


AWS VPN for access Kafka [Dev Account]

- [Create Certificate](#)
- [Import certificates to ACM](#)
- [Set AWS VPN](#)
- [Set authorization rule](#)
- [Add target network associations](#)
- [Configure AWS VPN certificate \(This step is required only once when the VPN is new\)](#)
- [Import the VPN certificate to your Tunnelblick](#)
- [Next steps](#)

[Authenticate AWS Client VPN users with federated authentication.](#)

- [How to add or remove user groups to the VPN](#)

Ticket:  [ECCO-991](#) - Getting issue details... [STATUS](#)

We want to explore an AWS VPN as a solution that could help alleviate the difficulties of performing a checkout locally. Therefore, the below steps explain how we can set a VPN to access a Kafka cluster (MKS).

Create Certificate

Since it is for our Dev account and to improve how we can perform a checkout locally, we could generate a certificate using the OpenVPN solution.

1. Clone repository

```
git clone https://github.com/OpenVPN/easy-rsa.git
```

2. Generate a certificate with the below steps

```

#
# Go to the directory where you cloned the repository
#
cd easy-rsa
cd easyrsa3

#
# Init pki and build ca for the server and client
#
./easyrsa init-pki
./easyrsa build-ca nopass
./easyrsa build-server-full server nopass
./easyrsa build-client-full client1.domain.tld nopass

#
# Optional: Create a directory to keep the new certificates
#
mkdir ~/custom_easyrsa_folder

cp pki/ca.crt ~/custom_easyrsa_folder
cp pki/issued/server.crt ~/custom_easyrsa_folder

cp pki/private/server.key ~/custom_easyrsa_folder
cp pki/issued/client1.domain.tld.crt ~/custom_easyrsa_folder
cp pki/private/client1.domain.tld.key ~/custom_easyrsa_folder

#
# Go to the directory where you moved the certificates
#
cd ~/custom_easyrsa_folder

```

Note: we have a .crt and .key for the client and server.

Import certificates to ACM

1. Import server certificate

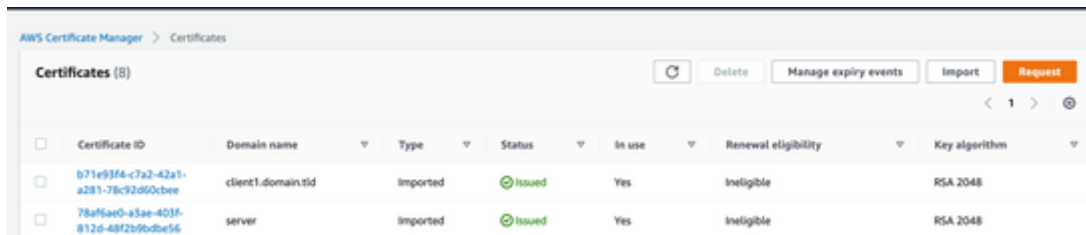
```

aws acm import-certificate --certificate fileb://server.crt --
private-key fileb://server.key --certificate-chain fileb://ca.crt --
region us-east-1

```

2. Import client certificate

```
aws acm import-certificate --certificate fileb://client1.domain.tld.crt --private-key fileb://client1.domain.tld.key --certificate-chain fileb://ca.crt --region us-east-1
```



The screenshot shows the AWS Certificate Manager console. At the top, there's a header 'AWS Certificate Manager > Certificates'. Below it, a section titled 'Certificates (8)' contains buttons for 'Delete', 'Manage expiry events', 'Import', and 'Request'. A table lists the certificates with columns: Certificate ID, Domain name, Type, Status, In use, Renewal eligibility, and Key algorithm. Two certificates are visible: one for 'client1.domain.tld' and another for 'server', both with a status of 'Issued' and 'In use'.

