

AMF Plus Cloud on Amazon Web Services (AWS) Installation Guide

Installation Guide

Introduction

AMF Plus Cloud is a scalable cloud-based network management platform. It supports Allied Telesis switching, firewall, and wireless products, as well as a wide range of third-party devices.

This installation guide enables you to install and configure your AMF Plus Cloud in an Amazon Web Services (AWS) cloud environment.

Note: This document contains a lot of AWS-specific terminology. For more detailed information about AWS terms and concepts, please refer to the AWS documentation. Also, the screenshots shown were current at the time of creation, but are subject to change.

Contents

Introduction	1
Procedure overview	3
Create an Amazon Machine Image	4
Prerequisites	4
Create an API key	4
Install required packages	9
Preparing the VHD image file and the Python script file	9
Upload VHD image file and create AMI	10
Create an instance	12
Prerequisites	12
Create VPC	13
Create instance	16
Create and configure an internet gateway	28
Create a route table	30
SSH connection settings	35
SSH key pair	35
Accessing the instance via SSH using "PuTTY"	36
SSH connection to the instance using SSH client of Ubuntu (Linux)	39
Connecting to your local network	40
How to use the VPN function of AMF Plus Cloud	40
How to use AWS (VPC) VPN function	44
Connection with tenant networks (multi-tenant mode)	58
How to use AMF Cloud's VPN function	58
Firmware update	64
Prerequisite	64
About ISO files and VHD files	64
Update procedure	64
Tips and troubleshooting	65
Lost network connection	65
When the SSH server function is disabled	65
Creating an instance snapshot	66

Procedure overview

The general procedure for setting up this product on AWS is as follows:

1. [“Create an Amazon Machine Image”](#)

Upload the VHD image file of this product to Amazon EC2 to create an Amazon Machine Image (AMI).

2. [“Create an instance”](#)

Create an instance (virtual machine) of this product from the AMI created in Step 1.

3. [“SSH connection settings”](#)

Access the instance using an SSH client (for example, PuTTY).

4. [“Connecting to your local network”](#)

Create an IPsec VPN with the local network to enable secure communication between AWS and devices on the local network.

Create an Amazon Machine Image

To create an instance (virtual machine) of this product on AWS, you need to create an Amazon Machine Image (AMI), which is a virtual machine template. This section explains how to upload the VHD image file to AWS and create an AMI.

Note: This process is only required the first time you install. After initial setup, you can use the **software-upgrade** command to update the firmware (see the [“Firmware update”](#) section).

Prerequisites

To create an AMI, you need:

- A computer that can connect to the Internet running Linux (Ubuntu or Debian).

Note: Windows is not supported.

- An AWS API key (AKID and SAK, ID and password to use the API) with full access permissions for Amazon EC2 and Amazon S3.

Note: Creating this key is described in the [“Create an API key”](#) section.

- The .vhd disk image file.
- The .py Python upload script file.

Note: These files are available from the Software Download Centre.

- The Amazon EC2 API tool (a tool for performing various operations on EC2 from the command line). Please refer to the [AWS CLI installation documentation](#) for details.

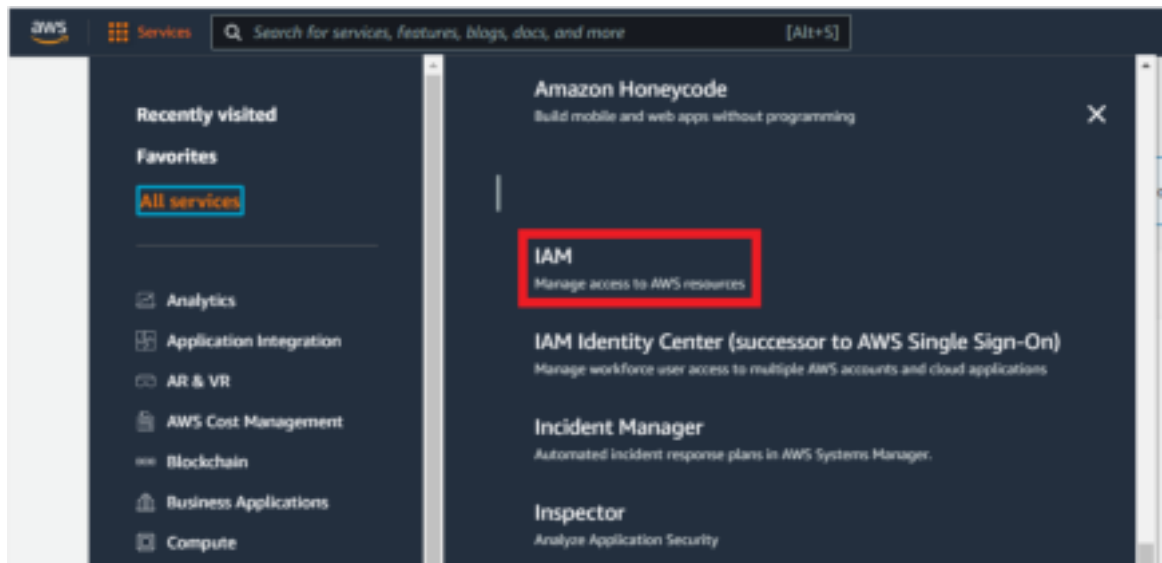
Create an API key

To create an AMI, you first need an AWS API key.

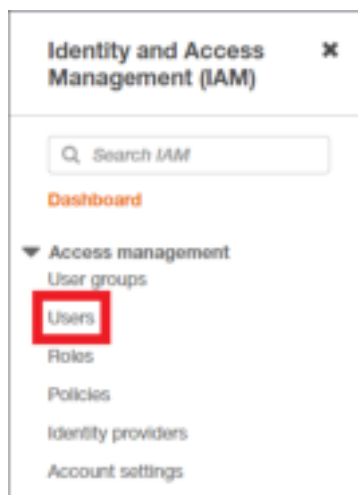
To create an API key, you must have execute permissions for the user access and encryption key management service “Identity and Access Management (IAM)”. Please refer to the [AWS Identity and Access Management documentation](#) for details.

If you are configuring Roles in the navigation pane, you must create a role named **vmimport**, specify in the trust relationship policy document that VM Import assumes this role, and attach an IAM policy to the role. Please refer to [the AWS VM Import/Export documentation](#) for details.

1. From the home screen of the AWS Management Console, select **Services > All Services > IAM**.



2. On the IAM dashboard screen, click **Users** under **Access Management** on the left menu.



3. Click on your IAM user-name.



4. Switch to the **Security credentials** tab.



5. On the **Security credentials** tab, click **Create Access Key**.



6. On the **Access key best practices & alternatives** dialog, select a Use case, and click **Next**.

Access key best practices & alternatives [info](#)

Avoid using long-term credentials like access keys to improve your security. Consider the following use cases and alternatives.

Use case

☐ **Command Line Interface (CLI)**
You plan to use this access key to enable the AWS CLI to access your AWS account.

☐ **Local code**
You plan to use this access key to enable application code in a local development environment to access your AWS account.

☐ **Application running on an AWS compute service**
You plan to use this access key to enable application code running on an AWS compute service (like Amazon EC2, Amazon ECS, or AWS Lambda) to access your AWS account.

☐ **Third-party service**
You plan to use this access key to enable access for a third-party application or service that monitors or manages your AWS resources.

☐ **Application running outside AWS**
You plan to use this access key to authenticate workloads running in your data center or other infrastructure outside of AWS that needs to access your AWS resources.

☐ **Other**
Your use case is not listed here.

[Cancel](#) [Next](#)

7. On the **Set description tag** dialog, you can optionally set a tag to describe the purpose of the key. Once you are done, click **Create Access Key**.

Set description tag - optional [info](#)

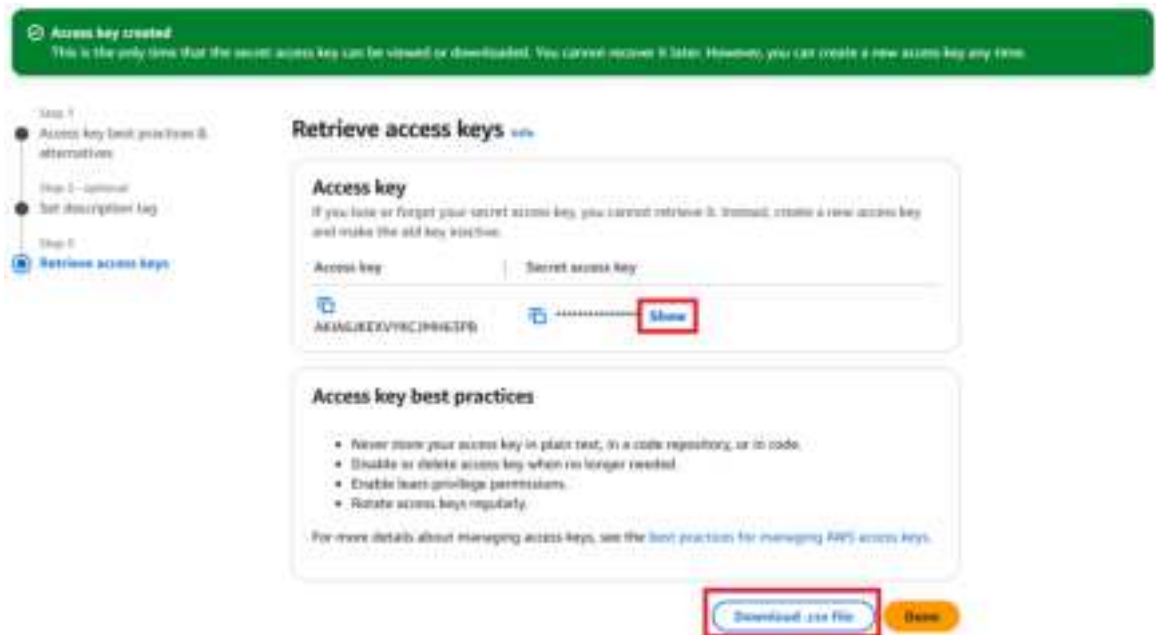
The description for this access key will be attached to this user as a tag and shown alongside the access key.

Description tag value
Describe the purpose of this access key and where it will be used. A good description will help you relate this access key confidently later.

Maximum 256 characters. Allowed characters are letters, numbers, spaces, representable in UTF-8, and `:/=+~@`.

[Cancel](#) [Previous](#) [Create access key](#)

8. The **Create access key** dialog will be displayed. Click **Download .csv file** to save the access key ID and secret access key information.



Alternatively, click **Show** next to the secret access key to display the key. You can then copy the access key ID and secret access key information, and save them locally.



Note: Keep your access key in a safe place. A generated key can only be downloaded once. Do not send the key by e-mail. Do not hand over the key information, even if you receive an inquiry from AWS or Amazon.com. An authorized Amazon representative will never ask you for a key.

9. You have now created your AWS API key.

Install required packages

Install the required packages on your Linux (Ubuntu or Debian) computer. The following packages are required to upload the VHD image file of this product to AWS and create an AMI:

- Python version 3.7 or later
- Ec2-api-tools
- boto3
- Python-pip

For example, to install them on Ubuntu, enter the following commands:

```
ubuntu@ubuntu-pc:~/tmp$ sudo apt-get install ec2-api-tools
ubuntu@ubuntu-pc:~/tmp$ sudo apt-get install python3
ubuntu@ubuntu-pc:~/tmp$ sudo apt-get install python3-pip
ubuntu@ubuntu-pc:~/tmp$ sudo pip install "boto3<=1.35.80"
```

Preparing the VHD image file and the Python script file

Create a temporary folder on your computer. Copy the VHD image file “vaa-X.X.X-X.X.vhd” (where X.X.X-X.X is the version you want to install) and the Python script “upload_vhd.py” to this location.

In the confirmation screen example below, it is assumed that these files are placed in the user's tmp folder directly under the home directory of the user.

```
ubuntu@ubuntu-pc:~/tmp$ ls
vaa-5.5.3-2.1.vhd upload_vhd.py
```

Upload VHD image file and create AMI

Use the Python script “upload_vhd.py” to upload the VHD image file to AWS and create an AMI.

The command line format and arguments for executing the script are as follows:

```
format
python upload_vhd.py IMAGEFILE AMINAME --region NAME --bucket NAME --akid KEY
--sak KEY

argument
  IMAGEFILE : VHD image file of this product to be imported (Example: vaa-5.5.3-2.1.vhd)
  AMINAME : AMI name (Example: vaa-5.5.3)
  --region NAME : AWS region to use (e.g. ap-northeast-1)
                  *For a list of region names, please refer to Amazon's user guide.
  --bucket NAME : AWS S3 bucket to temporarily upload the VHD file to (e.g. vaa.upload)
                  *If it does not exist, the bucket will be created automatically.
  --akid KEY : API key access key ID (for access to EC2 and S3) (e.g. AKIDABCDF)
  --sak KEY : API key secret key (for access to EC2 and S3) (e.g. SAKABCDF)
```

The VHD image file is temporarily uploaded to your AWS S3 bucket.

Note: Please refer to Amazon's user guide for charges incurred by using S3.
Bucket names must be unique across S3 (you cannot use a bucket name used by another S3 user). Refer to Amazon's user guide for bucket naming conventions.

An execution example is shown below:

```
ubuntu@ubuntu-pc:~/tmp$ python upload_vhd.py vaa-5.5.3-2.1.vhd vaa-5.5.3 --
region ap-northeast-1 --bucket vaa.upload --akid AKIDABCDF --sak SAKABCDF
upload_image: Creating Bucket
upload_image: Uploading disk image
upload_image: 10% (12MB/120MB)
upload_image: 20% (24MB/120MB)
upload_image: 30% (36MB/120MB)
upload_image: 40% (48MB/120MB)
upload_image: 50% (60MB/120MB)
upload_image: 60% (72MB/120MB)
upload_image: 70% (84MB/120MB)
upload_image: 80% (96MB/120MB)
upload_image: 90% (108MB/120MB)
import_snapshot: Converting disk image to EBS snapshot
import_snapshot: ImportTaskId=import-snap-0153ad4f76fb9e4bb
import_snapshot: 2%
import_snapshot: 43%
import_snapshot: 100%
import_snapshot: Snapshot created snap-0a20e8cb894a2f65d
import_snapshot: Deleting disk image from S3
register_image: Creating AMIs
register_image: AMI created ami-038777b2b30e26eb6
```

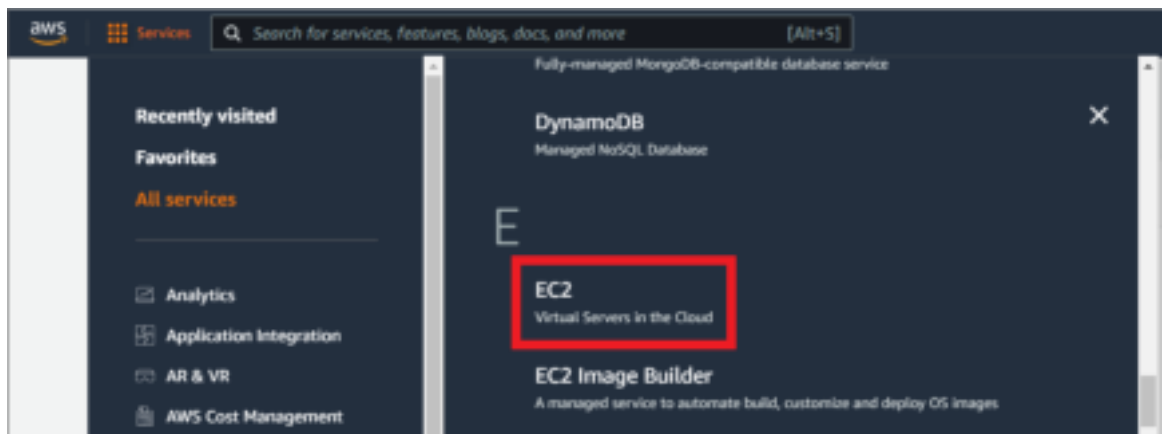
Note: The content displayed when the script is executed is an example. The displayed contents may differ depending on the settings in AWS.

When this product is successfully uploaded to AWS, the following message will be displayed.
 XXXXXXXX is automatically generated during the process.

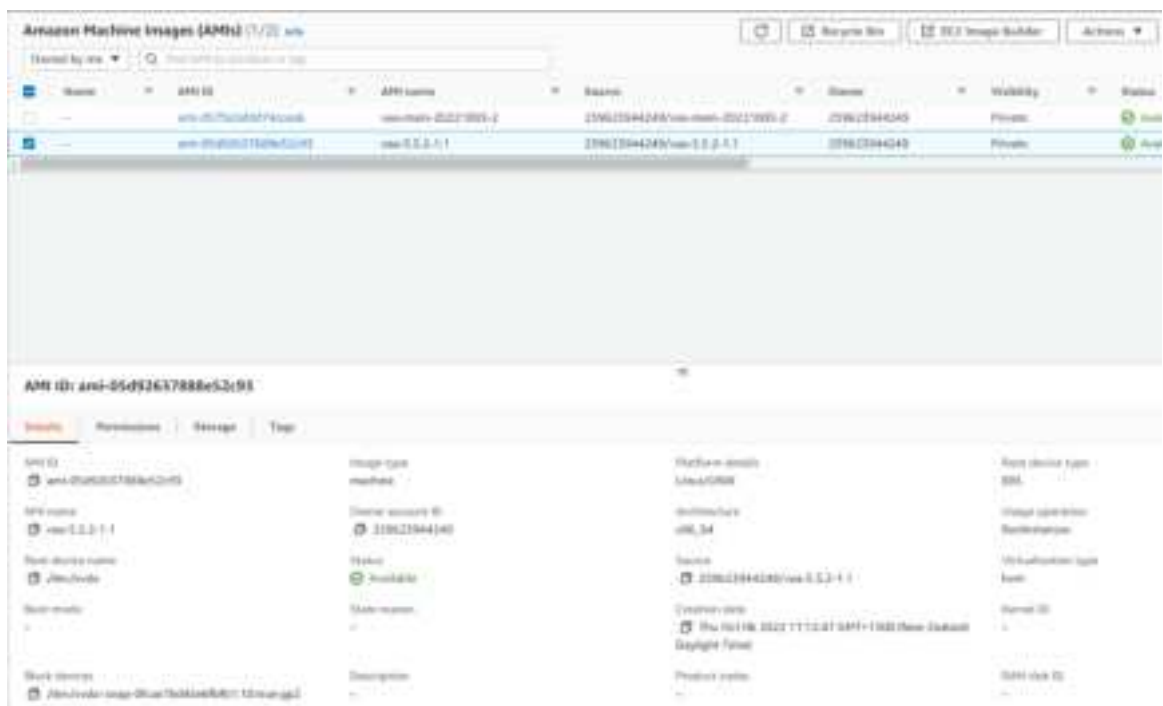
```
register_image: AMI created ami-XXXXXXX
```

You can also check the AMI in the EC2 dashboard.

