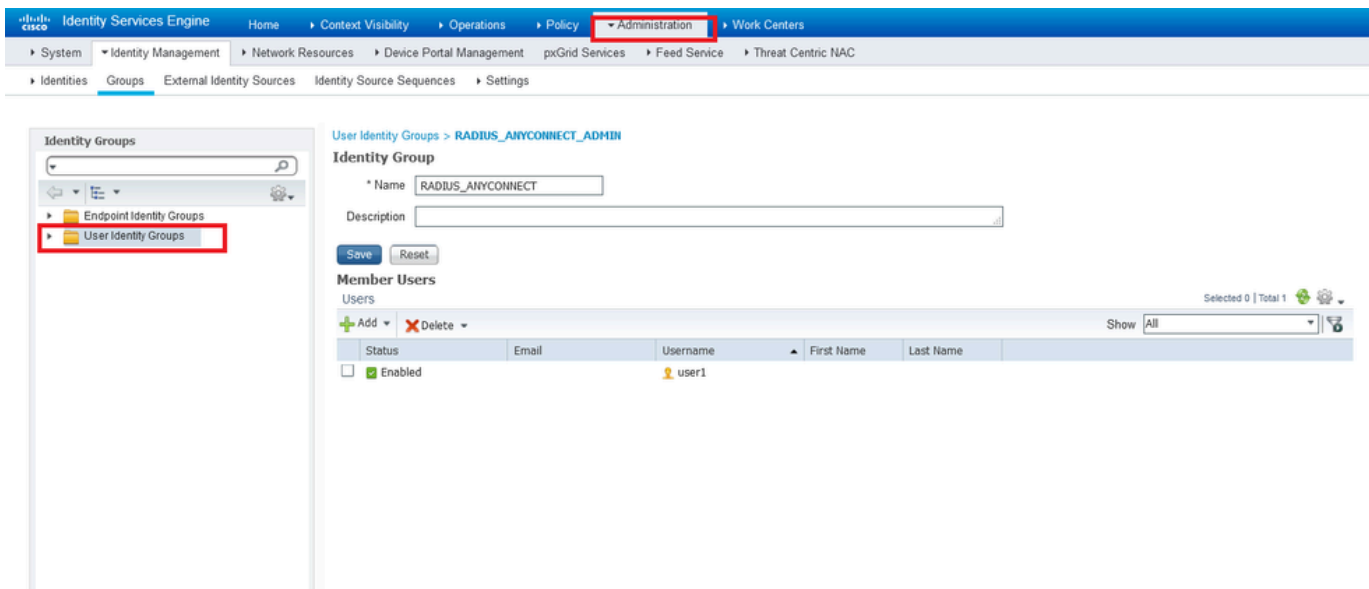
Use Second Shared Secret Show

CoA Port 1700 Set To Default

RADIUS DTLS Settings

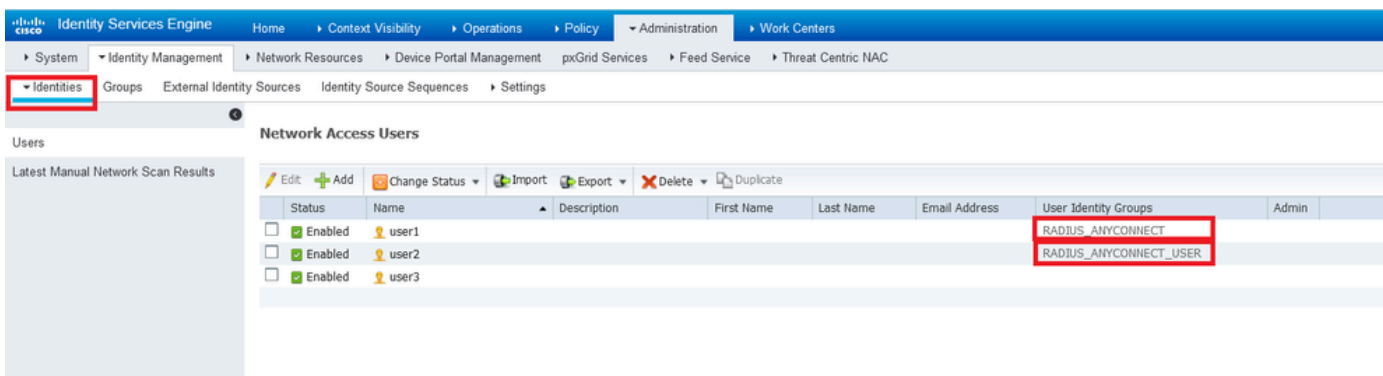
Step 2. Create identity groups.