Certificate ID	Domain name	Type	Status	In use	Renewal eligibility	Key algorithm
b71e93f4-c7a2-42a1-a281-78c92d50cbee	client1.domain.tld	Imported	Issued	Yes	Ineligible	RSA 2048
78af6ae0-a3ae-403f-812d-48f2b9bde56	server	Imported	Issued	Yes	Ineligible	RSA 2048

Set AWS VPN

Go to the VPC service to create the VPN

1. Go to the section Client VPN endpoints and click on create VPN endpoints

▼ Virtual private network (VPN)

Customer gateways

Virtual private gateways

Site-to-Site VPN connections

Client VPN endpoints



The screenshot shows the AWS Client VPN endpoints console. At the top, there's a header 'Client VPN endpoints (1/1) info'. Below it, a section titled 'Filter client VPN endpoints' contains a search bar and buttons for 'Actions', 'Download client configuration', and 'Create client VPN endpoint'. A table lists the endpoints with columns: Name, Client VPN endpoint ID, State, and Client CIDR. One endpoint is visible.

Name	Client VPN endpoint ID	State	Client CIDR

2. Set the CIDR value according to what you need
Take a look at this web page: <https://www.ipaddressguide.com/cidr>

Create client VPN endpoint [Info](#)

Create a client VPN endpoint to enable access to networks over a TLS VPN session.

Details

Name tag - optional
Creates a tag with the key set to Name and the value set to the specified string.

Name must be 255 characters or less in length.

Description - optional
A brief description of the client VPN endpoint.

Client IPv4 CIDR [Info](#)
The IP address range, in CIDR notation, from which client IP addresses are allocated.

CIDR block cannot be larger than /12 or smaller than /22.

3. Authentication information
Choose mutual authentication and pick the server and client certificates imported to ACM

Authentication information [Info](#)

Server certificate ARN
The server certificate must be provisioned with or imported into AWS Certificate Manager (ACM).

Authentication options
Choose one or a combination of authentication methods to use.

☒ Use mutual authentication

☐ Use user-based authentication

Client certificate ARN [Info](#)

4. Not connection logging and Not client handler to avoid an extra cost
5. Other parameters
 - f. Put the google DNS 8.8.8.8
 - g. Check UDP option
 - h. Choose the VPC id where the Kafka cluster is running.
 - i. Create a new Security group through the EC2 service
 - i. Select the target network association (here, we choose the networks from where you will allow the connection)
 - ii. validate if we want to enable anywhere connections or whatever you want :)

Other parameters - optional

DNS server 1 IP address

The IP address of the DNS server to use. There are no default DNS servers.

DNS server 2 IP address

The IP address of the DNS server to use. There are no default DNS servers.

Transport protocol [Info](#)

Transport protocol used by the TLS sessions.

☒ UDP☐ TCP☒ Enable split-tunnel [Info](#)

VPC ID

Security group IDs

Security groups to be applied to the endpoint.

sg-01b2b48605be8081d (mko-elquelolea-public-all) X
delete me

VPN port

AWS client VPN supports ports 443 and 1194 for both TCP and UDP.

☐ Enable self-service portal [Info](#)

6. Add some tags and click on create VPN

Set authorization rule

1. Go to client VPN endpoints again and choose the new VPN
2. Click on the authorizations rule in the bottom section

VPC > Client VPN endpoints > cvpn-endpoint-0678cf5f713a2abcd

cvpn-endpoint-0678cf5f713a2abcd / mko-elquelolea-test [Info](#) [Download client configuration](#) [Actions](#)

Details

Client VPN endpoint ID cvpn-endpoint-0678cf5f713a2abcd	Server certificate ARN arn:aws:acm-us-east-1:965108296213:certificate/78af6ae0-a3ae-409f-812d-4872b9b0e456	Connection log false	Transport protocol udp
Description delete me my friend	Creation time February 16, 2023, 16:31 (UTC-08:00)	Cloudwatch log group -	Split-tunnel Enabled
State Available	VPN port 443	Cloudwatch log stream -	VPC ID vpc-06ca24e83f5b76113
Authentication type certificate-authentication	Security Group IDs sg-01b2b48600be8081d	Client CIDR 40.0.0.0/22	Self-service portal URL -
Directory ID -	Client connect handler ARN -	SAML provider ARN -	Client connect handler state Applied
DNS name *cvpn-endpoint-0678cf5f713a2abcd.prod.clients.vpn.us-east-1.amazonaws.com	Session timeout hours 10	Self-service SAML provider ARN -	Client login banner text -
DNS servers 8.8.8.8		Client certificate ARN arn:aws:acm-us-east-1:965108296213:certificate/b71e93f4-c7a2-42a1-a281-78c92860cbee	