1. From the home screen of the AWS Management Console, select **Services > All Services > EC2**.



2. On the EC2 dashboard screen, click **AMI** under **Image** on the left menu.



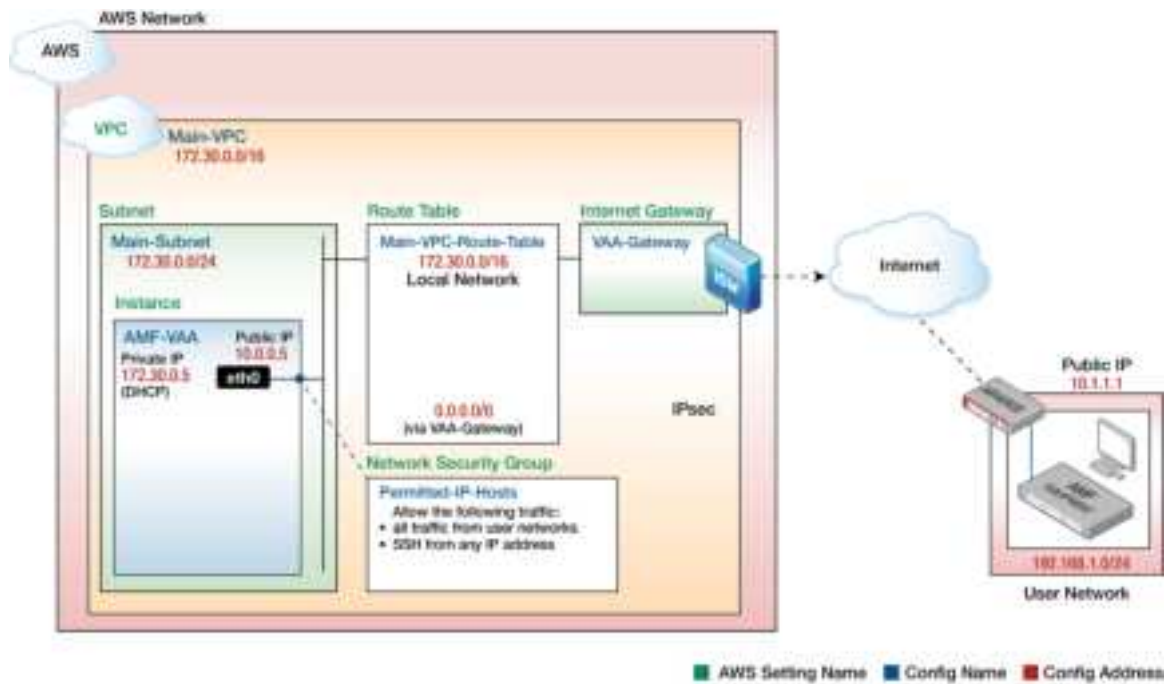
Create an instance

The next step in the process is to create a instance (virtual machine).

Prerequisites

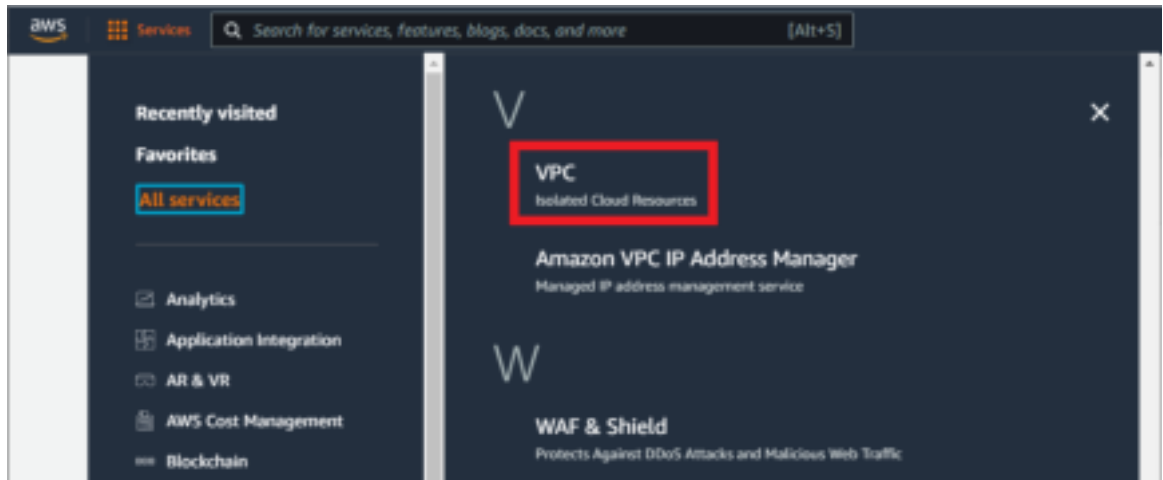
To create an instance, you need an AMI as a template. This section assumes that you have already completed the “[Create an Amazon Machine Image](#)” section.

Network configuration, SSH keys, access control, etc, also need to be planned in advance. This document assumes these have already been completed.

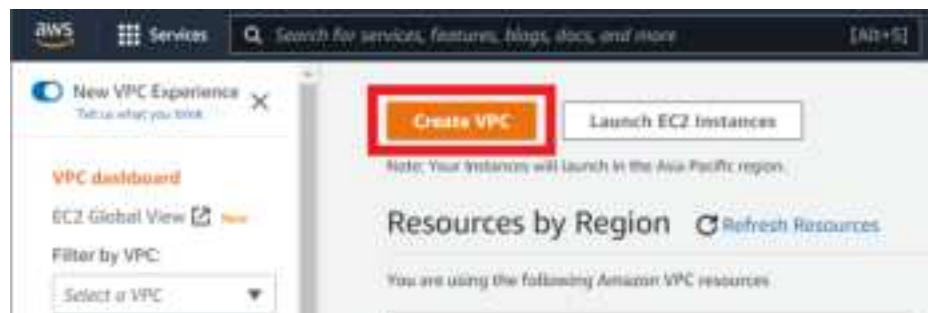


Create VPC

1. From the home screen of the AWS Management Console, select **Services** > **All Services** > **VPC**.



2. Click **Create VPC** on the VPC dashboard screen.



3. On the **Configure VPC** screen, configure the following settings and click **Create VPC**.

VPC > Your VPCs > Create VPC

Create VPC [info](#)

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

VPC settings

Resources to create [info](#)
Create only the VPC resource or the VPC and other networking resources.

☒ VPC only ☐ VPC and more

Name tag - optional [info](#)
Creates a tag with a key of `Name` and a value that you specify.

Main-VPC

IPv4 CIDR block [info](#)

☒ IPv4 CIDR manual input
☐ IPAM-allocated IPv4 CIDR block

IPv4 CIDR

172.30.0.0/16

IPv6 CIDR block [info](#)

☒ No IPv6 CIDR block
☐ IPAM-allocated IPv6 CIDR block
☐ Amazon-provided IPv6 CIDR block
☐ IPv6 CIDR owned by me

Tenancy [info](#)

Default

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

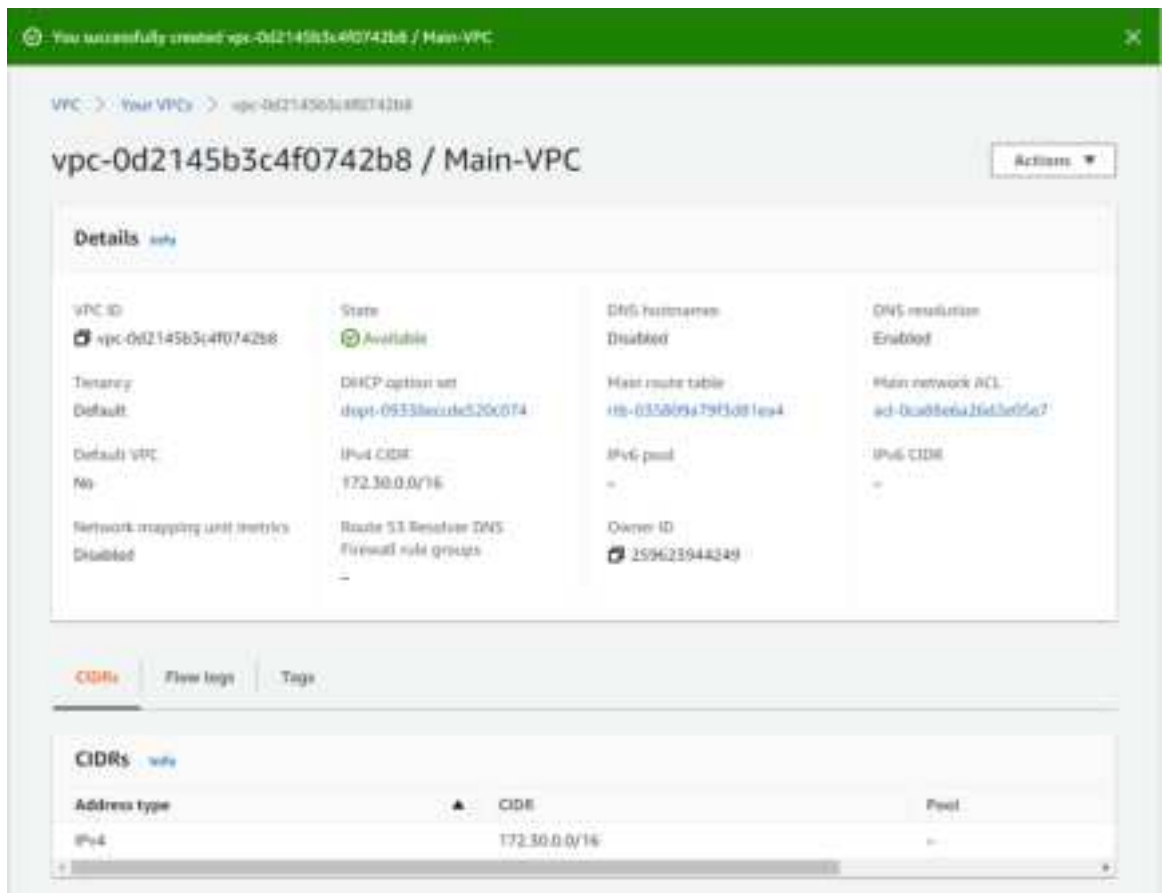
Key	Value - optional	
Name	Main-VPC	Remove

[Add new tag](#)

You can add 40 more tags.

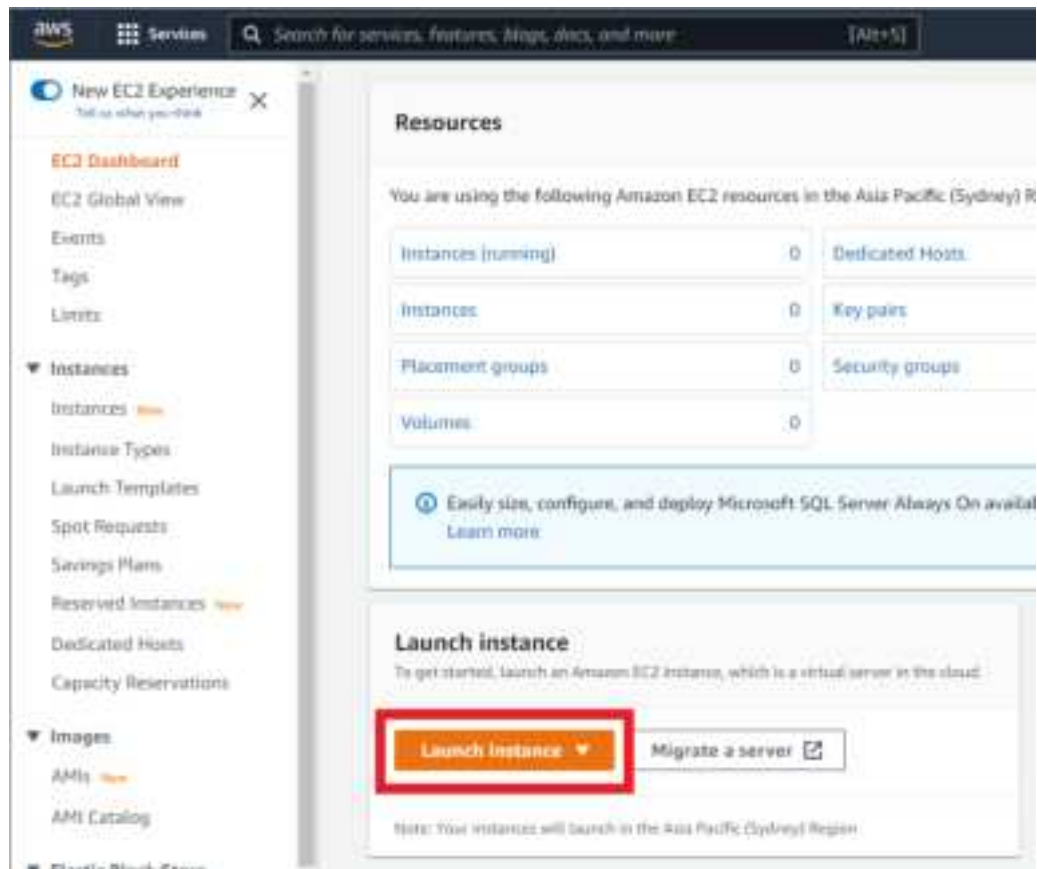
Cancel **Create VPC**

4. If the VPC is successfully created, you will see a screen like the one below.



Create instance

1. From the **Services** menu of the AWS Management Console, select **All Services** > **EC2** to open the EC2 dashboard screen, then click **Launch Instance** > **Launch Instance**.



2. On the **Launch an instance** screen, configure the settings as follows:

- a. Name and tags

Name and tags [Info](#)

Name

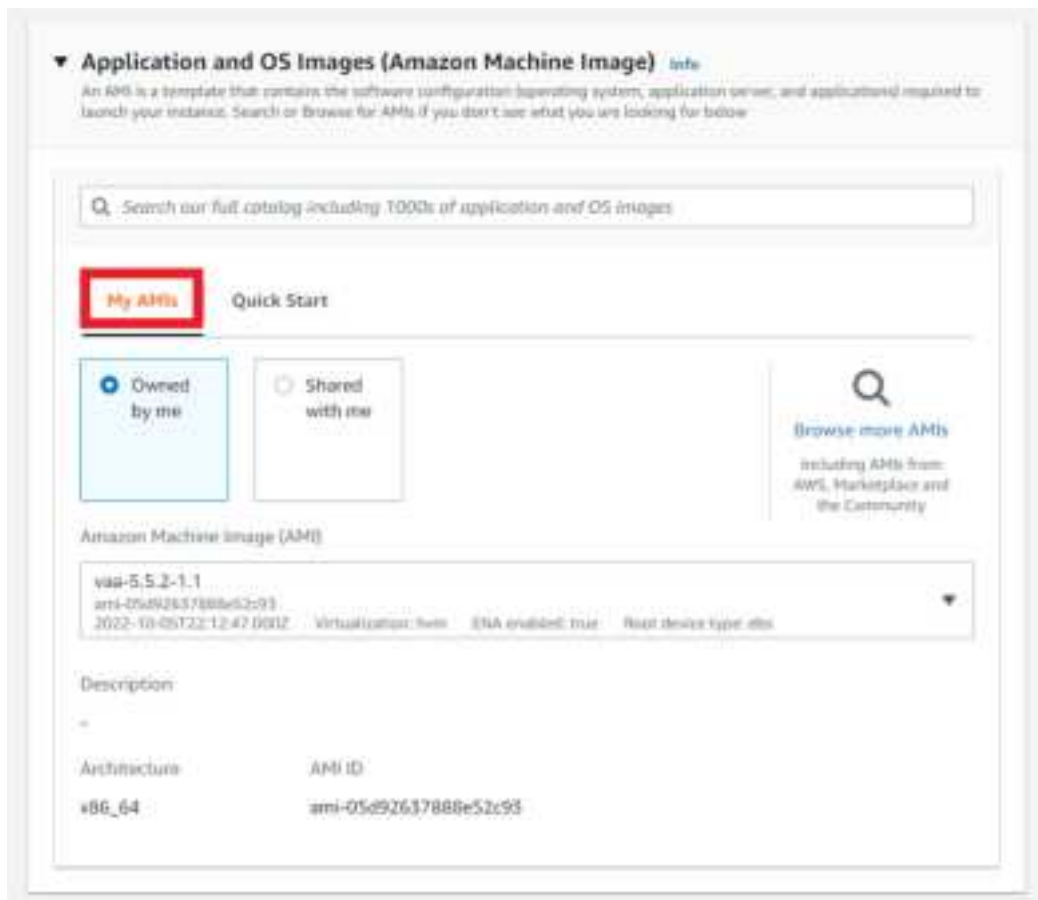
AMFCloud

[Add additional tags](#)

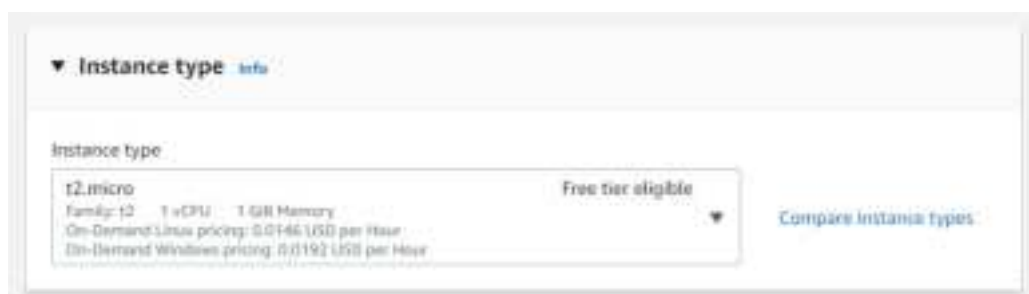
Enter a name for your instance.

b. Application and OS Images (Amazon Machine Image)

Click the **My AMIs** tab and the AMI you just created should be selected. If a different one is selected, select the AMI you just created from the drop-down list.