Define identity groups to associate users with similar characteristics and who share similar permissions. These are used in the next steps. Navigate to **Administration>Groups>User Identity Groups**.



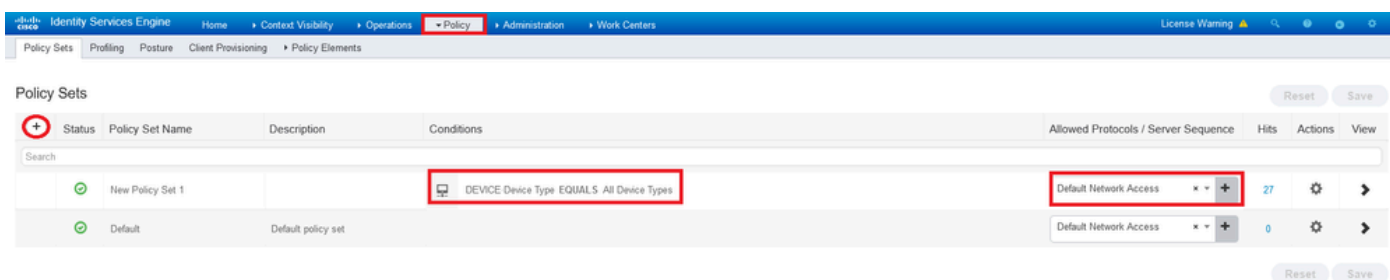
Step 3. Associate users to identity groups.

Associate users to the right identity group. Navigate to **Administration>Identities>Users**.



Step 4. Create Policy Set.

Define a new policy set and define the conditions that match the policy. In this example, all device types are allowed under the conditions. For this, navigate to **Policy>Policy sets**.



Step 5. Create an Authorization Policy.

Define a new Authorization Policy with the required conditions to match the policy. Ensure to include the identity groups created in step 2 as a condition.

Identity Services Engine Home Context Visibility Operations Policy Administration Work Centers License Warning

Policy Sets Profiling Posture Client Provisioning Policy Elements

Policy Sets → New Policy Set 1 Reset Save

Status	Policy Set Name	Description	Conditions	Allowed Protocols / Server Sequence	Hits
🟢	New Policy Set 1		DEVICE Device Type EQUALS All Device Types	Default Network Access	27

Authentication Policy (1)
 Authorization Policy - Local Exceptions
 Authorization Policy - Global Exceptions
 Authorization Policy (3)


Status	Rule Name	Conditions	Results		Hits	Actions
			Profiles	Security Groups		
🟢	ISE_CLASS_ADMIN	AND DEVICE Device Type EQUALS All Device Types IdentityGroup-Name EQUALS User Identity Groups:RADIUS_ANYCONNECT	Select from list	Select from list	7	⚙️
🟢	ISE_CLASS_USER	AND DEVICE Device Type EQUALS All Device Types IdentityGroup-Name EQUALS User Identity Groups:RADIUS_ANYCONNECT_USER	Select from list	Select from list	9	⚙️
🟢	Default		DenyAccess	Select from list	8	⚙️

Reset Save

Step 6. Create an Authorization Profile.

The authorization profile includes the actions that are taken when the authorization policy is matched. Create a new Authorization Profile that includes the next attributes:

- RADIUS Class = <Group-policy-ASA>
- Access Type: ACCESS_ACCEPT.

 **Note:** You must edit the configuration displayed in the previous images to match the name of the group-policies you defined in your ASA configuration.

Status	Rule Name	Conditions	Results		Hits	Actions
			Profiles	Security Groups		
🟢	ISE_CLASS_ADMIN	AND DEVICE Device Type EQUALS All Device Types IdentityGroup-Name EQUALS User Identity Groups:RADIUS_ANYCONNECT	Select from list Create a New Authorization Profile	Select from list	7	⚙️
🟢	ISE_CLASS_USER	AND DEVICE Device Type EQUALS All Device Types IdentityGroup-Name EQUALS User Identity Groups:RADIUS_ANYCONNECT_USER	Select from list	Select from list	9	⚙️
🟢	Default		DenyAccess	Select from list	8	⚙️

Add New Standard Profile

Authorization Profile

Name: CLAS_25_RADIUS_ADMIN

Description:

Access Type: ACCESS_ACCEPT

Network Device Profile: Cisco

Service Template:

Track Movement:

Passive Identity Tracking:

Common Tasks

Advanced Attributes Settings

Radius:Class = RADIUS-ADMIN

Attributes Details

Access Type = ACCESS_ACCEPT
Class = RADIUS-ADMIN

Save Cancel

This should be the Group-policy name

Step 7. Review the Authorization Profile configuration.