Target network associations | Security groups | **Authorization rules** | Route table | Connections | Tags

Authorization rules (1) [Info](#)

Filter authorization rules

Endpoint ID	State	Description	Group ID	Access all	Destination CIDR
cvpn-endpoint-0678cf5f713a2abcd	Active	-	-	True	10.0.0.0/16

3. Add the rule to the destination

Add authorization rule [Info](#)

Add authorization rules to grant clients access to the networks.

Details

Client VPN endpoint ID
cvpn-endpoint-0678cf5f713a2abcd

Destination network to enable access
The IP address, in CIDR notation, of the destination network.

10.0.0.0/16

Grant access to:

☒ Allow access to all users

☐ Allow access to users in a specific access group

Description - optional
A brief description of the authorization rule.

description

[Cancel](#) [Add authorization rule](#)

Add target network associations

1. It is the step where we need to add all the subnets associated with the Kafka cluster (you can get them by looking at the MKS service)

Details

Client VPN endpoint ID: cvpn-endpoint-0678cf5f713a2abcd
 Description: delete me my friend
 State: Available
 Authentication type: certificate-authentication
 Directory ID: -
 DNS name: *cvpn-endpoint-0678cf5f713a2abcd.prod.clients.vpn.amazonaws.com
 DNS servers: 8.8.8.8

Server certificate ARN: arn:aws:acm:us-east-1:965108296213:certificate/78af5ae0-a5ae-409f-812d-48f2b9b0be56
 Creation time: February 16, 2023, 16:31 UTC-08:00
 VPN port: 443
 Security Group IDs: sg-01b2b48605be8081d
 Client connect handler ARN: -
 Session timeout hours: 10

Connection log: false
 Cloudwatch log group: -
 Cloudwatch log stream: -
 Client CIDR: 40.0.0.0/22
 SAML provider ARN: -
 Self-service SAML provider ARN: -
 Client certificate ARN: arn:aws:acm:us-east-1:965108296213:certificate/b71e99d4-c7a2-42a1-a281-78c92860bbe

Transport protocol: udp
 Split tunnel: Enabled
 VPC ID: vpc-06ca24e83fb876115
 Self-service portal URL: -
 Client connect handler state: Applied
 Client login banner text: -

Target network associations (3)

Association ID	State	Network ID	Security groups	Endpoint ID	Description
cvpn-assoc-06107b0ee532344c	Associated	subnet-00264cc3f57bcefd	sg-01b2b48605be8081d	cvpn-endpoint-0678cf5f713a...	-
cvpn-assoc-09bde06d2e3f1b70	Associated	subnet-001766c6054f6cc90	sg-01b2b48605be8081d	cvpn-endpoint-0678cf5f713a...	-
cvpn-assoc-0d1c3ef11f1b9b88a	Associated	subnet-07a741852f39a9904	sg-01b2b48605be8081d	cvpn-endpoint-0678cf5f713a...	-

Configure AWS VPN certificate (This step is required only once when the VPN is new)

i This step is just needed if we don't have a certificate configured to use the VPN as a client. You could skip this step and use the certificate in the next step.

1. Download the client configuration file from the VPC service -> client VPN endpoint

Button: Download client configuration

Client VPN endpoints (1/1)

Name	Client VPN endpoint ID	State	Client CIDR
mko-elquelolea-test	cvpn-endpoint-0678cf5f713a2abcd	Available	40.0.0.0/22

2. Go to the directory where you downloaded the certificate and open the file.

e.g. nano downloaded-client-config.ovpn

3. Add the client cat and key created in the first step of the process

e.g.