**c. Instance type**

The instance requirements differ depending on the usage environment. Refer to the datasheet and the AWS documentation and select the appropriate instance type.



d. Key Pair (Login)

Next, select or create an SSH key pair for your instance.

If you have already registered an SSH public key with AWS, you can select your key pair name from the drop-down.



▼ **Key pair (login)** [info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Select ▼

⌂ Create new key pair



▼ **Key pair (login)** [info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

MainKeyPair ▼

⌂ Create new key pair

If you have not yet registered a public key with AWS, follow the steps below to create a new key pair.

1. Click on the **Create new key pair** button.



▼ **Key pair (login)** [info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Select ▼

⌂ **Create new key pair**

2. In the **Key pair name** field, enter a name for your key pair. For the **Primary key file format**, select the format for the tools you intend to use to connect. Use the **.pem** format for OpenSSH, and the **.ppk** format for PuTTY.

Once you have configured your keys, click on the **Create key pair** button.

Create key pair

Key pairs allow you to connect to your instance securely.

Enter the name of the key pair below. When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** [Learn more](#)

Key pair name

az4000s-cloud-test

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

☒ RSA
RSA encrypted private and public key pair

☐ ED25519
ED25519 encrypted private and public key pair (Not supported for Windows instances)

Private key file format

☒ .pem
For use with OpenSSH

☐ .ppk
For use with PuTTY

Cancel **Create key pair**

3. Your new private key will be downloaded. For more information about using this key, refer to the “[SSH connection settings](#)” section.

Note: Keep the key file in a safe place. The key can only be downloaded this one time, when it is created.

e. Network settings

Click **Edit** and configure as follows:

▼

Network settings

info

VPC - required

info

vpc-0d2145b3c4f0742b8 (Main-VPC)

172.30.0.0/16

↻

Subnet

info

subnet-05ea7c4e@481c3464

Main-Subnet

↻

Create new subnet

+

vpc: vpc-0d2145b3c4f0742b8

Owner: 259623944348

Availability Zone: ap-southeast-2b

IP addresses available: 251

CIDR: 172.30.0.0/24

Auto-assign public IP

info

Enable

▼

Firewall (security groups)

info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group

Select existing security group

Security group name - required

launch-wizard-1

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _/[]!@#%^&*~

Description - required

info

launch-wizard-1 created 2022-10-06T00:48:22.230Z

Inbound security groups rules

▼

Security group rule 1 (TCP, 22, 0.0.0.0/0, Allow SSH)

Remove

Type

info

ssh

▼

Protocol

info

TCP

Port range

info

22

Source type

info

Anywhere

▼

Source

info

Q Add CIDR, prefix list or security

0.0.0.0/0

X

Description - optional

info

Allow SSH

▼

Security group rule 2 (All, All, 192.168.1.0/24, Allow From User Network)

Remove

Type

info

All traffic

▼

Protocol

info

All

Port range

info

All

Source type

info

Custom

▼

Source

info

Q Add CIDR, prefix list or security

192.168.1.0/24

X

Description - optional

info

Allow From User Network

Select the VPC that you created earlier.

Click **Create new subnet**.

Network settings [info](#)

VPC - required [info](#)

vpc-0d2145b3c4f0742b8 (Main-VPC)
172.31.0.0/16

Subnet - required

Select

[Create new subnet](#)

Auto-assign public IP [info](#)

Select

Firewall (security groups) [info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group ☐ Select existing security group

Security group name - required

launch-wizard-1

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _-!@#%&'()*+<()>?*

Description - required [info](#)

launch-wizard-1 created 2022-10-06T00:48:22.250Z

Inbound security groups rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0) Remove

Type info	Protocol info	Port range info
ssh	TCP	22

Source type info	Source info	Description - optional info
Anywhere	0.0.0.0/0	e.g. SSH for admin desktop

Warning: Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

[Add security group rule](#)

The **Create subnet** screen will be displayed. Set as follows and click **Create subnet**.

VPC

VPC ID
Create subnets in this VPC.

vpc-0d2145b5c4f0742b8 (Main-VPC)

Associated VPC CIDRs

IPv4 CIDRs

172.30.0.0/16

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.

Main-Subnet

The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

No preference

IPv4 CIDR block [Info](#)

172.30.0.0/24

Tags - optional

Key	Value - optional	
Name	Main-Subnet	Remove

[Add new tag](#)

You can add 49 more tags.

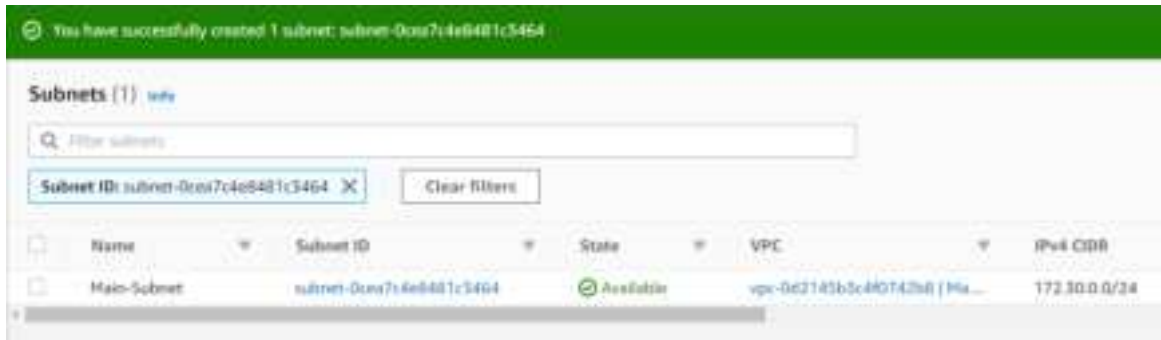
[Remove](#)

[Add new subnet](#)

[Cancel](#) [Create subnet](#)

A screen similar to the following appears when the subnet is successfully created.

Note: You may need to refresh the subnet list after creation is complete to have your new subnet appear.



Select **Enable** for automatic public IP assignment



Select **Create Security Group**.

Firewall (security groups) [info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group

☐ Select existing security group

Security group name - required

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _ / [] { } ^ * & # % ' .

Description - required [info](#)

Inbound security group rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0)

Remove

Type info	Protocol info	Port range info
<input type="text" value="ssh"/>	TCP	22
Source type info	Source info	Description - optional info
<input type="text" value="Anywhere"/>	Q. Add CIDR, prefix list or security group <input type="text" value="0.0.0.0/0"/> X	e.g. SSH for admin desktop

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

X

Add security group rule

Click on **Add security group rule**. Configure two security group rules as below.

Inbound security groups rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0, Allow SSH) Remove

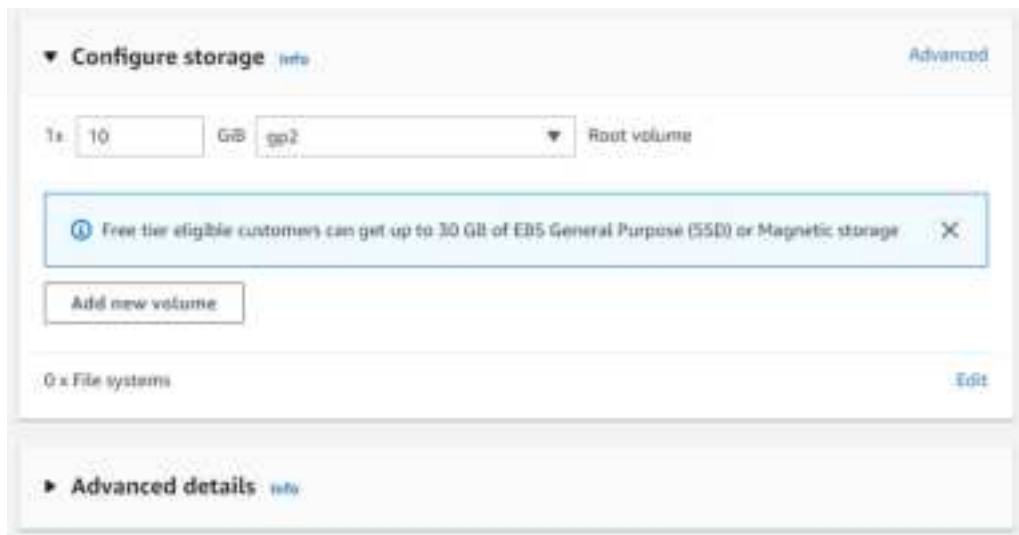
Type info	Protocol info	Port range info
ssh	TCP	22
Source type info	Source info	Description - optional info
Anywhere	<input type="text" value="0.0.0.0/0"/> <input type="button" value="Add CIDR, prefix list or security group"/>	Allow SSH

▼ Security group rule 2 (All, All, 192.168.1.0/24, Allow From User Network) Remove

Type info	Protocol info	Port range info
All traffic	All	All
Source type info	Source info	Description - optional info
Custom	<input type="text" value="192.168.1.0/24"/> <input type="button" value="Add CIDR, prefix list or security group"/>	Allow From User Network

f. Configure storage and Advanced details

The storage requirements differ depending on the usage environment. Refer to the datasheet and the AWS documentation and select the appropriate storage.



▼ **Configure storage** [info](#) [Advanced](#)

1x: 10 GiB gp2 Root volume

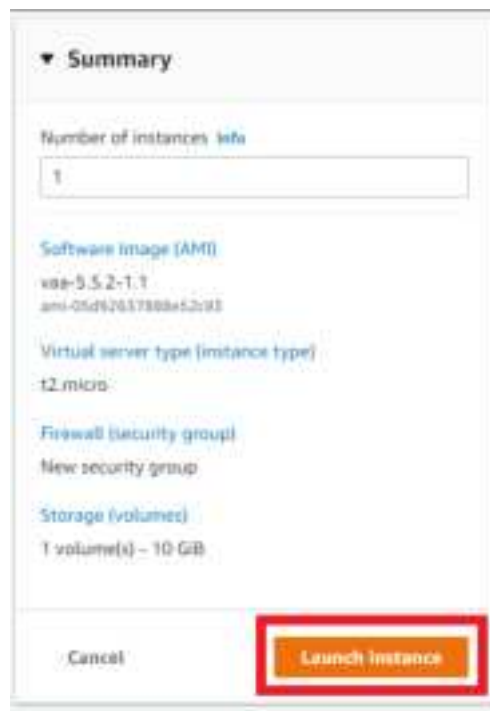
Free tier eligible customers can get up to 30 GiB of EBS General Purpose (SSD) or Magnetic storage

Add new volume

0 x File systems [Edit](#)

► **Advanced details** [info](#)

4. Click **Launch Instance**.



▼ **Summary**

Number of instances [info](#)

1

Software image (AMI)

x86_64.5.2-1.1
ami-05d62637888e52c93

Virtual server type (instance type)

t2.micro

Firewall (security group)

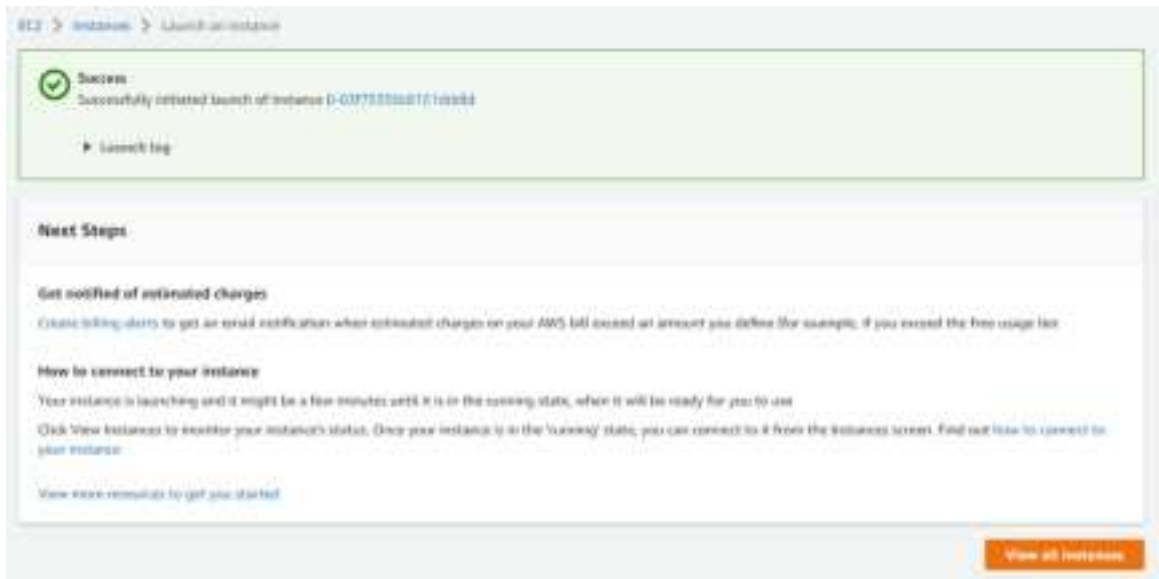
New security group

Storage (volumes)

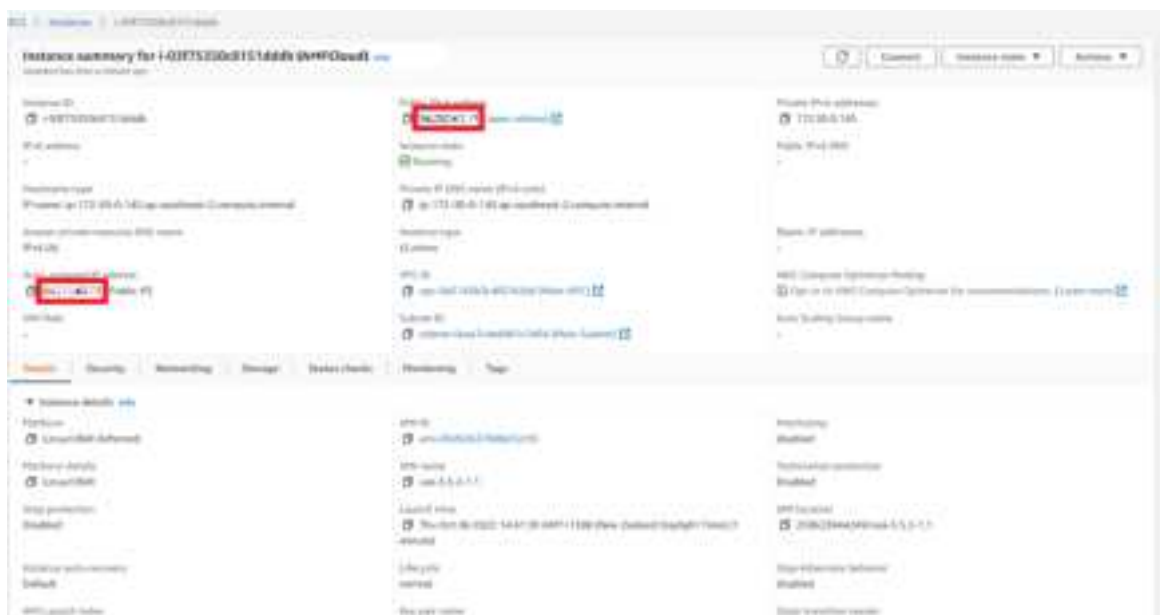
1 volume(s) - 10 GiB

Cancel **Launch Instance**

5. If the instance is successfully created, you will see a screen like the one below.

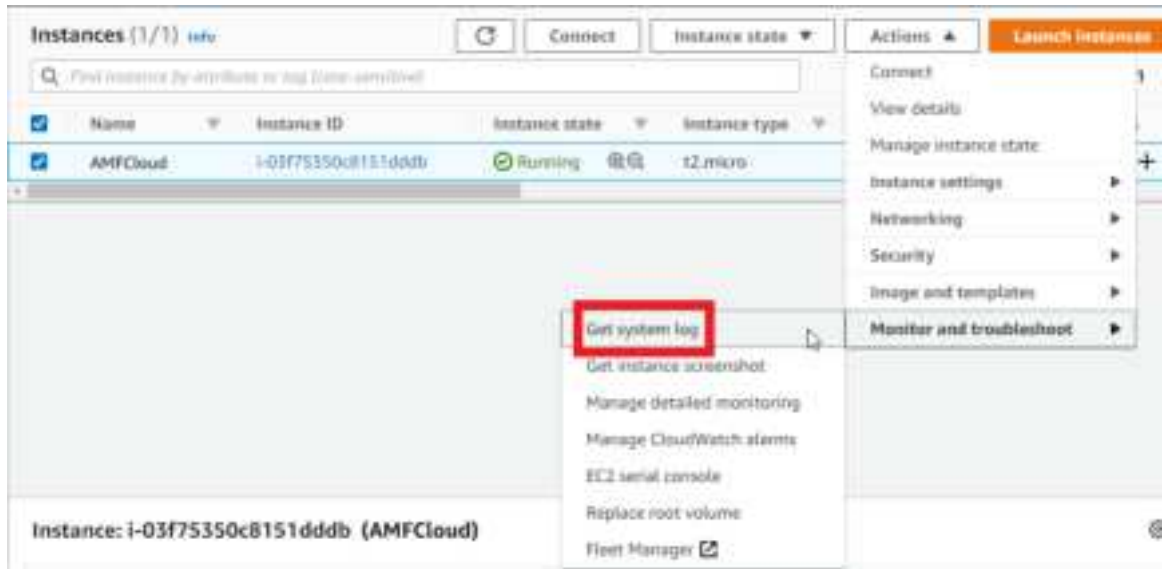


You can also check the public IP address of the instance used for SSH connection, as well as other settings, on the following screen.