Identity Services Engine Home Context Visibility Operations Policy Administration Work Centers

Policy Sets Profiling Posture Client Provisioning Policy Elements

Dictionarys Conditions Results

Authentication

Authorization

Authorization Profiles

Downloadable ACLs

Profiling

Posture

Client Provisioning

Authorization Profile

* Name CLASS_25_RADIUS_ADMIN

Description

* Access Type ACCESS_ACCEPT

Network Device Profile Cisco

Service Template

Track Movement

Passive Identity Tracking

Common Tasks


Advanced Attributes Settings

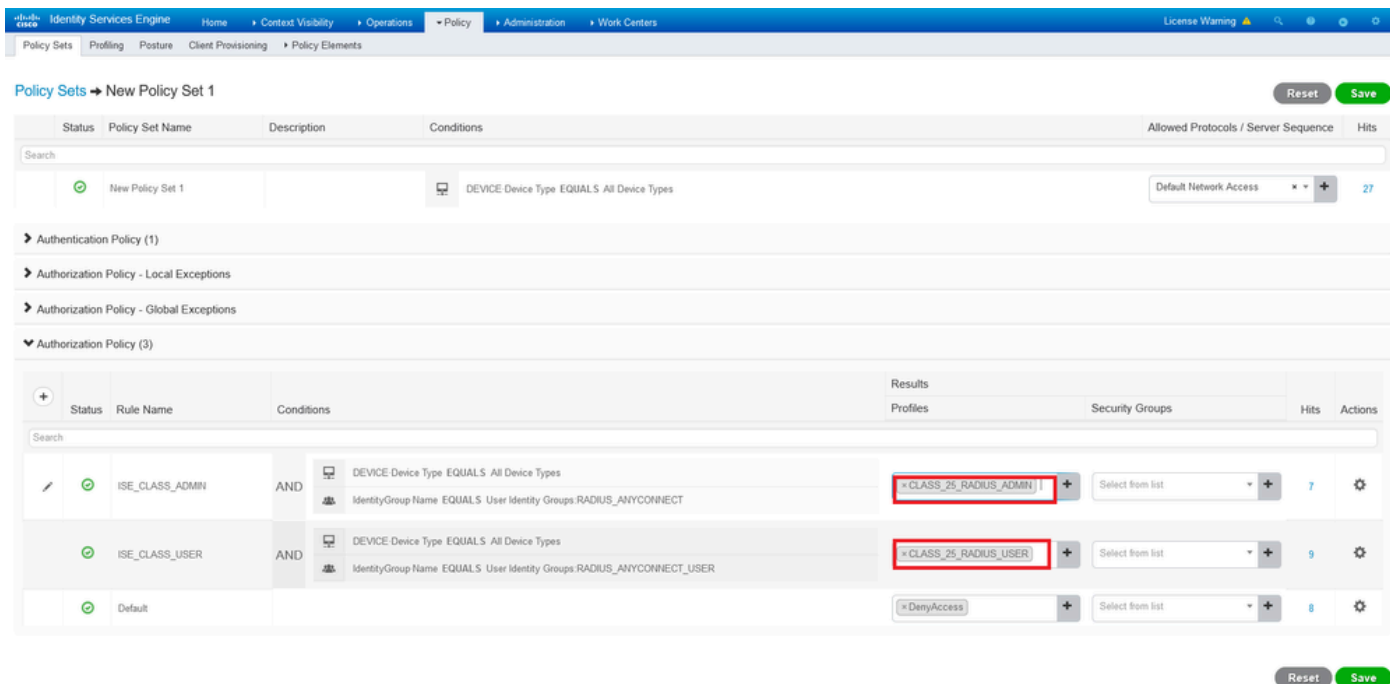
Radius:Class = RADIUS-ADMIN

Attributes Details

Access Type = ACCESS_ACCEPT
Class = RADIUS-ADMIN

Save Reset

 **Note:** In the same policy set, you can have **n** authorization policies, to map each identity group to a specific group-policy defined on the ASA..



With this configuration example, you can assign the group policy dynamically to each Secure Client user through ISE configuration based on the identity group the user belongs to.