a. Go to the directory where you have the certificates custom_easyrsa_folder

b. Open the certificate crt client1.domain.tld.crt

c. Copy the certificate into the .ovpn file between the tags <cert> </cert>

```
<cert>
-----BEGIN CERTIFICATE-----
HERE
-----END CERTIFICATE-----
</cert>
```

d. Open the key file client1.domain.tld.key

e. Copy client1.domain.tld.key -> the key into the .ovpn file between tags <key> </key>

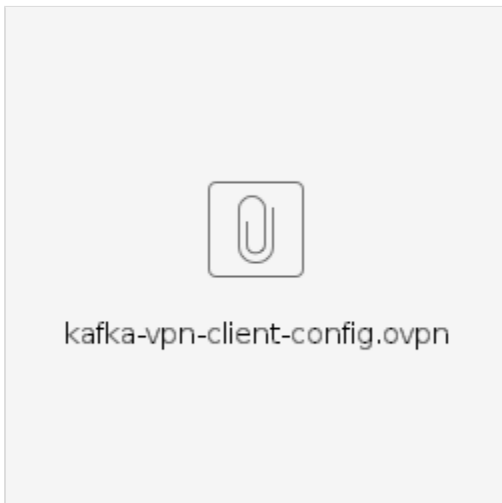
```
<key>
-----BEGIN PRIVATE KEY-----
here
-----END PRIVATE KEY-----
</key>
```

4. Now open `.ovpn` file again and modify the property `remote`, adding a subfix value; in this case, I added the value `article`.

```
client
dev tun
proto udp
remote article.cvpn-endpoint-0678cf5f713a2abcd.prod.clientvpn.us-east-1.amazonaws.com 443
remote-random-hostname
resolv-retry infinite
nobind
remote-cert-tls server
cipher AES-256-GCM
verb 3
<ca>
```

Import the VPN certificate to your Tunnelblick

As we already have a VPN, you can use this certificate to test it.



Once you import the certificate and connect to the VPN, you can execute the below command from your terminal to ensure it works properly.

```
netstat -r
```




```
⇒ k8s git:(feature/BXC-723-add-tiltfile-kafka-local-env) ✗ netstat -r
Routing tables

Internet:
Destination      Gateway          Flags            Netif  Expire
default          192.168.0.1     UGScg           en0
10/16            40.0.0.129      UGSc            utun11
40.0.0.128/27    40.0.0.130      UGSc            utun11
40.0.0.130       40.0.0.130      UH              utun11
127              localhost       UCS             lo0
localhost        localhost       UH              lo0
```

```
telnet z-3.devkafka.rwtj5d.c16.kafka.us-east-1.amazonaws.com 2182
Trying 54.93.131.246...
Connected to z-3.devkafka.rwtj5d.c16.kafka.us-east-1.amazonaws.com.
Escape character is '^['.
```

Next steps

- Investigate how we can authenticate and authorize users in a better way
- Work on this spike  [ECCO-1131](#) - Getting issue details... STATUS
 - <https://support.jumpcloud.com/support/s/article/Single-Sign-On-SSO-with-AWS-Client-VPN>
 - [AWS approach](#)
 - https://www.youtube.com/watch?v=l4TpB_MV8tI&list=PL8K2AhiVhR195HTta5rYctdS_PH61PuXM&index=1

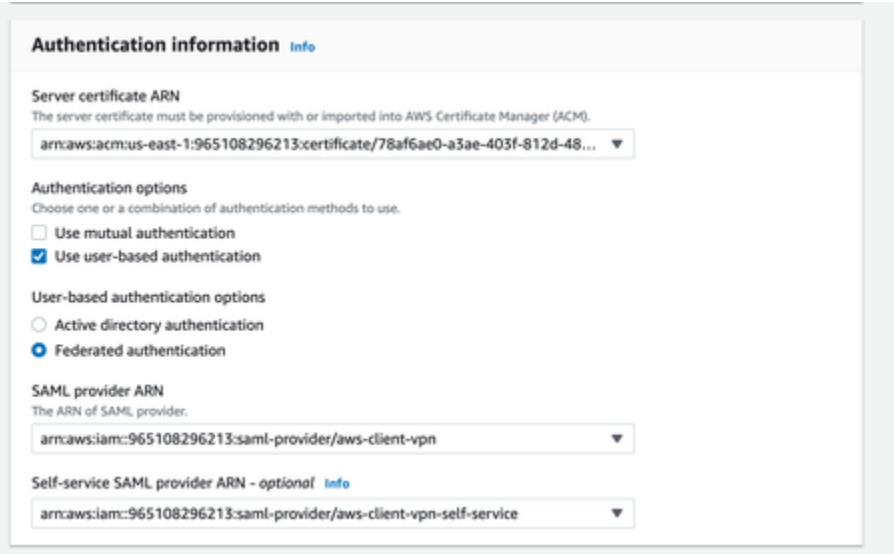
Authenticate AWS Client VPN users with federated authentication.