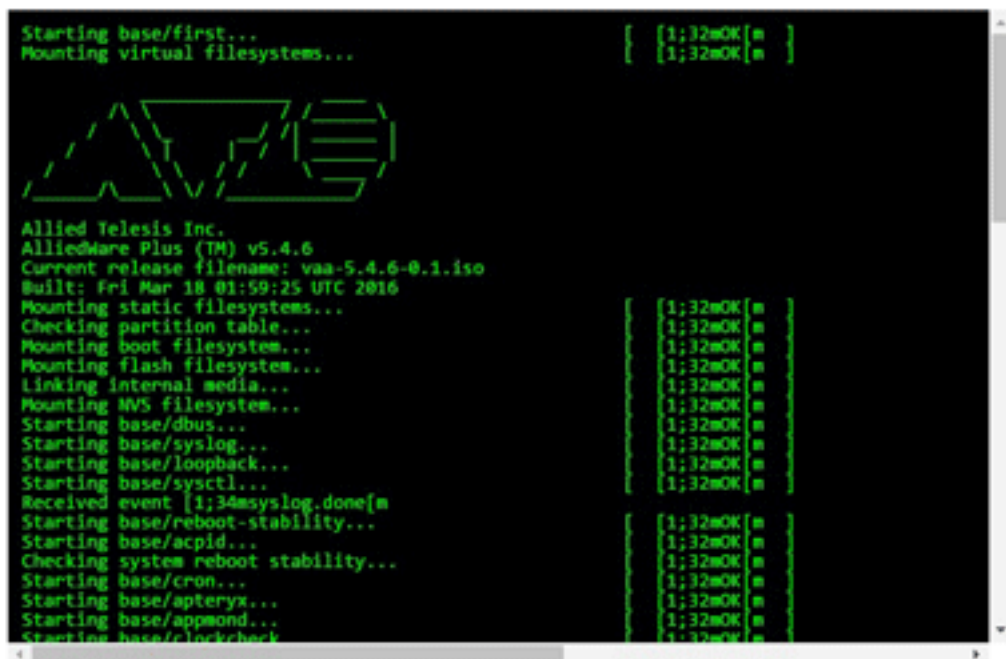


Note: AWS does not provide a virtual console to access your instances. Control of the instance is only possible via SSH. However, it is possible to view messages output to the internal console of AMF Plus Cloud as read-only logs.

To do this, open the **EC2** dashboard screen by clicking **Services > All services > EC2**. Click **Instances** under **Instances** from the left menu. Select the instance you created earlier, then select **Actions > Monitor and troubleshoot > Get system log** at the top of the screen.



A read-only log is displayed, so confirm that the message at startup is displayed as follows.



If you don't see anything in the read-only log, wait a few minutes and try refreshing the display. Log files are not updated in real time; they are updated according to a refresh timer determined by AWS.

Create and configure an internet gateway

VPCs are not connected to the Internet by default. To enable communication between your VPC and the Internet, you need to create an Internet gateway, attach it to your VPC, and set a default route in your VPC's route table by following the steps below.

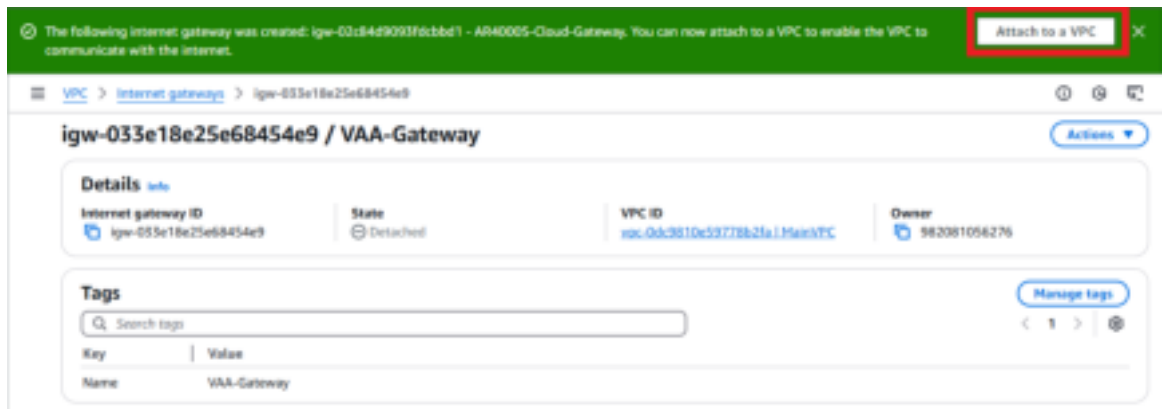
1. From the AWS Management Console's **Services** menu, select **All Services** > **VPC** to open the VPC dashboard screen. On the left menu, under **Virtual private cloud**, click **Internet gateway**. Click **Create internet gateway**.



2. The **Create internet gateway** screen will be displayed. Enter the following information and click **Create internet gateway**.

A screenshot of the 'Create internet gateway' screen in the AWS Management Console. The breadcrumb trail shows 'VPC > Internet gateways > Create internet gateway'. The title is 'Create internet gateway'. Below the title, there is a description: 'An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.' The 'Internet gateway settings' section has a 'Name tag' field with the value 'VAA-Gateway'. Below that is the 'Tags - optional' section, which includes a table with a 'Key' column (containing 'Name') and a 'Value - optional' column (containing 'VAA-Gateway'). There is a 'Remove' button next to the tag. At the bottom right, there is a 'Create internet gateway' button highlighted with a red box.

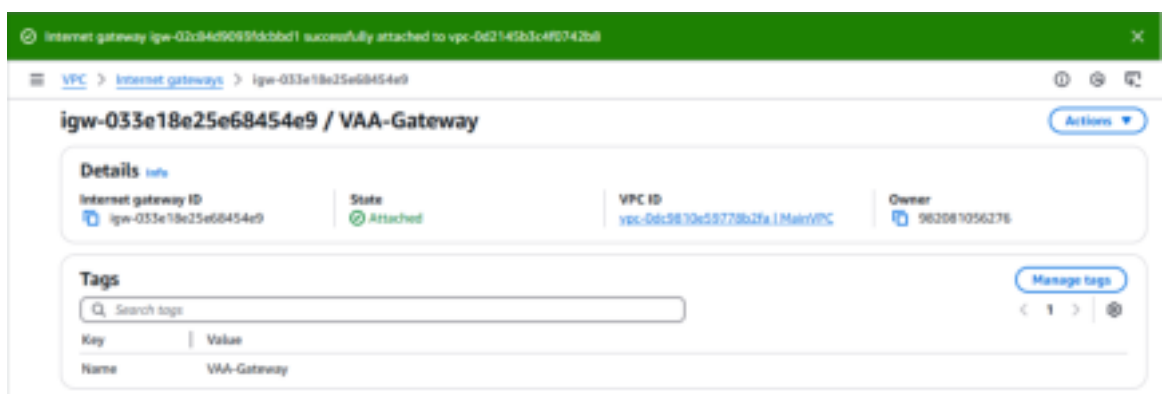
- When the Internet gateway is successfully created, the following screen will be displayed. Click **Attach to a VPC** in the upper right.



- The **Attach to VPC** screen will be displayed. Select the VPC you created earlier and click **Attach internet gateway**.



- The following is displayed when the Internet gateway attachment is successfully completed.



Create a route table

Next, create a route table, register a default route, and configure settings to allow communication from the VPC to the Internet via the Internet gateway.

1. On the left menu of the VPC dashboard, under **Virtual private cloud** click **Route tables**. Click **Create route table**.



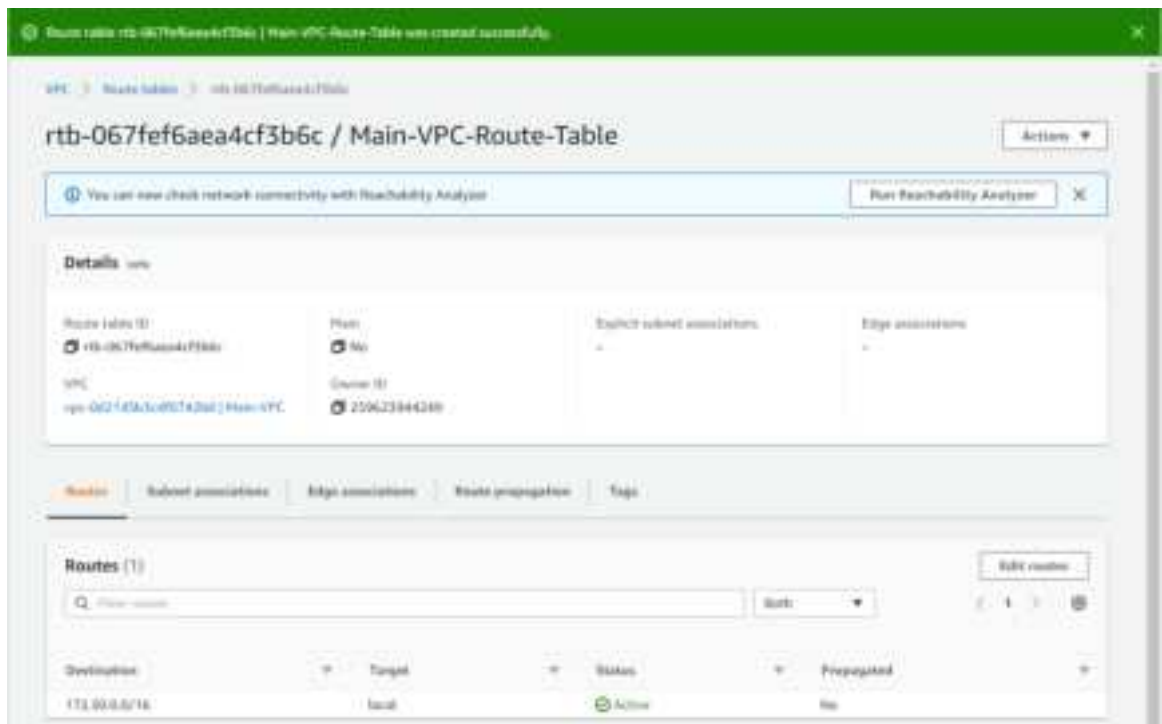
2. The **Create route table** screen will be displayed. Enter the following information and click **Create route table**.

 This screenshot shows the 'Create route table' form.

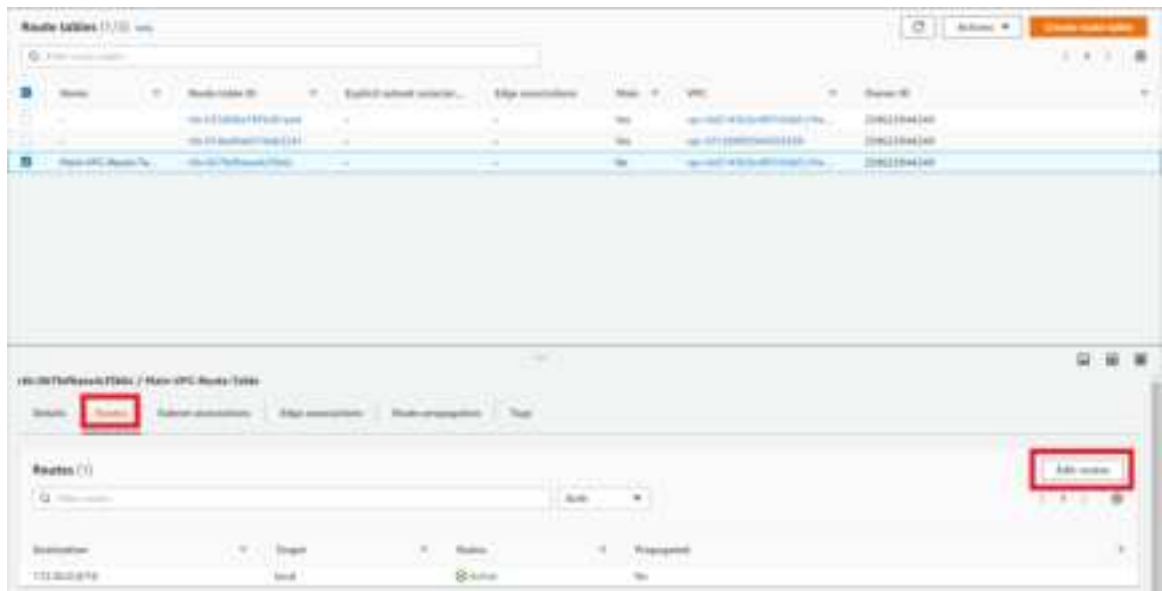
- Route table settings:**
 - Name - optional:** A text box containing 'Main-VPC-Route-Table'.
 - VPC:** A dropdown menu showing 'vpc-0d2145b3c4f0742b8 (Main-VPC)'.
- Tags:**
 - A section for adding tags with a 'Key' field (containing 'Name') and a 'Value - optional' field (containing 'Main-VPC-Route-Table').
 - An 'Add new tag' button.
 - A 'Remove' button.

 At the bottom right, there is a 'Cancel' button and a 'Create route table' button, which is highlighted with a red rectangular box.

3. When the route table is created successfully, the following screen will be displayed.



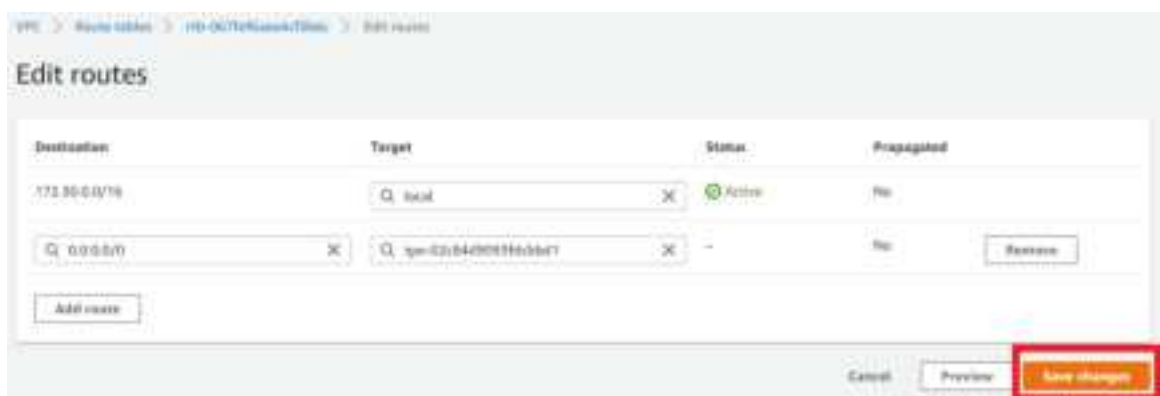
4. On the left menu, under **Virtual private cloud** click **Route tables**. Select the route table you created earlier, click the **Route** tab, and click **Edit Route**.



5. The **Edit routes** screen will be displayed. Click **Add route**.



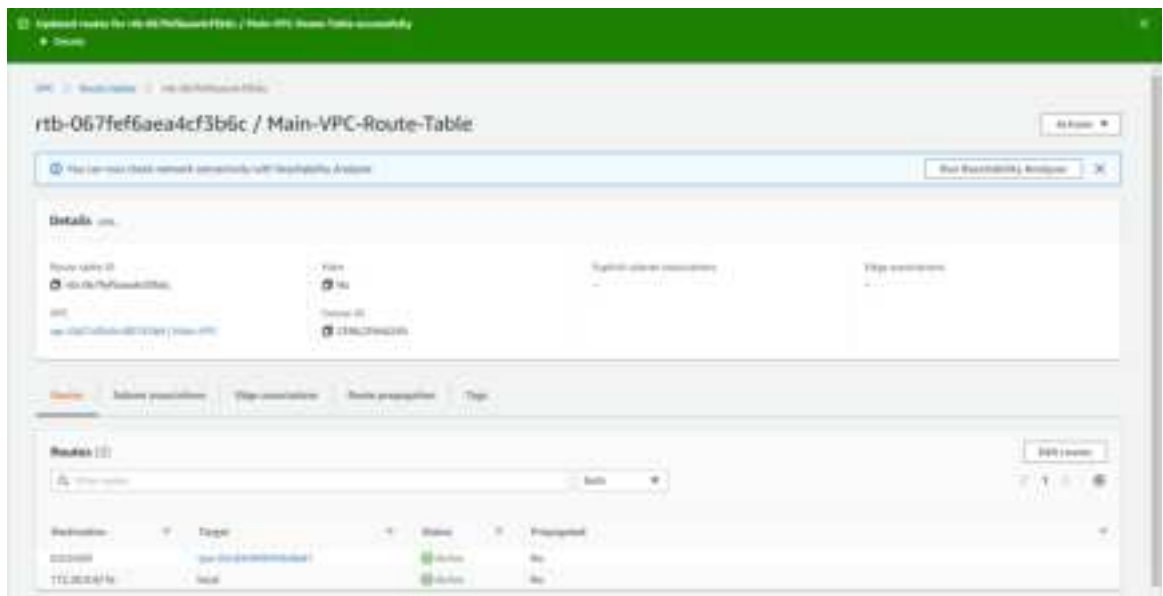
6. Configure additional routes as below and click **Save changes**.



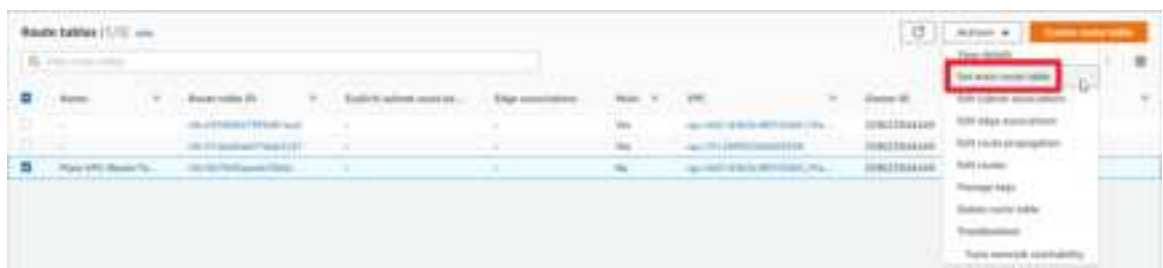
In the **Target** drop-down, if you select **Internet Gateway**, the Internet gateway created earlier will be displayed. Select it.