Verify

One of the most useful debugs is **debug radius**. It shows details of the radius authentication request and authentication response between the AAA server (ISE) and the ASA.

debug radius

Another useful tool is the command **test aaa-server**. You now see if the authentication is **ACCEPTED** or **REFUSED** and the attributes ('class' attribute in this example) that were exchanged in the authentication process.

```
test aaa-server authentication <aaa_server_group> [host <name>|<host_ip>] username <user> password <pas
```

Working Scenario

In the configuration example mentioned before, **user1** belongs to the **RADIUS-ADMIN** group policy per the ISE configuration. It can be verified if you run the test aaa-server and enable radius debugs on the ASA. The relevant lines from the debugs are marked in bold.

```
<#root>
```

```
ASAv# debug radius
ASAv#test aaa-server authentication ISE_AAA host 10.31.124.82 username user1 password *****
INFO: Attempting Authentication test to IP address (10.31.124.82) (timeout: 12 seconds)
```


RADIUS packet decode (authentication request)

```
-----  
Raw packet data (length = 84).....  
01 1e 00 54 ac b6 7c e5 58 22 35 5e 8e 7c 48 73 | ...T..|.X"5^.|Hs  
04 9f 8c 74 01 07 75 73 65 72 31 02 12 ad 19 1c | ...t..user1.....  
40 da 43 e2 ba 95 46 a7 35 85 52 bb 6f 04 06 0a | @.C...F.5.R.o...  
1f 7c 55 05 06 00 00 00 06 3d 06 00 00 00 05 1a | .|U.....=.....  
15 00 00 00 09 01 0f 63 6f 61 2d 70 75 73 68 3d | .....coa-push=  
74 72 75 65 | true
```

```
Parsed packet data.....  
Radius: Code = 1 (0x01)  
Radius: Identifier = 30 (0x1E)  
Radius: Length = 84 (0x0054)  
Radius: Vector: ACB67CE55822355E8E7C4873049F8C74  
Radius: Type = 1 (0x01) User-Name  
Radius: Length = 7 (0x07)  
Radius: Value (String) =  
75 73 65 72 31 |
```

user1

```
Radius: Type = 2 (0x02) User-Password  
Radius: Length = 18 (0x12)  
Radius: Value (String) =  
ad 19 1c 40 da 43 e2 ba 95 46 a7 35 85 52 bb 6f | ...@.C...F.5.R.o  
Radius: Type = 4 (0x04) NAS-IP-Address  
Radius: Length = 6 (0x06)  
Radius: Value (IP Address) = 10.31.124.85 (0x0A1F7C55)  
Radius: Type = 5 (0x05) NAS-Port  
Radius: Length = 6 (0x06)  
Radius: Value (Hex) = 0x6  
Radius: Type = 61 (0x3D) NAS-Port-Type  
Radius: Length = 6 (0x06)  
Radius: Value (Hex) = 0x5  
Radius: Type = 26 (0x1A) Vendor-Specific  
Radius: Length = 21 (0x15)  
Radius: Vendor ID = 9 (0x00000009)  
Radius: Type = 1 (0x01) Cisco-AV-pair  
Radius: Length = 15 (0x0F)  
Radius: Value (String) =  
63 6f 61 2d 70 75 73 68 3d 74 72 75 65 | coa-push=true  
send pkt 10.31.124.82/1645  
rip 0x00007f03b419fb08 state 7 id 30  
rad_vrfy() : response message verified  
rip 0x00007f03b419fb08  
: chall_state ''  
: state 0x7  
: reqauth:  
ac b6 7c e5 58 22 35 5e 8e 7c 48 73 04 9f 8c 74  
: info 0x00007f03b419fc48  
session_id 0x80000007  
request_id 0x1e  
user 'user1'  
response '***'  
app 0  
reason 0  
skey 'cisco123'  
sip 10.31.124.82  
type 1
```

RADIUS packet decode (response)

Raw packet data (length = 188).....