We implemented the approach recommended on this page, [Authenticate AWS Client VPN users with AWS IAM Identity Center](#), since it avoids having a dependency on the IT team for the configuration and at the same time, we can take advantage of the federated authentication configured on the root account (**Engineering Production**).

We created two SAML applications:

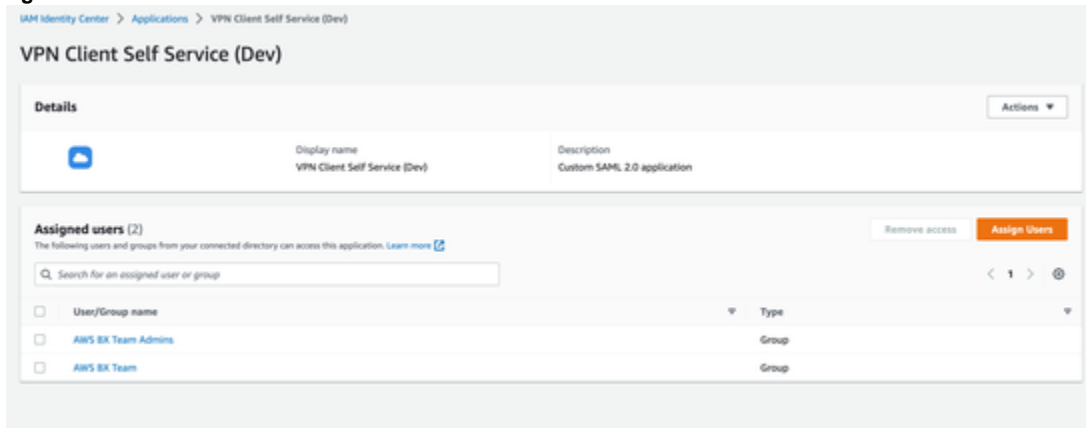
- [AWS VPN \(dev\)](#)
 - This allows defining the group of users enabled to use the VPN
 - The metadata generated by this application is used to set up an identity provider in the dev account (**Engineering Dev**)
 - [Identity provider](#)
- [VPN Client Self Service \(Dev\)](#)
 - This allows defining the group of users enabled to download the VPN configuration (.openvpn file)
 - The metadata generated by this application is used to set up an identity provider in the dev account (**Engineering Dev**)
 - [Identity provider](#)

The VPN uses the identity providers created previously, and we set them when creating the VPN. e.g.