7. When the route editing is successfully completed, the following screen will be displayed.



8. On the left menu, under **Virtual private cloud** click **Route tables**. Select the route table you just created, and then click **Action > Set main route table**.



9. When the **Set main route table** screen appears, enter “set” and click **OK**.



10. When the main route table configuration is completed successfully, the following screen will be displayed.



SSH connection settings

Since AWS does not provide console access to instances (virtual machines), configuration and management of this product on AWS must be done via SSH (Secure Shell).

This section describes how to access the CLI of this product with public key authentication using an SSH key pair, using “PuTTY” for Windows and the `ssh` command for Ubuntu (Linux) as an SSH client.

SSH key pair

A cryptographic method that uses different keys for data encryption and decryption is called **asymmetric cryptography**, and the two keys used in that method are collectively called a **key pair** or **public key pair**. In asymmetric cryptography, data encrypted with one key of a key pair can only be decrypted with the other key of the pair.

SSH supports **public key authentication** using this property, and the key pair used in this authentication method is called an **SSH key pair**.

An SSH key pair consists of two keys:

- Public Key

A **public key** is a key that does not need to be kept secret. With SSH public key authentication, the user's public key is installed in advance on the access destination host (server, etc.). Public keys can be made public, so it's okay to install the same public key on multiple hosts.

In this product, the public key of the key pair set at the time of instance creation is automatically installed as the public key for the manager user at the time of initial startup. You can log in to this product as a manager user.

- Private Key

A **private key** is a key that is kept securely by its owner and should never be disclosed to anyone else. Since the private key is the only key that can decrypt data encrypted with the public key, the server takes advantage of this property in SSH public key authentication. This allows the server to compare the accessing user's key with the public key installed on the server, determine whether they possess the correct private key, and grant or deny access based on that result.

To access an instance of this product via SSH, you need to configure your SSH client software to authenticate using the private key that corresponds to the public key you entered when creating the instance.

Accessing the instance via SSH using "PuTTY"

The following section explains how to use "PuTTY", a typical SSH client for Windows, to connect to the instance via SSH.

For more details, please refer to the user guides for AWS and PuTTY.

Prerequisite

Download and install PuTTY from putty.org. The MSI installer or ZIP archive contain PuTTY and all of its companion utilities. You can also download each program individually. You will need to download the following program:

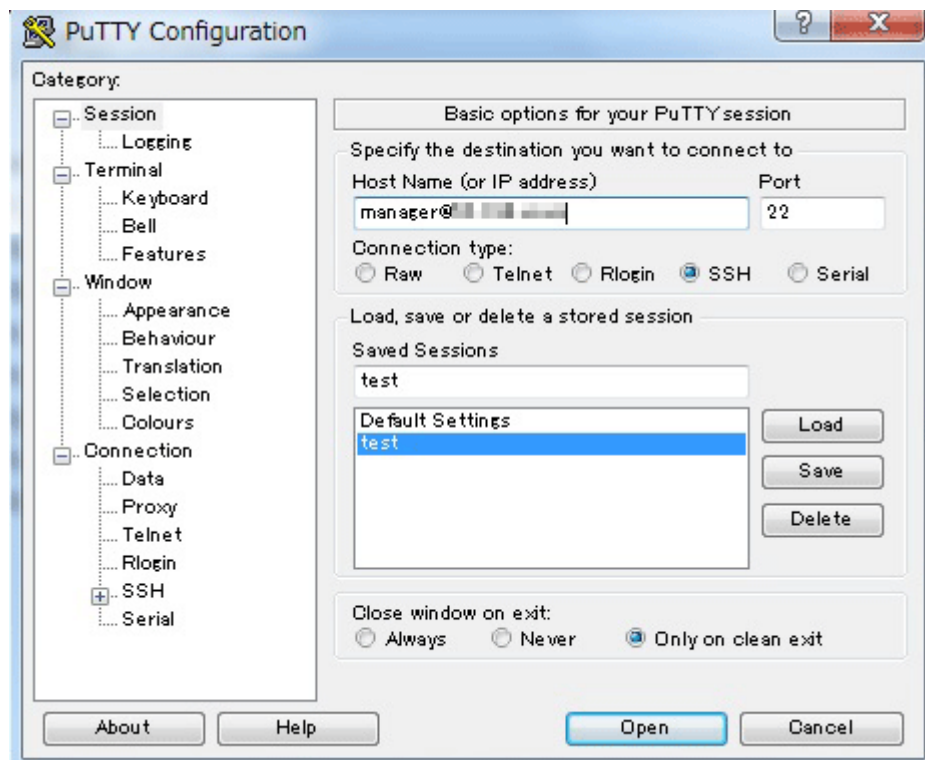
- `putty.exe` (used for SSH connection)

Make an SSH connection to the instance using PuTTY and PPK private key

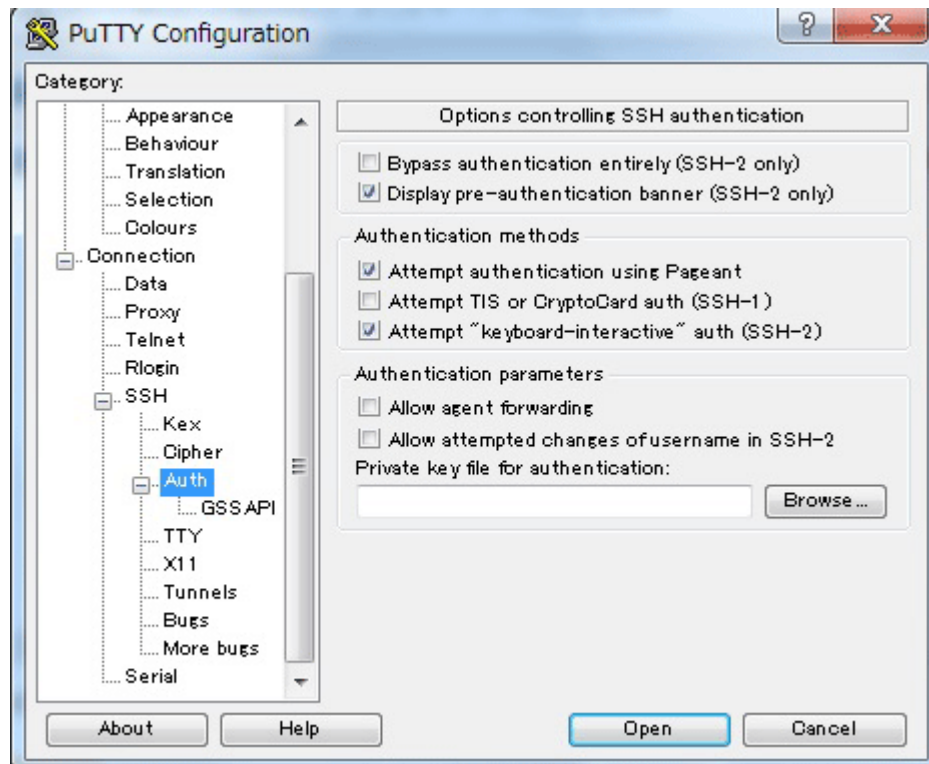
Using your private key, you can connect to your instance using PuTTY.

1. When PuTTY is opened, a window like the one shown below will be displayed. Enter "manager@ (the public IP address of this instance)" in the **Host Name** field.

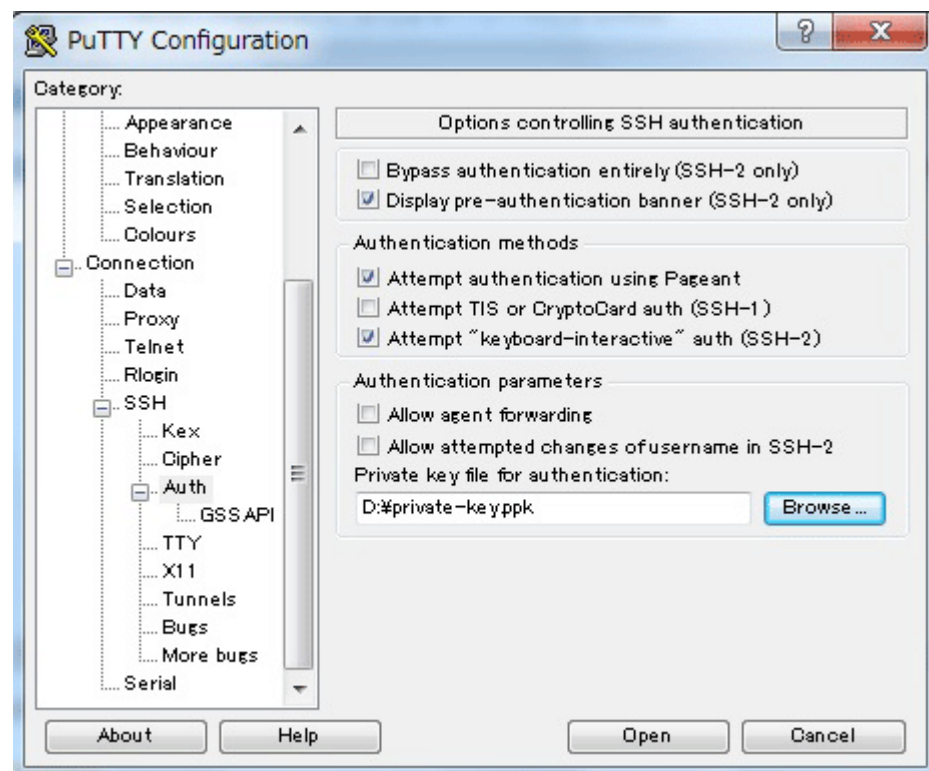
Note: You can check the public IP address from the instance screen of the EC2 dashboard.



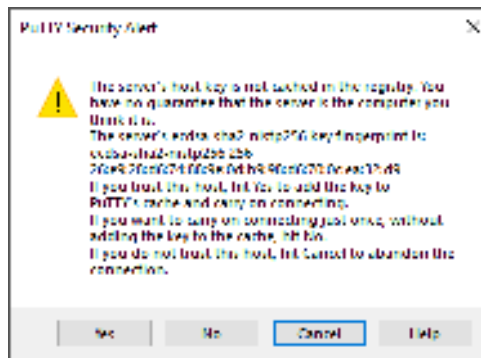
- Next, click **Connection > SSH > Auth** in the left panel.



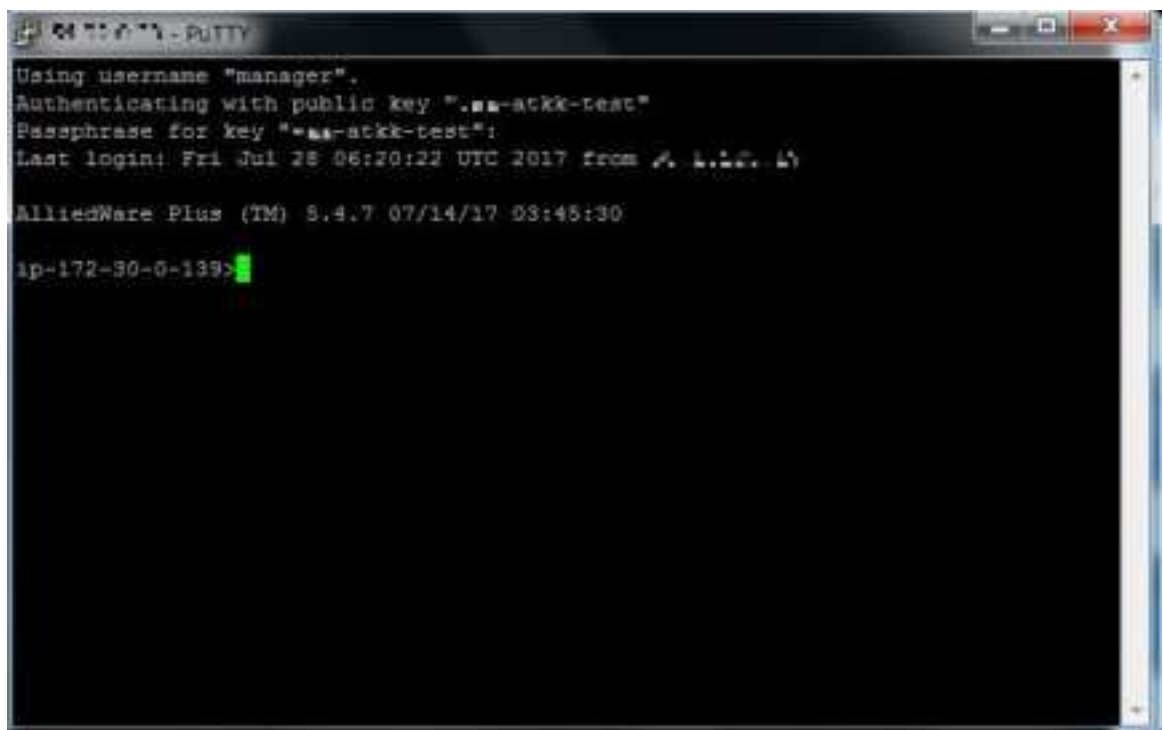
- Click the **Browse** button. Specify the PPK file of the private key saved earlier. Click **Open** to start an SSH session.



4. If this is your first time connecting to an instance of the product, a security alert dialog box will appear asking if you trust the host you are connecting to. Click **Yes** to save the key to your cache.



5. This completes the SSH connection to this product and displays the AlliedWare Plus CLI screen.



SSH connection to the instance using SSH client of Ubuntu (Linux)

The following describes how to SSH into this product using the standard OpenSSH SSH client in many Linux and UNIX-like environments.

See the man page for the ssh command for more information.

1. In the command shell, move the current directory to the location of the private key file downloaded from AWS when creating the instance.

```
ubuntu@ubuntu-pc:~/tmp$ cd ~/.ssh
```

Note: For security reasons, it is recommended that you set the permissions on the private key file to be read-only for the file owner and inaccessible for everyone else. You can do so with the following commands:

```
ubuntu@ubuntu-pc:~/.ssh$ chmod 400 vaa-atkk-test.pem
ubuntu@ubuntu-pc:~/.ssh$ ls -la vaa-atkk-test.pem
-r----- 1 vaa vaa 1696 Jul 15 15:06 vaa-atkk-test.pem
```

2. Make an SSH connection to the product with the **ssh** command. Use the **-i** option to specify the PEM file downloaded when creating the key pair on AWS EC2. **manager** is the default user name, and **XX.XXX.XX.XXX** is the public IP address of the product instance.

Note: You can check the public IP address of the product instance from the instance screen of the EC2 dashboard.

```
ubuntu@ubuntu-pc:~/.ssh$ ssh -i vaa-atkk-test.pem manager@XX.XXX.XX.XXX
```

3. When connecting to the server for the first time, you will be asked to confirm the public key of the server. Type “yes” and press the **Enter** key.

```
The authenticity of host 'XX.XXX.XX.XXX (XX.XXX.XX.XXX)' can't be established.
ECDSA key fingerprint is 7f:4e:5c:04:e2:bc:b1:dc:e5:27:b4:86:17:33:9c:0c.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'XX.XXX.XX.XXX' (ECDSA) to the list of known hosts.
```

4. This completes the SSH connection to this product and displays the AlliedWare Plus CLI screen.

```
Last login: Mon Jul 31 05:27:39 UTC 2017 from xx.x.xxx.xx

AlliedWare Plus(TM) 5.5.2 XX/XX/XX XX:XX:XX

ip-172-30-0-139>
```

Connecting to your local network

In order to use this product from the local network, it is necessary to connect AWS (VPC) and the local network. There are two ways to do this:

- Build an IPsec tunnel between the AMF Plus Cloud itself and the local network's VPN router.
- Build an IPsec tunnel between the AWS virtual private gateway and the VPN router on the local network.

The following sections will describe each method using our AT-AR4050S (hereafter referred to as “AR router”) as an example of the VPN router on the local network side.

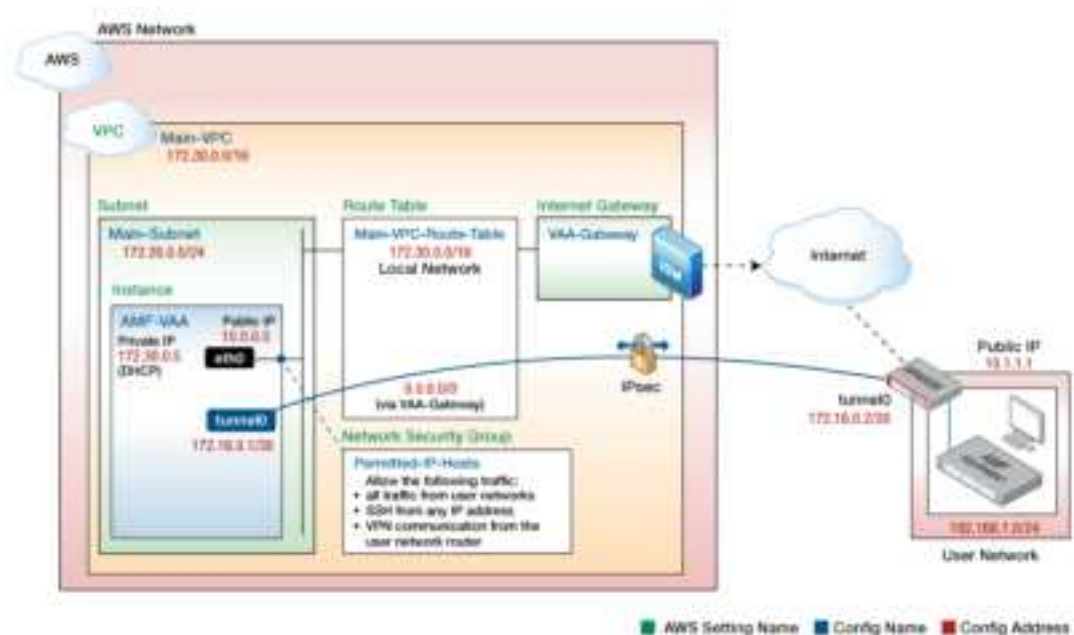
Note: This example assumes that the internet gateway has been set as explained in the [“Create an instance”](#) section.