02 1e 00 bc 9e 5f 7c db ad 63 87 d8 c1 bb 03 41	_ ..C.....A
37 3d 7a 35 01 07 75 73 65 72 31 18 43 52 65 61		7=z5..user1.CRea
75 74 68 53 65 73 73 69 6f 6e 3a 30 61 31 66 37		uthSession:0a1f7
63 35 32 52 71 51 47 52 72 70 36 5a 35 66 4e 4a		c52RqQGRrp6Z5fNJ
65 4a 39 76 4c 54 6a 73 58 75 65 59 35 4a 70 75		eJ9vLTjsXueY5Jpu
70 44 45 61 35 36 34 66 52 4f 44 57 78 34 19 0e		pDEa564fRODwx4..
52 41 44 49 55 53 2d 41 44 4d 49 4e 19 50 43 41		RADIUS-ADMIN.PCA
43 53 3a 30 61 31 66 37 63 35 32 52 71 51 47 52		CS:0a1f7c52RqQGR
72 70 36 5a 35 66 4e 4a 65 4a 39 76 4c 54 6a 73		rp6Z5fNJeJ9vLTjs
58 75 65 59 35 4a 70 75 70 44 45 61 35 36 34 66		XueY5JpupDEa564f
52 4f 44 57 78 34 3a 69 73 65 61 6d 79 32 34 2f		RODwx4:iseamy24/
33 37 39 35 35 36 37 34 35 2f 33 31		379556745/31

Parsed packet data.....

Radius: Code = 2 (0x02)

Radius: Identifier = 30 (0x1E)

Radius: Length = 188 (0x00BC)

Radius: Vector: 9E5F7CDBAD6387D8C1BB0341373D7A35

Radius: Type = 1 (0x01) User-Name

Radius: Length = 7 (0x07)

Radius: Value (String) =

75 73 65 72 31

user1

Radius: Type = 24 (0x18) State

Radius: Length = 67 (0x43)

Radius: Value (String) =

52 65 61 75 74 68 53 65 73 73 69 6f 6e 3a 30 61		ReauthSession:0a
31 66 37 63 35 32 52 71 51 47 52 72 70 36 5a 35		1f7c52RqQGRrp6Z5
66 4e 4a 65 4a 39 76 4c 54 6a 73 58 75 65 59 35		fNJeJ9vLTjsXueY5
4a 70 75 70 44 45 61 35 36 34 66 52 4f 44 57 78		JpupDEa564fRODwx
34		4

Radius: Type = 25 (0x19) Class

Radius: Length = 14 (0x0E)

Radius: Value (String) =

52 41 44 49 55 53 2d 41 44 4d 49 4e

RADIUS-ADMIN

Radius: Type = 25 (0x19) Class

Radius: Length = 80 (0x50)

Radius: Value (String) =

43 41 43 53 3a 30 61 31 66 37 63 35 32 52 71 51		CACS:0a1f7c52RqQ
47 52 72 70 36 5a 35 66 4e 4a 65 4a 39 76 4c 54		GRrp6Z5fNJeJ9vLT
6a 73 58 75 65 59 35 4a 70 75 70 44 45 61 35 36		jsXueY5JpupDEa56
34 66 52 4f 44 57 78 34 3a 69 73 65 61 6d 79 32		4fRODwx4:iseamy2
34 2f 33 37 39 35 35 36 37 34 35 2f 33 31		4/379556745/31

rad_procpkt: ACCEPT

RADIUS_ACCESS_ACCEPT

: normal termination

RADIUS_DELETE

```
remove_req 0x00007f03b419fb08 session 0x80000007 id 30
free_rip 0x00007f03b419fb08
radius: send queue empty
```

```
INFO: Authentication Successful
```

Another way to verify if user1 was assigned the correct group policy by ISE when connected via Secure Client is with the **show vpn-sessiondb anyconnect** command.

```
<#root>
```

```
ASAv#
```

```
show vpn-sessiondb anyconnect
```

```
Session Type: AnyConnect
```

```
Username      : user1
```

```
                Index      : 28
Assigned IP   : 10.100.2.1      Public IP      : 10.100.1.3
Protocol      : AnyConnect-Parent SSL-Tunnel DTLS-Tunnel
License       : AnyConnect Premium
Encryption    : AnyConnect-Parent: (1)none SSL-Tunnel: (1)AES-GCM-256 DTLS-Tunnel: (1)AES256
Hashing       : AnyConnect-Parent: (1)none SSL-Tunnel: (1)SHA384 DTLS-Tunnel: (1)SHA1
Bytes Tx      : 15604           Bytes Rx       : 28706
```

```
Group Policy  : RADIUS-ADMIN
```

```
Tunnel Group : DefaultWEBVPNGroup
```

```
Login Time    : 04:14:45 UTC Wed Jun 3 2020
Duration      : 0h:01m:29s
Inactivity    : 0h:00m:00s
VLAN Mapping  : N/A           VLAN           : none
Audt Sess ID  : 0a6401010001c0005ed723b5
Security Grp  : none
```

Troubleshoot

You can also use the **debug radius** and **test aaa-server** commands to troubleshoot when issues occur. The most common issues are described next.