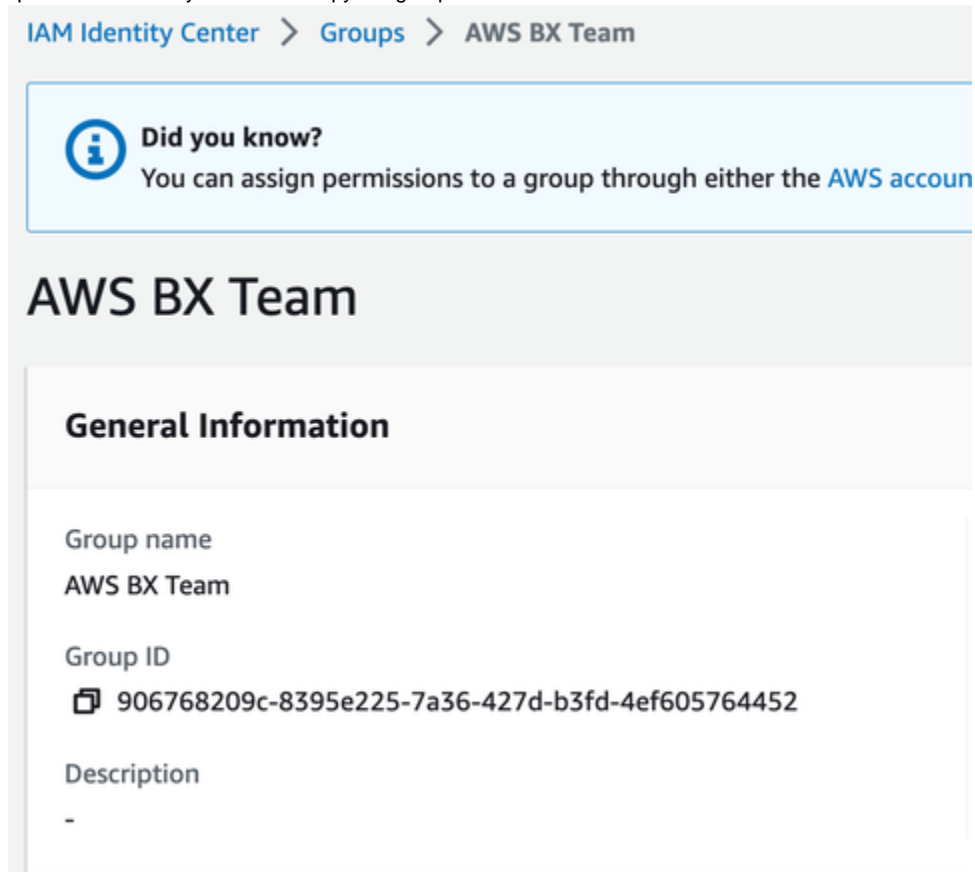


How to add or remove user groups to the VPN

1. Go to the root account (**Engineering Production**), where the Identity center is enabled.
2. Select the applications **AWS VPN (dev)** and **VPN Client Self Service (Dev)**
 - a. We have to make this change for both applications.
3. Click on **Assign Users**



4. Select the new group that you want to add
5. Go to the **groups** option in the Identity center and copy the group id



6. Open the VPN created in the Dev account
7. Select the VPN and add the authorization rule for the new group

Client VPN endpoints (1/2) [Info](#)

Filter client VPN endpoints

Actions

Download client configuration

Create client VPN endpoint

Name	Client VPN endpoint ID	State	Client CIDR
-	cvpn-endpoint-09b1366ea15a5ff7b	Available	40.0.0.0/22
mko-elqutalea-test	cvpn-endpoint-057b7f715a2abcf	Available	40.0.0.0/22

cvpn-endpoint-09b1366ea15a5ff7b

Details

Target network associations

Security groups

Authorization rules

Route table

Connections

Tags

Authorization rules (2) [Info](#)

Filter authorization rules

Remove authorization rule

Add authorization rule

Endpoint ID	State	Description	Group ID	Access all	Destination CIDR
cvpn-endpoint-09b1366ea15a5ff7b	Active	AWS BK Team	906768209c-839fa225...	False	10.0.0.0/16
cvpn-endpoint-09b1366ea15a5ff7b	Active	AWS BK Team Admins	906768209c-9c9377f2...	False	10.0.0.0/16

VPC > Client VPN endpoints > cvpn-endpoint-09b1366ea15a5ff7b > Add authorization rule

Add authorization rule [Info](#)

Add authorization rules to grant clients access to the networks.

Details

Client VPN endpoint ID

cvpn-endpoint-09b1366ea15a5ff7b

Destination network to enable access

The IP address, in CIDR notation, of the destination network.

10.0.0.0/16

Grant access to:

☐ Allow access to all users

☒ Allow access to users in a specific access group

Access group ID

Unique group identifier. It can be active directory SID or group name in IDP.

Copy the group id here

Description - optional

A brief description of the authorization rule.

description

Cancel

Add authorization rule