How to use the VPN function of AMF Plus Cloud

In this configuration, this product itself becomes a VPN router and builds an IPsec tunnel with the VPN router (AR router) on the local network side.

Therefore, VPN connection settings are performed for this product itself. Settings on the AWS (VPC) side, such as a virtual private gateway, are not required, but for the security group, you add a rule to allow VPN communication from the AR router.

Note: The following is an example. Adjust the settings as appropriate to your actual environment.



	This product	AR router
Tunnel interface name	tunnel0	tunnel0
Tunnel operating mode	IPsec (IPv4)	IPsec (IPv4)
Tunnel end address (as viewed from this product)	172.30.0.5 (eth0's private IP)	10.1.1.1 (public IP)
Tunnel end address (as seen from AR router)	10.0.0.5 (instance public IP)	10.1.1.1 (public IP)
Address to set for tunnel I/F	172.16.0.1/30	172.16.0.2/30
ISAKMP Phase 1 ID	vaa0 (host name format string)	10.1.1.1 (IP address)
ISAKMP pre-shared key	abcdefghijklmnopqrstuvwxyz1234	

Note: You can check the public IP address of this product instance from the instance screen of the EC2 dashboard.

Settings on the AWS side

Add an inbound rule that allows VPN communication from the AR router to the security group applied to the instance of this product.

Type	Protocol	Port range	Source	Explanation
Custom UDP rule	UDP	500	10.1.1.1 (public IP address of AR router)	ISAKMP
Custom UDP rule	UDP	4500	10.1.1.1 (public IP address of AR router)	NAT-T (UDP-encap ISAKMP/ESP)

Settings on the AMF Plus Cloud side

This product has a VPN function equivalent to an AR router, so the settings are similar to those of the AR router described later.

However, this product has a private IP address (172.30.0.5) set, and the public IP address (10.0.0.5) of this product has been converted by the NAT function of AWS. In order to correctly identify this product when connecting to ISAKMP, it is necessary to set the tunnel local name to send the name of the local device (host name format string).

1. Set the ISAKMP pre-shared key to be used with the AR router (10.1.1.1). Use the **crypto isakmp** key command for this.

```
crypto isakmp key abcdefghijklmnopqrstuvwxyz1234 address 10.1.1.1
```

2. Create IPsec tunnel interface tunnel0. To do this, create a tunnel interface with the **interface** command and set the following information:
 - Local side tunnel end address (tunnel source). Specify the eth0 interface of this product
 - Remote side tunnel end address (tunnel destination). Specify the public IP address of the AR router.
 - ISAKMP local name (tunnel local name). Specify a name so that the AR router can identify this product.
 - Tunnelling method (tunnel mode ipsec)
 - Application of IPsec protection to the tunnel interface (tunnel protection ipsec)
 - IP address of the tunnel interface (ip address)
 - MTU of the tunnel interface (mtu)

```
interface tunnel 0
 tunnel source eth0
 tunnel destination 10.1.1.1
 tunnel local name arcloud
 tunnel mode ipsec ipv4
 tunnel protection ipsec
 ip address 172.16.0.1/30
 mtu 1300
```

3. Set a route to the local network (192.168.1.0/24). Use the **ip route** command for this. However, until the VPN connection is enabled, it will be set so that the same route cannot be used.

```
ip route 192.168.1.0/24 tunnel0
ip route 192.168.1.0/24 null 254
```

Settings on the AR router side

Next, configure the VPN settings on the AR router side, which is the VPN router on the local network.

Note: Here we assume that the AR router is connected to the Internet via the ppp0 interface. Also, it is assumed that Internet connection settings and AMF Plus Cloud settings have been completed.

As mentioned above, this product has a private IP address (172.30.0.5), and the public IP address (10.0.0.5) of this product has been converted by the NAT function of AWS. On the router side, it is necessary to specify the same name as that set for this product in the **tunnel remote name** so that this product can be identified correctly during ISAKMP connection.

1. Set the ISAKMP pre-shared key to be used with this product. Use the **crypto isakmp key** command for this.

Since the public IP of this product is actually NAT-converted, this product is identified here by a string ID in the form of a host name.

```
crypto isakmp key abcdefghijklmnopqrstuvwxyz1234 hostname arcloud
```

2. Create IPsec tunnel interface tunnel0. To do this, create a tunnel interface with the interface command and set the following information:
 - Local side tunnel end address (tunnel source). Specify the ppp0 interface of the AR router.
 - Remote side tunnel end address (tunnel destination). Specify the public IP address of this product.
 - ISAKMP remote name (tunnel local name). In order to identify the other party via NAT, specify the same name as set in this product.
 - Tunnelling method (tunnel mode ipsec)
 - Application of IPsec protection to the tunnel interface (tunnel protection ipsec)
 - IP address of the tunnel interface (ip address)
 - MSS rewrite setting on tunnel interface (ip tcp adjust-mss)
 - MTU of tunnel interface (mtu)

```
interface tunnel 0
 tunnel source ppp0
 tunnel destination 10.0.0.5
 tunnel remote name arcloud
 tunnel mode ipsec ipv4
 tunnel protection ipsec
 ip address 172.16.0.2/30
 ip tcp adjust-mss 1260
 mtu 1300
```

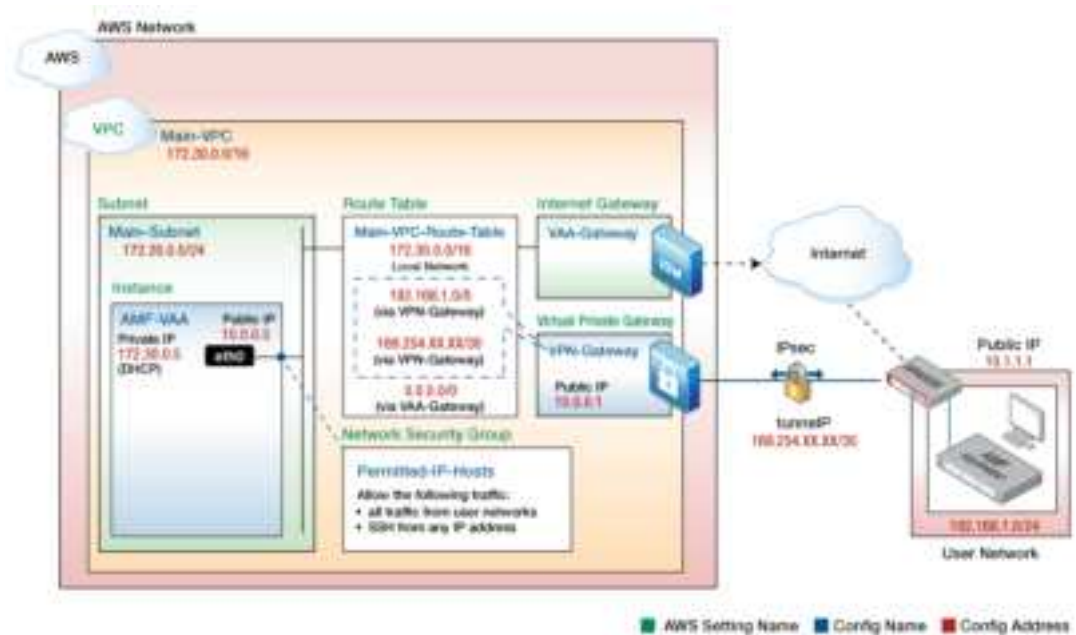
3. Set the route to this product (172.30.0.5/32). Use the **ip route** command for this. However, until the VPN connection is enabled, it will be set so that the same route cannot be used.

```
ip route 172.30.0.5/32 tunnel0
ip route 172.30.0.5/32 null 254
```

At this point, IP communication between this product on AWS and the local network can be established.

How to use AWS (VPC) VPN function

The basic configuration for connecting a VPC and a local network using the VPN function of AWS (VPC) is as follows.



This configuration utilizes a virtual private gateway provided by AWS (VPC) as a VPN router. Therefore, VPN connection settings are made for AWS (VPC). No settings are required on the product side.

Settings on the AWS side

The AWS-side components required to establish a VPN connection between AWS and your network are:

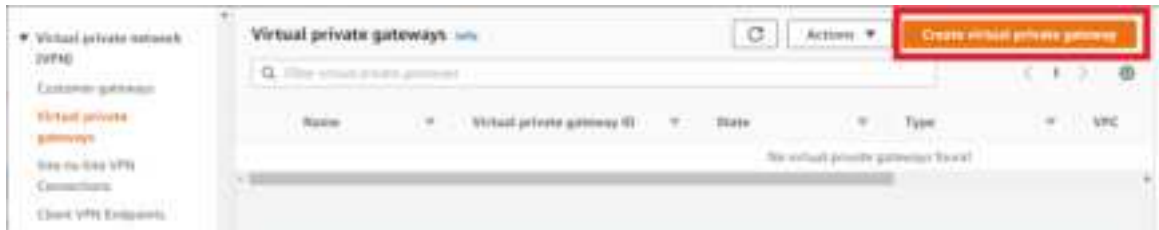
- Virtual Private Gateway - Virtual VPN router on AWS side
- VPN connection - A collection of information necessary for VPN connection between AWS and local network

For more information on VPN terminology in AWS (VPC), please refer to Amazon's user guide.

Create a virtual private gateway

Create a virtual private gateway which is a VPN router on the AWS side.

1. From the **Services** menu of the AWS Management Console, select **All Services** > **VPC** to open the VPC dashboard screen. Then, from the left menu, select **Virtual private gateways** under **Virtual private network (VPN)**, then click **Create virtual private gateway**.



2. The **Create virtual private gateway** screen will be displayed. Set as follows and click **Create virtual private gateway**.

 This screenshot displays the 'Create virtual private gateway' form. At the top, there's a breadcrumb trail: 'VPC > Virtual private gateways > Create virtual private gateway'. Below this is the title 'Create virtual private gateway' with an 'Info' link. A descriptive sentence states: 'A virtual private gateway is the VPN concentrator on the Amazon side of the site-to-site VPN connection.' The form is divided into two main sections: 'Details' and 'Tags'. In the 'Details' section, the 'Name tag - optional' field contains the text 'VPN-Gateway'. Below it, there's a note: 'Value must be 255 characters or less in length.' The 'Autonomous System Number (ASN)' section has two radio buttons: 'Amazon default ASN' (which is selected) and 'Custom ASN'. The 'Tags' section includes a description of tags and a table with one tag: 'Name' as the key and 'VPN-Gateway' as the value. There are 'X' icons to remove the tag and an 'Add new tag' button. At the bottom right, there are 'Cancel' and 'Create virtual private gateway' buttons, with the latter highlighted by a red box.

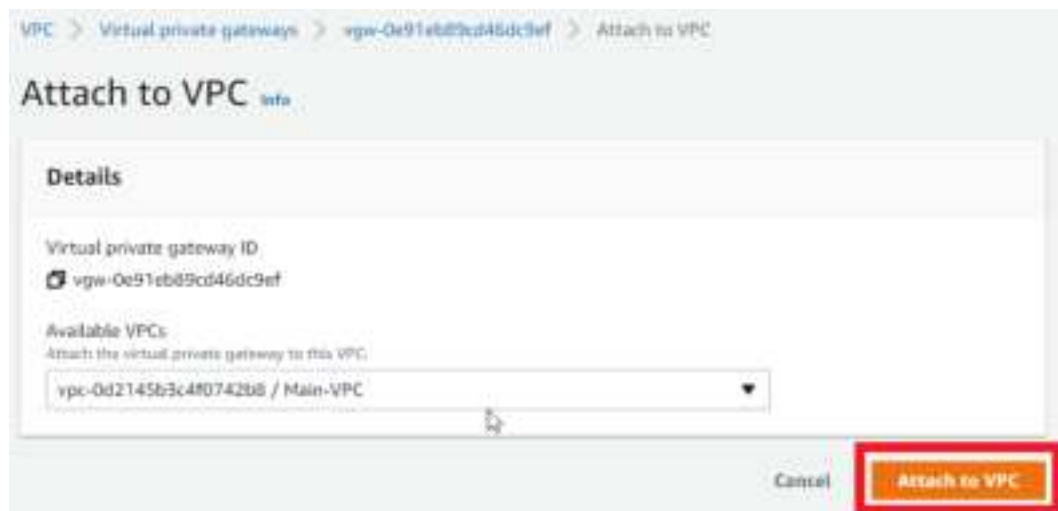
3. If the virtual private gateway is successfully created, you will see a screen like the one below.



4. Select **Actions** and click **Attach to VPC**.



5. The **Attach to VPC** screen will be displayed. Select the VPC you created earlier and click **Attach to VPC**.



6. When the attachment to the VPC is completed successfully, the following screen will be displayed.



Create a VPN connection

1. From the left menu of the VPC dashboard screen, click **Site-to-Site VPN Connections** under **Virtual private network (VPN)** and click **Create VPN connection**.



2. The **Create VPN connection** screen will be displayed. Set as follows and click **Create VPN connection**.

 This screenshot shows the 'Create VPN connection' form. The breadcrumb trail at the top is 'VPC > VPN connections > Create VPN connection'. The title is 'Create VPN connection' with an 'info' link. Below the title is a subtitle: 'Select the resources and additional configuration options that you want to use for the site-to-site VPN connection.' The form is divided into a 'Details' section.

- Name tag – optional:** A text box contains 'VPN-to-internal-AMF-Network'. A note below says 'Value must be 255 characters or less in length.'
- Target gateway type:** Three radio buttons are present: 'Virtual private gateway' (selected), 'Transit gateway', and 'Not associated'.
- Virtual private gateway:** A dropdown menu shows 'vgw-Oe91eb89cd46dc9ef / VPN-Gateway'.
- Customer gateway:** Two radio buttons are present: 'Existing' and 'New' (selected).
- IP address:** A text box contains '111.108.37.27'. A note below says 'Specify the IP address for your customer gateway device's external interface.'

Certificate ARN
The ARN of a private certificate provisioned in AWS Certificate Manager (ACM).

BGP ASN [info](#)
The ASN of your customer gateway device.

Value must be in 1 - 2147483647 range.

Routing options [info](#)

☐ Dynamic (requires BGP)

☒ Static

Static IP prefixes [info](#)

Local IPv4 network CIDR - optional
The IPv4 CIDR range on the customer gateway (on-premises) side that is allowed to communicate over the VPN tunnels. The default is 0.0.0.0/0.

Remote IPv4 network CIDR - optional
The IPv4 CIDR range on the AWS side that is allowed to communicate over the VPN tunnels. The default is 0.0.0.0/0.

Tunnel 1 options - optional [info](#)

Tunnel 2 options - optional [info](#)

Tags
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs. Name tag helps you track your resources more easily. We recommend adding Name tag.

Key	Value - optional	
<input type="text" value="Name"/>	<input type="text" value="VPN-to-internal-AMF-Network"/>	<input type="button" value="Remove"/>

You can add 49 more tags.

- Once the VPN connection is successfully created, you will see a screen similar to the one below.

The VPN console shows the VPN connection details.

VPN connections (1/1)

Name	VPN ID	Status	Virtual private gateway	Transit gateway	Customer gateway	Customer gateway ARN	Local IP
VPN-to-internal-AMF-Network	vgn-0123456789012345	Pending	vgn-0123456789012345		vgn-0123456789012345	arn:aws:ec2:us-east-1:123456789012:vpn-gw/vgn-0123456789012345	192.168.1.0/24

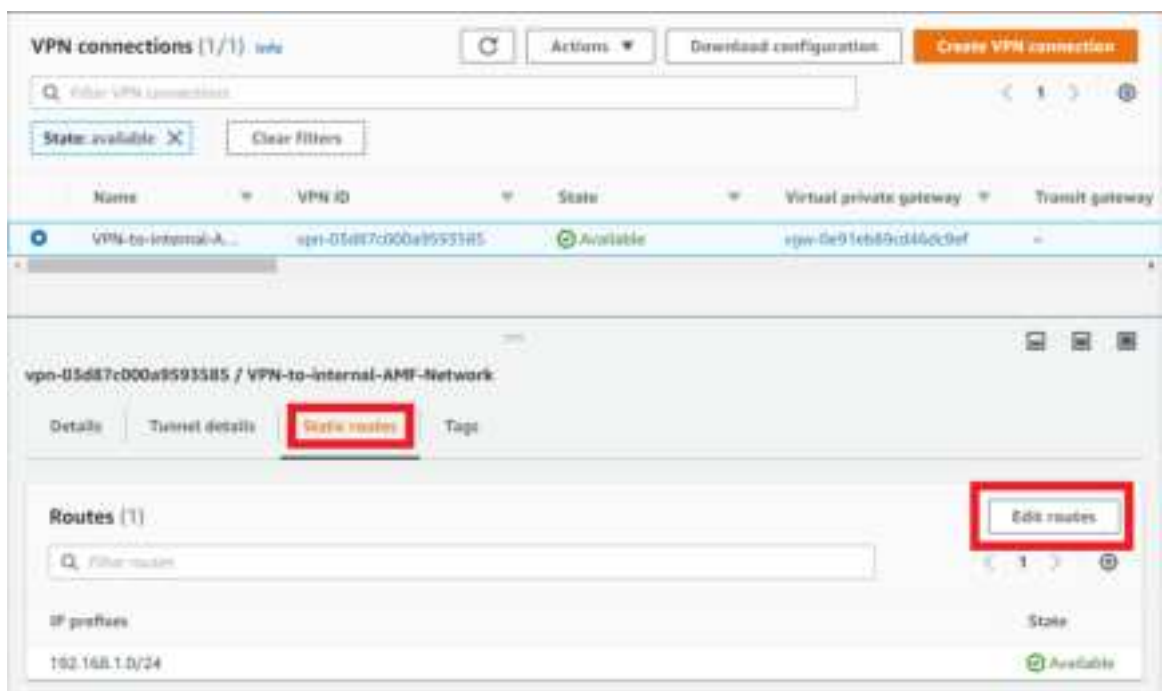
Add Static IP Prefix

If you want to communicate (ping, etc.) from the AR router to AWS via the tunnel, you need to tell the VPN gateway on the AWS side the range of link-local addresses used on the tunnel. Otherwise, even if the packet arrives from the AR router to AWS, the return packet will be discarded by the VPN gateway.