Non-working Scenario 1

If the Authentication fails on Anyconnect and the ISE replies with a REJECT. You need to verify either the user is associated with a **User Identity Group** or the password is incorrect. Navigate to **Operations>Live logs > Details**.

```
<#root>
```

```
RADIUS packet decode (response)
```

```
-----
Raw packet data (length = 20).....
03 21 00 14 dd 74 bb 43 8f 0a 40 fe d8 92 de 7a | .!...t.C..@....z
27 66 15 be | 'f..
```

```
Parsed packet data.....
Radius: Code = 3 (0x03)
Radius: Identifier = 33 (0x21)
Radius: Length = 20 (0x0014)
Radius: Vector: DD74BB438F0A40FED892DE7A276615BE
rad_procpkt:
```

REJECT

```
RADIUS_DELETE
remove_req 0x00007f03b419fb08 session 0x80000009 id 33
free_rip 0x00007f03b419fb08
radius: send queue empty
```

ERROR: Authentication Rejected: AAA failure

The screenshot displays the Identity Services Engine interface. On the left, the 'Overview' section shows a table with the following data:

Event	5400 Authentication failed
Username	user1
Endpoint Id	
Endpoint Profile	
Authentication Policy	New Policy Set 1 >> Default
Authorization Policy	New Policy Set 1 >> Default
Authorization Result	DenyAccess

Below this, the 'Authentication Details' section shows:

Source Timestamp	2020-06-02 23:22:53.577
Received Timestamp	2020-06-02 23:22:53.577
Policy Server	iseamy24
Event	5400 Authentication failed
Failure Reason	15039 Rejected per authorization profile

On the right side, the 'Steps' section lists a sequence of events:

- 11001 Received RADIUS Access-Request
- 11017 RADIUS created a new session
- 11117 Generated a new session ID
- 15049 Evaluating Policy Group
- 15008 Evaluating Service Selection Policy
- 15048 Queried PIP - DEVICE.Device Type
- 15041 Evaluating Identity Policy
- 22072 Selected identity source sequence - All_User_ID_Stores
- 15013 Selected Identity Source - Internal Users
- 24210 Looking up User in Internal Users IDStore - user1
- 24212 Found User in Internal Users IDStore
- 22037 Authentication Passed
- 15036 Evaluating Authorization Policy
- 15048 Queried PIP - DEVICE.Device Type
- 15048 Queried PIP - Network Access.UserName
- 15048 Queried PIP - IdentityGroup.Name
- 15016 Selected Authorization Profile - DenyAccess
- 15039 Rejected per authorization profile
- 11003 Returned RADIUS Access-Reject

Note: In this example, **user1** is not associated with any **User Identity Group**. Therefore, it hits the Default Authentication and Authorization policies under the **New Policy Set 1** with the **DenyAccess** action. You can modify this action to **PermitAccess** in the Default Authorization Policy to allow the users without the User identity group associated to authenticate.

Non-working Scenario 2

If the Authentication fails on Anyconnect and the default Authorization policy is PermitAccess, the authentication is accepted. However, the class attribute is not presented in the Radius response, therefore the user is located in the DfltGrpPolicy and it does not connect due to the configured command: **vpn-simultaneous-logins 0**.

<#root>

RADIUS packet decode (response)

Raw packet data (length = 174).....

02 24 00 ae 5f 0f bc b1 65 53 64 71 1a a3 bd 88		.\$...eSdq....
7c fe 44 eb 01 07 75 73 65 72 31 18 43 52 65 61		.D...user1.CRea
75 74 68 53 65 73 73 69 6f 6e 3a 30 61 31 66 37		uthSession:0a1f7
63 35 32 32 39 54 68 33 47 68 6d 44 54 49 35 71		c5229Th3GhmDTI5q
37 48 46 45 30 7a 6f 74 65 34 6a 37 50 76 69 4b		7HFE0zote4j7PviK
5a 35 77 71 6b 78 6c 50 39 33 42 6c 4a 6f 19 50		Z5wqkx1P93B1Jo.P
43 41 43 53 3a 30 61 31 66 37 63 35 32 32 39 54		CACS:0a1f7c5229T
68 33 47 68 6d 44 54 49 35 71 37 48 46 45 30 7a		h3GhmDTI5q7HFE0z
6f 74 65 34 6a 37 50 76 69 4b 5a 35 77 71 6b 78		ote4j7PviKZ5wqkx
6c 50 39 33 42 6c 4a 6f 3a 69 73 65 61 6d 79 32		1P93B1Jo:iseamy2
34 2f 33 37 39 35 35 36 37 34 35 2f 33 37		4/379556745/37

Parsed packet data.....