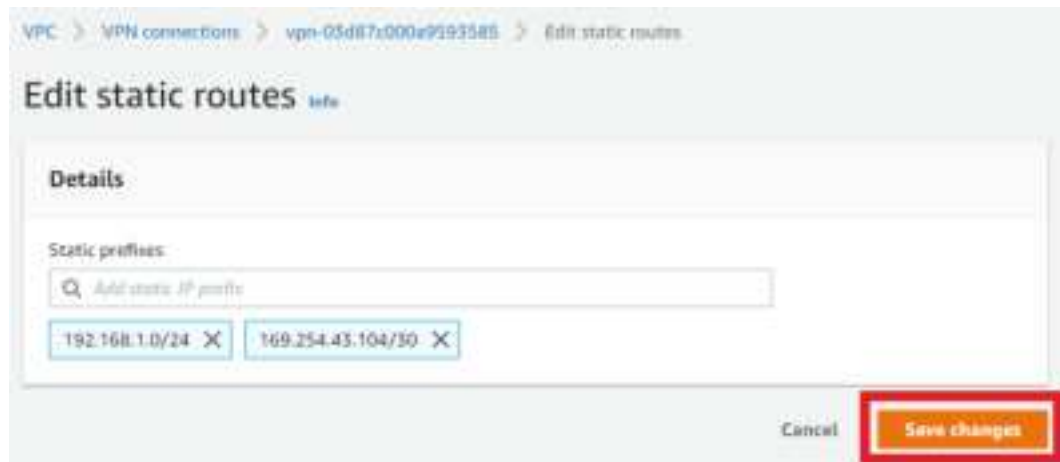
This can be addressed by adding the link-local prefix as a static route.

Register a static route as follows:

1. Click **Site-to-Site VPN Connections** under **Virtual private network (VPN)** from the left menu of the VPC dashboard screen. Select your VPN, click on the **Static routes** tab, and click on **Edit routes**.



- The **Edit static routes** screen will appear, add a static IP prefix and click **Save changes**. Check the link-local address used on the tunnel in the “[Tunnel settings](#)” section.



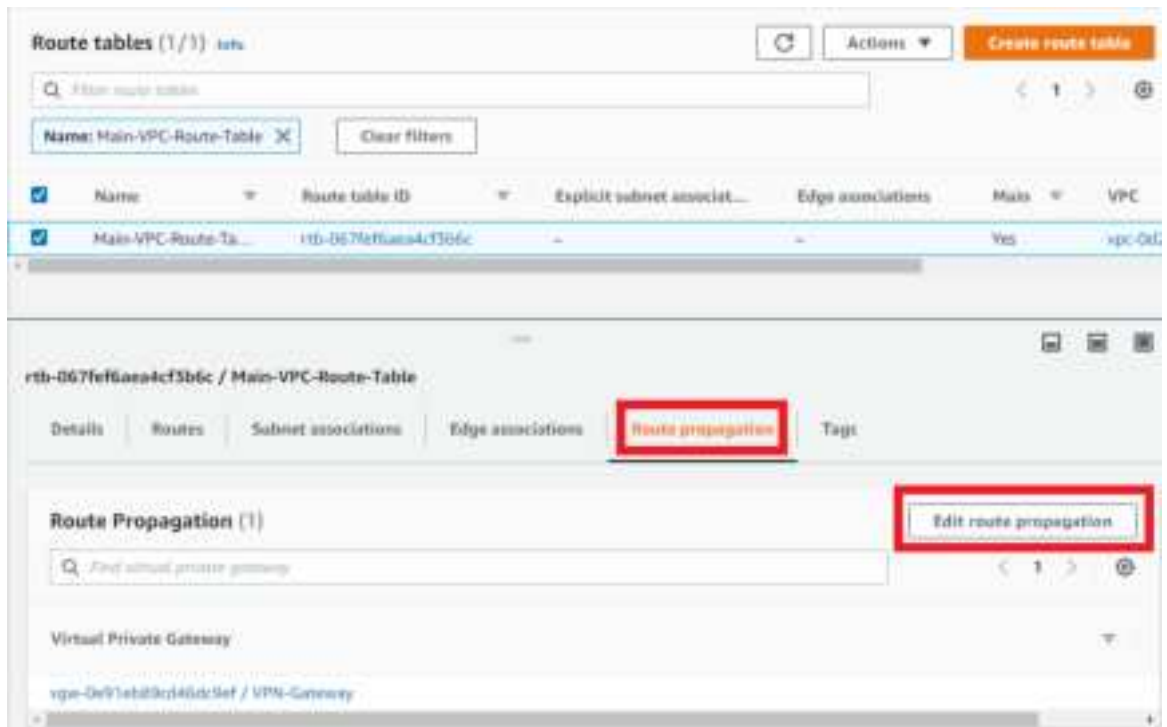
- If the static IP prefix is added successfully, you will see a screen like the one below.



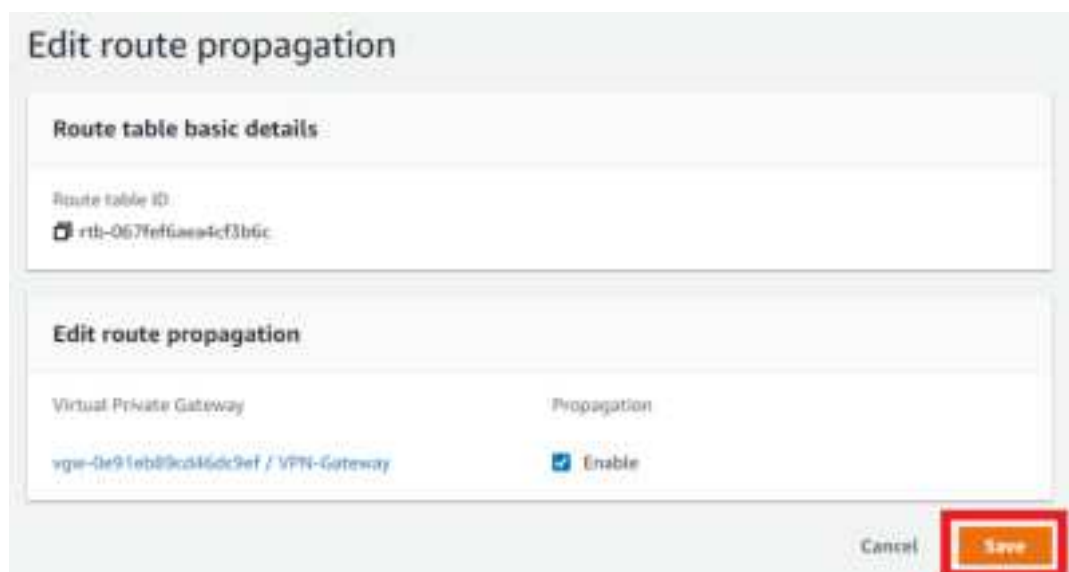
Enable route propagation

In order for VPN static IP prefixes (static routes) to be properly installed in the routing table, route propagation must be enabled. Otherwise, VPN static route traffic may not be routed correctly.

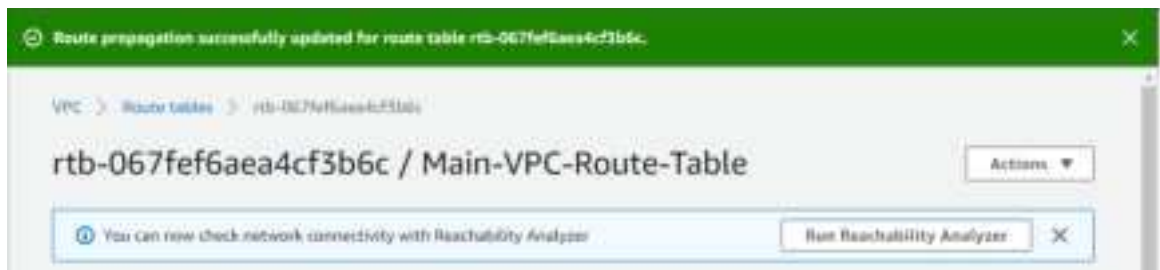
1. From the left menu of the VPC dashboard screen, click **Route Tables** under **Virtual private cloud**. Select the route table you created earlier, click the **Route propagation** tab, and click **Edit route propagation**.



2. The **Edit route propagation** screen will be displayed. Check **Enable** in the **Propagation** column and click **Save**.



- After successfully editing the route propagation, you will see a screen like the one below.



Settings on the AR router side

Next, we will explain the IPsec-related settings of the AR router, which is the VPN router on the local network side.

For network configuration, see the [“How to use AWS \(VPC\) VPN function”](#) section.

Here we assume that the AR router is connected to the Internet via the ppp0 interface.

Also, it is assumed that the settings on the AWS side have been completed. See the [“Settings on the AWS side”](#) section.

Once the VPN settings on the AWS side are complete, you will be able to download configuration samples for various VPN routers from the AWS dashboard.

The following explains how to set the AR4050S based on the setting sample of the Cisco Systems ISR series.

The reason for using the setting sample for the ISR series instead of the general-purpose setting sample is that the latter is closer to the setting of the AR4050S.

- From the left menu of the VPC dashboard screen, click **Site-to-Site VPN connections** under **Virtual private network (VPN)**. Select your VPN and click **Download configuration**.



2. The **Download configuration** screen will be displayed. Set as follows and click **Download**.

Download configuration

Choose the sample configuration you wish to download based on your customer gateway. Please note these are samples, and will need modification to use Advanced Algorithms, Certificates, and/or IPv6.

Vendor
The manufacturer of the customer gateway device (for example, Cisco Systems, Inc.):
Cisco Systems, Inc.

Platform
The class of the customer gateway device (for example, I-Series):
ISR Series Routers

Software
The operating system running on the customer gateway device (for example, ScreenOS):
IOS 12.4*

IKE version
The IKE version you are using for your VPN connection:
Rev 1

Cancel Download

The downloaded configuration sample has many sections, but this manual will extract only the important parts and show the configuration for the ISR series and for the AR4050S in comparison.

The important sections in the configuration sample are:

- Custom ISAKMP profile
- Key
- Custom IPSEC profile
- Assign Profile to Tunnel Peer
- tunnel

ISAKMP profile configuration (“Policy” on Cisco)

Corresponding part of the configuration sample for Cisco ISR.

```
crypto isakmp policy 200
  encryption aes 128
  authentication pre-share
  group 2
  lifetime 28800
  hash sha
  exit
```

AR4050S settings.

```
awplus(config)# crypto isakmp profile AWS-ISAKMP-Phase-1
awplus(config-isakmp-profile)# transform 1 integrity sha1 encryption aes128 group
2
awplus(config-isakmp-profile)# lifetime 28800
awplus(config-isakmp-profile)# dpd-interval 10
awplus(config-isakmp-profile)# version 1 mode main
awplus(config-isakmp-profile)# end
awplus#
```

Note: For IKE exchange mode, Cisco automatically tries both modes (aggressive, main), but AR4050S requires manual configuration.

Use the **show isakmp profile** command to check the settings.

```
awplus# show isakmp profile AWS-ISAKMP-Phase-1
ISAKMP Profile: AWS-ISAKMP-Phase-1
  Version: IKEv1
  Mode: Main
  Authentication: PSK
  Expiry: 8h
  DPD Interval: 10s
  DPD Timeout: 150s
  Transforms:
    Integrity Encryption DH Group
    1 SHA1 AES128 2
```

ISAKMP pre-shared key setting

Corresponding part of the configuration sample for Cisco ISR.

```
crypto keyring keyring-vpn-4234d12b-0
  local-address 10.1.1.1
  pre-shared-key address 10.0.0.1 key j3mqY_4dtzOHG7uP9mREjNkQxyeqnmEc
  exit
```

AR4050S settings.

```
awplus(config)# crypto isakmp key j3mqY_4dtzOHG7uP9mREjNkQxyeqnmEc address
10.0.0.1
```

Use the **show isakmp key** command to check the settings.

```
awplus# show isakmp key
Hostname/IP address Key
-----
10.0.0.1 j3mqY_4dtzOHG7uP9mRE
                                     jNkQxyeqnmEc
```

Custom ISAKMP Profile Assignment to AWS Peers

AR4050S settings.

```
awplus(config)# crypto isakmp peer address 10.0.0.1 profile AWS-ISAKMP-Phase-1
```

Use the **show isakmp peer** command to check the settings.

```
awplus# show isakmp peer
Peer Profile (* incomplete) Key
-----
10.0.0.1 AWS-ISAKMP-Phase-1 PSK
```

IPsec settings

Corresponding part of the configuration sample for Cisco ISR.

```
crypto IPsec transform-set IPsec-prop-vpn-4234d12b-0 esp-aes 128 esp-sha-hmac
mode tunnel
exit
```

AR4050S settings.

```
awplus(config)# crypto IPsec profile AWS-IPSEC-Phase-2
awplus(config-IPsec-profile)# transform 1 protocol esp integrity sha1 encryption
aes128
awplus(config-IPsec-profile)# pfs 2
awplus(config-IPsec-profile)# lifetime seconds 3600
awplus(config-IPsec-profile)# exit
awplus(config)# exit
awplus#
```

Use the **show ipsec profile** command to check the settings.

```
awplus# show ipsec profile AWS-IPSEC-Phase-2
IPsec Profile: AWS-IPSEC-Phase-2
Replay-window: 32
Expiry: 1h
PFS group: 2
Transforms:
    Protocol Integrity Encryption
    1 ESP SHA1 AES128
```

Tunnel settings

Corresponding part of the configuration sample for Cisco ISR.

```
interface Tunnel1
ip address 169.254.XX.XX 255.255.255.252
ip virtual-reassembly
tunnel source 10.1.1.1
tunnel destination 10.0.0.1
tunnel mode IPsec ipv4
tunnel protection IPsec profile IPsec-vpn-4234d12b-0
! This option causes the router to reduce the Maximum Segment Size of
! TCP packets to prevent packet fragmentation.
ip tcp adjust-mss 1387
no shutdown
exit
```

AR4050S settings.

```
awplus(config)# int tunnel1
awplus(config-if)# mtu 1434
awplus(config-if)# ip address 169.254.XX.XX/30
awplus(config-if)# tunnel source 10.1.1.1
awplus(config-if)# tunnel destination 10.0.0.1
awplus(config-if)# tunnel mode IPsec ipv4
awplus(config-if)# tunnel protection IPsec profile AWS-IPSEC-Phase-2
awplus(config-if)# ip tcp adjust-mss 1387
awplus(config-if)# end
```

Use the **show ip interface** and **show interface** commands to check the settings.

```
awplus# show ip interface brief
Interface          IP-Address          Status              Protocol
eth1                unassigned          admin up            running
eth2                unassigned          admin up            down
lo                  unassigned          admin up            running
vlan1               unassigned          admin up            down
vlan10              192.168.1.0/24      admin up            running
tunnel1             169.254.XX.XX/30    admin up            running
ppp0                10.1.1.1/32         admin up            running

awplus# show interface tunnel1
Interface tunnel1
  Link is UP, administrative state is UP
  Hardware is Tunnel
  IPv4 address 169.254.XX.XX/30 point-to-point 169.254.XX.XX
  index 14 metric 1 mtu 1434
  IPv4 mss 1387
  <UP,POINT-TO-POINT,RUNNING,MULTICAST>
  SNMP link-status traps: Disabled
  Tunnel source 10.1.1.1, destination 10.0.0.1
  Tunnel name local 10.1.1.1, remote 10.0.0.1
  Tunnel protocol/transport IPsec ipv4, key disabled, sequencing disabled
  Checksumming of packets disabled, path MTU discovery disabled
  Tunnel protection via IPsec (profile "AWS-IPSEC-Phase-2")
    input packets 0, bytes 0, dropped 0, multicast packets 0
    output packets 0, bytes 0, multicast packets 0 broadcast packets 0
  Time since last state change: 0 days 00:21:30
```


Note: While the tunnel interface is 'UP', the tunnel does not track the state of the peer. This means that the tunnel is ready to initiate connections or respond to peer initiation. To check if the tunnel is working, try pinging the link-local address of your AWS router (169.254.XXX.XXX) from your AT-AR4050S. If the ping is successful, the tunnel is up and working. Try pinging other desired networks to see if the routing is working as desired, and configure static routing if necessary.

Routing settings

In this example, the AR4050S does not have a default route. Use the following command to register the public IP address of the AWS router and the static route to the subnet to which this product belongs.

```
ip route 0.0.0.0/0 ppp0
ip route 172.30.0.0/24 169.254.XX.XX
```

For communication initiated from the AR4050S, the settings for correctly returning the return packet from this product to the AR4050S are configured in the [“IPsec settings”](#) section.

At this point, IP communication between this product on AWS and the local network can be established.

Connection with tenant networks (multi-tenant mode)

Note: The following explanation assumes that the basic settings for single mode in the [“Connecting to your local network”](#) section have been completed. If a connection between AMF Cloud and the user network is not required, that part can be omitted.

In multi-tenant mode, to use this product from each tenant network, you need to configure communication between the AMF container for the relevant tenant and the tenant network. This is done by building an L2TPv3 tunnel protected by IPsec between AMF Cloud itself and the VPN router of the tenant network. You then use AMF Cloud's bridge function to bridge the L2TPv3 tunnel to each container.

We will explain this method using an example where our AT-AR4050S (referred to as the “AR router”) is used as the VPN router on the tenant network side.

Note: The following are the minimum settings required to connect Azure to your network. During actual operation, please add appropriate access control and security settings using functions such as network security groups according to the requirements of your environment. Also, please design the network configuration appropriately to suit your actual environment.

How to use AMF Cloud's VPN function

The basic configuration for connecting AMF Cloud to a tenant network using the VPN function of AMF Cloud is described below.

In this configuration, AMF Cloud itself acts as a VPN router and bridge. You build an L2TPv3 + IPsec tunnel between the tenant network VPN router and bridging the L2TPv3 tunnel to each container using the bridge function of AMF Cloud.

Therefore, the VPN connection is configured for AMF Cloud itself. There is no configuration on the Azure side, such as the virtual network gateway, but for the network security group, you add a rule to allow VPN communication from the AR router.

In addition, in this configuration, the communication paths between each container and the tenant network are completely separated. Each tenant can specify its own IP address (IP addresses can overlap between tenants).

Note: The following is a reference example, so please adjust the settings as appropriate in your actual environment.

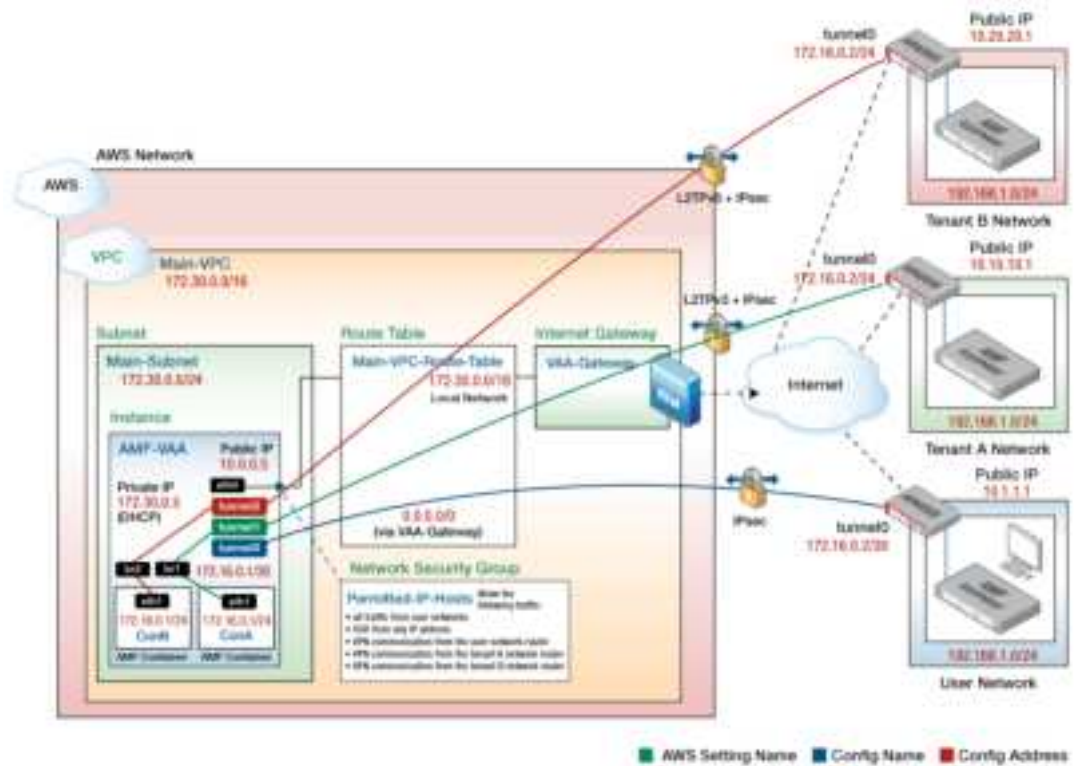


Table 1: Tenant A network connection parameters

	AMF CLOUD	AR ROUTER
Tunnel Interface Name	tunnel1	tunnel0
Tunnel Operation Mode	L2TPv3 + IPsec (IPv4)	L2TPv3 + IPsec (IPv4)
Tunnel end address (as seen from AMF Cloud)	172.30.0.5 (the private IP of eth0)	10.10.10.1 (public IP)
Tunnel end address (as seen from the AR router)	10.0.0.5 (the public IP of the instance)	10.10.10.1 (public IP)
Container Name (Bridge)	ConA (br1)	-
Address to be set for the tunnel I/F	-	172.16.0.2/24
Address to be set on eth1 of the container	172.16.0.1/24	-
ISAKMP Phase 1 ID	vaa1 (hostname format string)	10.10.10.1 (IP address)
ISAKMP pre-shared key	efghijklmnopqrstuvwxyz1234abcd	
L2TPv3 ID	11	12

Table 2: Tenant B network connection parameters

	AMF CLOUD	AR ROUTER
Tunnel Interface Name	tunnel2	tunnel0
Tunnel Operation Mode	L2TPv3 + IPsec (IPv4)	L2TPv3 + IPsec (IPv4)
Tunnel end address (as seen from AMF Cloud)	172.30.0.5 (the private IP of eth0)	10.20.20.1 (public IP)

	AMF CLOUD	AR ROUTER
Tunnel end address (as seen from the AR router)	10.0.0.5 (the public IP of the instance)	10.20.20.1 (public IP)
Container Name (Bridge)	ConB (br2)?	-
Address to be set for the tunnel I/F	-	172.16.0.2/24
Address to be set on eth1 of the container	172.16.0.1/24	-
ISAKMP Phase 1 ID	vaa2 (hostname format string)	10.20.20.1 (IP address)
ISAKMP pre-shared key	ijklmnopqrstuvwxyz1234abcdefgh	
L2TPv3 ID	21	22

Note: The public IP address of the AMF Cloud virtual machine can be confirmed in Azure, for example from the virtual machine's **Overview**, **Network**, or **Properties** screen.

Note: In this configuration, to show that the networks of each AMF container are independent, the same IP addresses are set for the AMF containers **ConA** and **ConB**, and the networks of Tenants A and B. However, this is to show that it is possible to configure overlapping VLANs and IP addresses between AMF containers, and is not a required setting.

Note: The following explanation assumes that the basic settings for single mode in the [“Connecting to your local network”](#) section have been completed. If a connection between AMF Cloud and the user network is not required, that part can be omitted.

Settings on the AWS side

Add an inbound rule that allows VPN communication from the AR router to the security group applied to the instance of this product.

Type	Protocol	Port range	Source	Explanation
Custom UDP rule	UDP	500	10.10.10.1 (public IP address of AR router)	ISAKMP A
Custom UDP rule	UDP	500	10.20.20.1 (public IP address of AR router)	ISAKMP B
Custom UDP rule	UDP	4500	10.10.10.1 (public IP address of AR router)	NAT-T (UDP-encap ISAKMP/ESP) A
Custom UDP rule	UDP	4500	10.20.20.1 (public IP address of AR router)	NAT-T (UDP-encap ISAKMP/ESP) B

AMF Cloud side settings

AMF Cloud has the same VPN function as the AR router, so the configuration is similar to that of the AR router, which will be described later.

However, since the AMF Cloud has a private IP address (172.30.0.5), and the AMF Cloud's public IP address (10.0.0.5) is converted by the Azure NAT function, the AR router must be configured to send the name of its own device (hostname format string) in **tunnel local name**. This means it can correctly identify the AMF Cloud during ISAKMP connection. This section shows the configuration

for establishing an L2TPv3 + IPsec tunnel between the AMF Cloud and the VPN routers of the Tenant A and B networks.

1. Using the **crypto isakmp key** command, set the ISAKMP pre-shared key to be used between the AR routers (10.10.10.1 and 10.20.20.1) of Tenant A and Tenant B.

Example:

```
crypto isakmp key efghijklmnopqrstuvwxyz1234abcd address 10.10.10.1
crypto isakmp key ijklmnopqrstuvwxyz1234abcdefgh address 10.20.20.1
```

2. Create L2TPv3 tunnel interfaces tunnel1 (for Tenant A) and tunnel2 (for Tenant B).

To do this, create a tunnel interface with the **interface** command and set the following information:

- Local side tunnel end address (**tunnel source**) - Specify the eth0 interface of the AMF Cloud
- Remote side tunnel end address (**tunnel destination**) - Specify the public IP address of the AR router
- ISAKMP local name (**tunnel local name**) - Specify an arbitrary string so that the AR router can identify the AMF Cloud
- L2TPv3 local ID (**tunnel local id**) - Assign an arbitrary number so that it is paired with the opposite side
- Tunneling method (**tunnel mode ipsec**)
- Application of IPsec protection to the tunnel interface (**tunnel protection ipsec**)

Example:

```
interface tunnel1
 tunnel source eth0
 tunnel destination 10.10.10.1
 tunnel local name vaa1
 tunnel local id 11
 tunnel remote id 12
 tunnel mode l2tp v3
 tunnel protection ipsec

interface tunnel2
 tunnel source eth0
 tunnel destination 10.20.20.1
 tunnel local name vaa2
 tunnel local id 21
 tunnel remote id 22
 tunnel mode l2tp v3
 tunnel protection ipsec
```

AR router settings

Next, we will explain the VPN settings on the AR router, which is the VPN router on the tenant network side. Here, we assume that the AR router is connected to the Internet via the ppp0 interface. We also assume that the Internet connection settings and the settings on the AMF Cloud side have been completed.

As mentioned above, a private IP address (172.30.0.5) is set for the AMF Cloud itself, and the public IP address (10.0.0.5) of the AMF Cloud is converted by the Azure NAT function. On the AR router side, you need to specify the same string as that set on the AMF Cloud side in tunnel remote name so that the AMF Cloud can be correctly identified during ISAKMP connection.

Tenant A side AR router

1. Set the ISAKMP pre-shared key to be used with AMF Cloud. To do this, use the **crypto isakmp key** command. Since the public IP of AMF Cloud is NAT translated, we identify AMF Cloud by a string ID in the form of a hostname.

Example:

```
crypto isakmp key efghijklmnopqrstuvwxyz1234abcd hostname vaa1
```

2. Create an L2TPv3 tunnel interface tunnel0.

To do this, create a tunnel interface with the **interface** command and set the following information:

- Local side tunnel end address (**tunnel source**) - Specify the ppp0 interface of the AR router
- Remote side tunnel end address (**tunnel destination**) - Specify the public IP address of the AMF Cloud
- ISAKMP remote name (**tunnel remote name**) - Specify the same string as set in the AMF Cloud to identify the other party via NAT
- L2TPv3 local ID (**tunnel local id**) - Assign an arbitrary number so that it is paired with the opposite side
- L2TPv3 remote ID (**tunnel remote id**) - Assign an arbitrary number so that it is paired with the opposite side
- Tunneling method (**tunnel mode ipsec**)
- Application of IPsec protection to the tunnel interface (**tunnel protection ipsec**)
- IP address of the tunnel interface (**ip address**)

Example:

```
interface tunnel0
 tunnel source ppp0
 tunnel destination 10.0.0.5
 tunnel remote name vaa1
 tunnel local id 12
 tunnel remote id 11
 tunnel mode l2tp v3
 tunnel protection ipsec
 ip address 172.16.0.2/24
```

Tenant B side AR router

1. Set the ISAKMP pre-shared key to be used with AMF Cloud. To do this, use the **crypto isakmp key** command. Since the public IP of AMF Cloud is NAT translated, we identify AMF Cloud by ISAKMP remote name.

Example:

```
crypto isakmp key ijklmnopqrstuvwxyz1234abcdefgh hostname vaa2
```

2. Create an L2TPv3 tunnel interface tunnel0.

To do this, create a tunnel interface with the **interface** command and set the following information.

- Local side tunnel end address (**tunnel source**) - Specify the ppp0 interface of the AR router
- Remote side tunnel end address (**tunnel destination**) - Specify the public IP address of the AMF Cloud
- ISAKMP remote name (**tunnel remote name**) - Specify the same string as set in the AMF Cloud to identify the other party via NAT
- L2TPv3 local ID (**tunnel local id**) - Assign an arbitrary number so that it is paired with the opposite side
- L2TPv3 remote ID (**tunnel remote id**) - Assign an arbitrary number so that it is paired with the opposite side
- Tunneling method (**tunnel mode ipsec**)
- Application of IPsec protection to the tunnel interface (**tunnel protection ipsec**)
- IP address of the tunnel interface (**ip address**)

Example:

```
interface tunnel0
 tunnel source ppp0
 tunnel destination 10.0.0.5
 tunnel remote name vaa2
 tunnel local id 22
 tunnel remote id 21
 tunnel mode l2tp v3
 tunnel protection ipsec
 ip address 172.16.0.2/24
```

Note: For further settings, see the [AMF Feature Overview and Configuration Guide](#).

Firmware update

To update the firmware of this product, use the **software-upgrade** command.

Prerequisite

It is necessary to download the maintenance firmware (ISO image file) of this product from our website and upload it to this product on AWS.

About ISO files and VHD files

The firmware for this product is distributed in the following two formats, each of which has a different purpose as follows:

- An ISO image file is used to update the firmware.
- The VHD image file is for uploading to AWS to create the AMI of this product.

For more information, see [“Create an Amazon Machine Image”](#).

The ISO image file provided on our website is for updating the firmware of this product that is already running on AWS.

Update procedure

To update the firmware of this product, log in to the CLI of this product and perform the following procedure.

1. Make sure the ISO image file exists on the file system.

```
awplus# dir
...
25499648 -rw- Jul 16 2022 20:45:45 vaa-5.5.3-2.2.iso
```

2. Specify the ISO image file using the **software-upgrade** command. A confirmation message will be displayed; verify the ISO is correct then enter “y”.

```
awplus# software-upgrade vaa-5.5.3-2.2.iso
Install this release to disk? (y/n): y
Upgrade succeeded, the changes will take effect after rebooting the device.
```

3. Reboot with new firmware.

```
awplus# reboot
```


Tips and troubleshooting

Lost network connection

This product has a mechanism called fail-safe mode as an automatic recovery method. The product enters failsafe mode when it detects that the network connection with AWS has been lost.

If this product cannot connect to some of the default servers that exist on AWS, it assumes that access to the management function is no longer possible and starts a 5-minute monitoring timer. If 5 minutes pass without the connection being restored, the product will restart with default settings.

This feature is primarily intended for automatic recovery from connectivity failures due to the following reasons:

- eth0 port shut-down
- Incorrect static IP address setting for eth0
- routing problems

When the product is launched with default settings, it can be accessed via SSH using the original SSH key pair assigned when the instance was created. In addition, the configuration file before restart will be renamed to “default_backup.cfg” and saved.

When the SSH server function is disabled

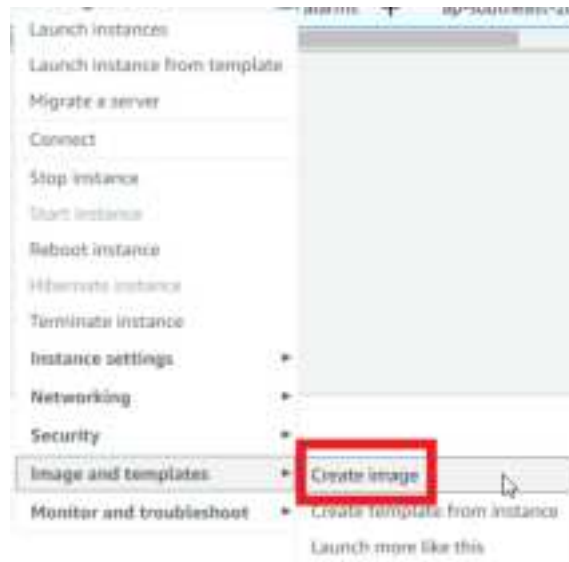
This product also starts a 5-minute monitoring timer when the SSH server function is disabled. This is because the management functions of this product can only be accessed via SSH.

If 5 minutes have passed with the SSH server function disabled, the product will restart with the default settings. The configuration file before restart will be renamed to “default_backup.cfg” and saved.

Creating an instance snapshot

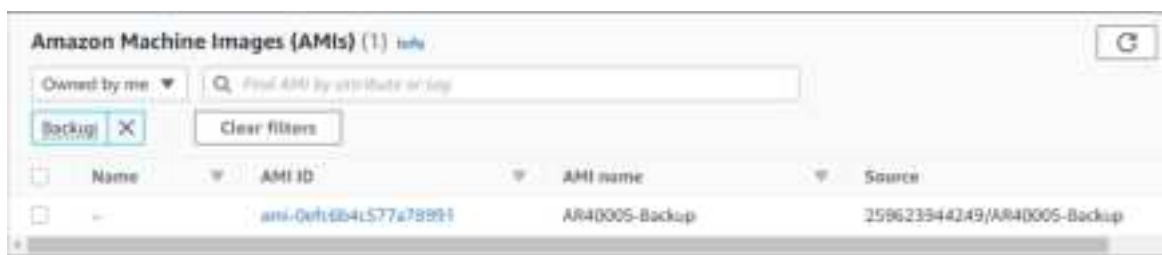
Follow the steps below to take a snapshot of an instance image while it is running normally. This snapshot can be used in case the connection to this product cannot be restored even with the above mechanism.

1. Click **Instances** on the EC2 Dashboard.
2. Select the desired instance, right-click, and select **Image and templates** > **Create image**.



3. Enter an **Image name**, and click **Create image**.

You can check the created snapshot image in **Image** > **AMI**.



To create a machine from this snapshot, create a new instance by selecting the created snapshot image in the **My AMIs** tab, using the same process described in the [“Create an instance”](#) section.

Note: If you have multiple AMIs of your own, when you click the **My AMIs** tab, a different one than the snapshot image you created may be selected. In that case, select the target snapshot from the drop-down list.

Note: If you recreate an instance from a snapshot, the MAC and IP addresses will be different than before. Therefore, it is necessary to manually reconfigure the network and re-register the annual license.

C613-04190-00 REV A



NETWORK SMARTER

North America Headquarters | 19800 North Creek Parkway | Suite 100 | Bothell | WA 98011 | USA | T: +1 800 424 4284 | F: +1 425 481 3895

Asia-Pacific Headquarters | 11 Tai Seng Link | Singapore | 534182 | T: +65 6383 3832 | F: +65 6383 3830

EMEA & CSA Operations | Incheonweg 7 | 1437 EK Rozenburg | The Netherlands | T: +31 20 7950020 | F: +31 20 7950021

alliedtelesis.com

© 2022 Allied Telesis, Inc. All rights reserved. Information in this document is subject to change without notice. All company names, logos, and product designs that are trademarks or registered trademarks are the property of their respective owners.