Radius: Code = 2 (0x02)
 Radius: Identifier = 36 (0x24)
 Radius: Length = 174 (0x00AE)
 Radius: Vector: 5F0FBCB1655364711AA3BD887CFE44EB
 Radius: Type = 1 (0x01) User-Name
 Radius: Length = 7 (0x07)
 Radius: Value (String) =
 75 73 65 72 31

user1

Radius: Type = 24 (0x18) State
 Radius: Length = 67 (0x43)
 Radius: Value (String) =
 52 65 61 75 74 68 53 65 73 73 69 6f 6e 3a 30 61 | ReauthSession:0a
 31 66 37 63 35 32 32 39 54 68 33 47 68 6d 44 54 | 1f7c5229Th3GhmDT
 49 35 71 37 48 46 45 30 7a 6f 74 65 34 6a 37 50 | I5q7HFE0zote4j7P
 76 69 4b 5a 35 77 71 6b 78 6c 50 39 33 42 6c 4a | viKZ5wqkx1P93B1J
 6f | o

Radius: Type = 25 (0x19) Class
 Radius: Length = 80 (0x50)
 Radius: Value (String) =
 43 41 43 53 3a 30 61 31 66 37 63 35 32 32 39 54 | CACS:0a1f7c5229T
 68 33 47 68 6d 44 54 49 35 71 37 48 46 45 30 7a | h3GhmDTI5q7HFE0z
 6f 74 65 34 6a 37 50 76 69 4b 5a 35 77 71 6b 78 | ote4j7PviKZ5wqkx
 6c 50 39 33 42 6c 4a 6f 3a 69 73 65 61 6d 79 32 | 1P93B1Jo:iseamy2
 34 2f 33 37 39 35 35 36 37 34 35 2f 33 37 | 4/379556745/37

rad_procpkt: ACCEPT

RADIUS_ACCESS_ACCEPT

: normal termination
 RADIUS_DELETE
 remove_req 0x00007f03b419fb08 session 0x8000000b id 36
 free_rip 0x00007f03b419fb08
 radius: send queue empty

INFO: Authentication Successful

ASAv#

If the **vpn-simultaneous-logins 0** is changed to '1', The user connects as shown in the output:

<#root>

ASAv# show vpn-sessiondb anyconnect

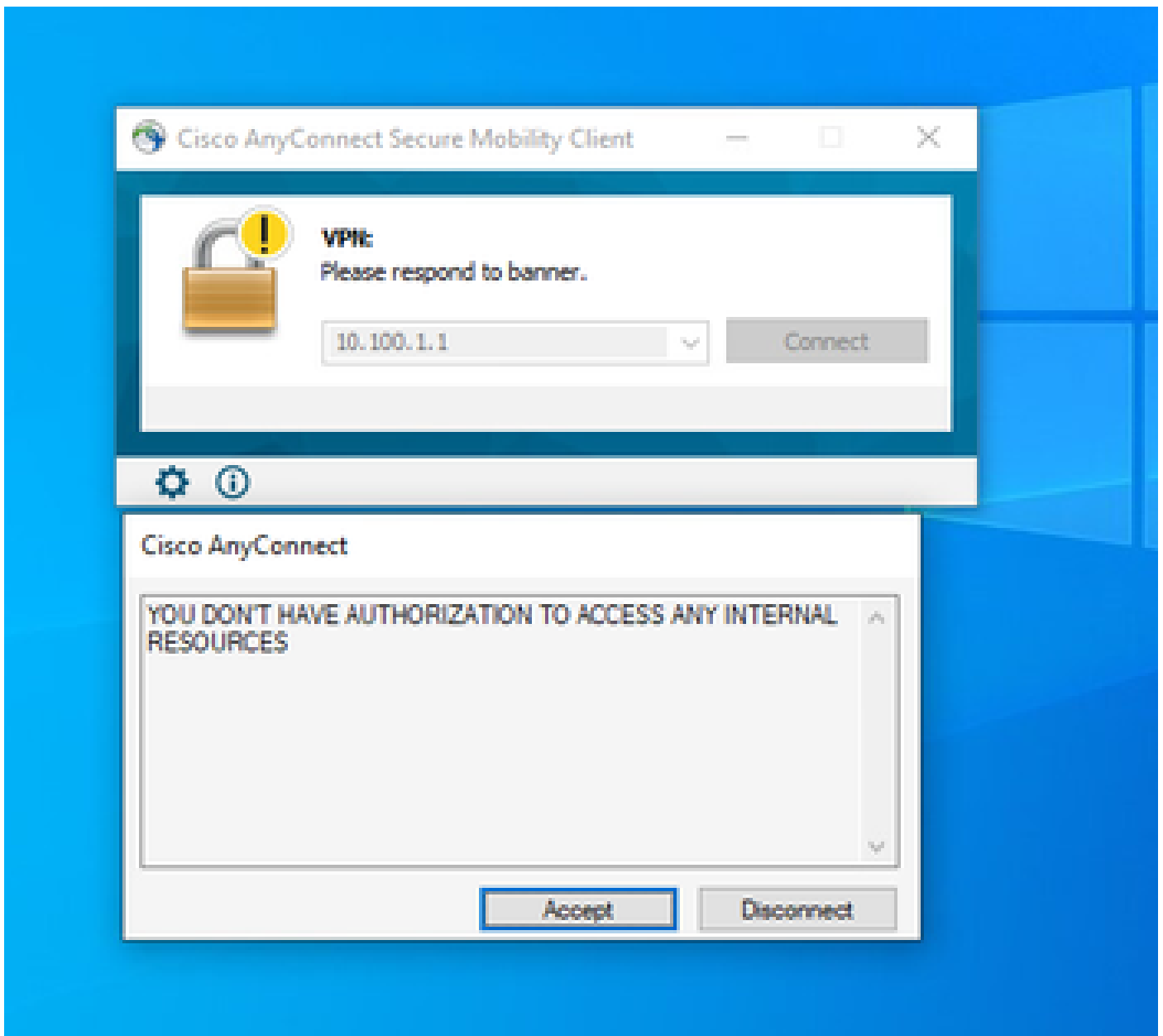
Session Type: AnyConnect

Username : user1

Index : 41
Assigned IP : 10.100.2.1 Public IP : 10.100.1.3
Protocol : AnyConnect-Parent SSL-Tunnel DTLS-Tunnel
License : AnyConnect Premium
Encryption : AnyConnect-Parent: (1)none SSL-Tunnel: (1)AES-GCM-256 DTLS-Tunnel: (1)AES256
Hashing : AnyConnect-Parent: (1)none SSL-Tunnel: (1)SHA384 DTLS-Tunnel: (1)SHA1
Bytes Tx : 15448 Bytes Rx : 15528

Group Policy : DfltGrpPolicy Tunnel Group : DefaultWEBVPNGroup

Login Time : 18:43:39 UTC Wed Jun 3 2020
Duration : 0h:01m:40s
Inactivity : 0h:00m:00s
VLAN Mapping : N/A VLAN : none
Audt Sess ID : 0a640101000290005ed7ef5b
Security Grp : none



Non-working Scenario 3

If the Authentication passes but the user does not have the right policies applied, for example, if the group-policy connected has the split tunnel instead of the full tunnel as it must be. The user can be in the wrong User identity group.

```
<#root>
```

```
ASAv# sh vpn-sessiondb anyconnect
```

```
Session Type: AnyConnect
```

```
Username      : user1
```

```
                Index      : 29
Assigned IP    : 10.100.2.1      Public IP      : 10.100.1.3
Protocol       : AnyConnect-Parent SSL-Tunnel
License        : AnyConnect Premium
Encryption     : AnyConnect-Parent: (1)none  SSL-Tunnel: (1)AES-GCM-256
Hashing        : AnyConnect-Parent: (1)none  SSL-Tunnel: (1)SHA384
```

Bytes Tx : 15592 Bytes Rx : 0

Group Policy : RADIUS-USERS

Tunnel Group : DefaultWEBVPNGroup

Login Time : 04:36:50 UTC Wed Jun 3 2020

Duration : 0h:00m:20s

Inactivity : 0h:00m:00s

VLAN Mapping : N/A VLAN : none

Audt Sess ID : 0a6401010001d0005ed728e2

Security Grp : none

Video

This video provides the steps to configure SSL Anyconnect With ISE Authentication And Class Attribute For Group-Policy Mapping.