Security Gateway Manual

Amazon AWS

Feb 06, 2026

CONTENTS

The Netgate® pfSense® Plus Firewall/VPN/Router for Amazon AWS is a stateful firewall and VPN appliance. It is suitable for use as a VPN endpoint for mobile devices, laptops, and desktop computers to ensure that data sent over unsecured wireless networks or untrusted wired networks is encrypted using industry standard encryption algorithms. It can also be used to establish a connection between one or many sites with the internet or each other.

This AMI can be run in any region where EC2 offers service on various sizes of instances and can run on both x86-64/amd64 and arm64/Graviton instances. pfSense Plus for AWS is available in the AWS Marketplace.

CHAPTER

ONE

GETTING STARTED

1.1 Prerequisites and Requirements

Using a Netgate® appliance instance to protect VPC subnets requires the following:

- Setup can take 15 minutes to one hour, depending on the user's familiarity with the tools.
- An AWS Account.
- Familiarity with AWS networking.
- A VPC.
- One internet-facing subnet, to which the Netgate appliance instance will have its primary/WAN interface connected.
- One or more private subnets, to which the Netgate appliance instance will have its secondary/LAN interface (and possibly additional optional interfaces) connected.
- Separate routing tables for the internet-facing subnet and the private subnet(s)
- Separate security groups for the internet-facing subnet and the private subnet(s).
- An elastic IP address or Public IP address for the WAN interface of the appliance.

For the purposes of this guide, the VPC will contain two subnets (public and private) as well as an Internet Gateway. The end result should look like the following diagram:

If all of these are already in place with an existing VPC, feel free to skip ahead to Launching an Instance.

1.2 Choosing Instance Type and Sizing

There are a range of specifications to choose from, and this document walks through the viable potential choices.

1.2.1 Available EC2 Instance Types

An instance type will depend on the expected network throughput as well as the types of services the Netgate® appliance will provide.

For general firewall appliances that do not require high throughput, **t3** amd64 AWS product instances are General Purpose Burstable Performance Instances that provide a lower baseline level of CPU performance with the ability to burst above the baseline to meet occasional increases in performance demands.

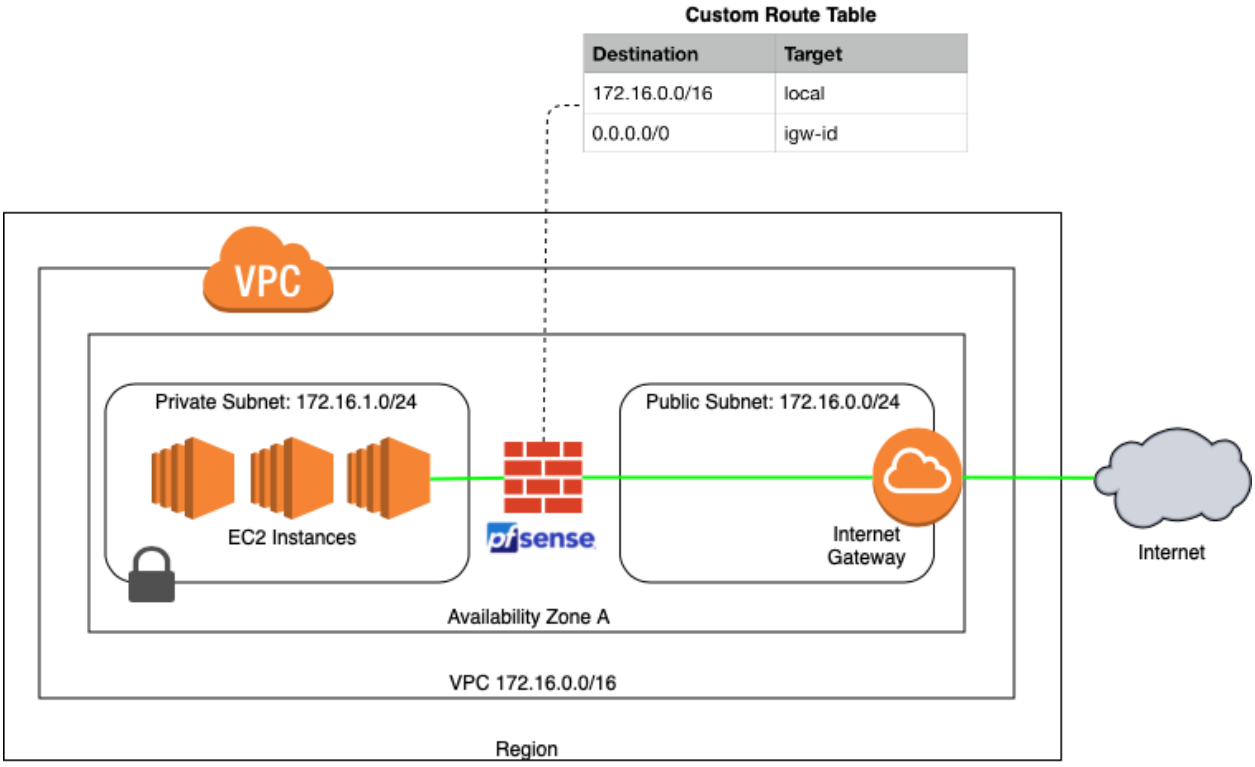


Fig. 1: Architecture Diagram

Warning: A **t3** instance will consume CPU credits while it exceeds the baseline CPU utilization value for its instance size (e.g. 20% for **t3.medium**). Consult the AWS T3 instance product information to find the baseline for each size. When CPU credits are exhausted, AWS limits the instance to its baseline CPU level even if CPU demand is high. This can lead to degraded performance, network timeouts, throughput problems, and other errors. If this happens on a regular basis, redeploy with a larger instance size or different type.

Businesses looking for higher VPN throughput while keeping costs manageable should consider arm64/Graviton instances.

Tip: If the appliance will provide advanced services like web proxying, IDS/IPS, or Server Load Balancing, consider an instance that provides more CPU and RAM, such as a **large** or **xlarge** subtype.

The available EC2 instance types are listed on the product pages for either the amd64 AWS product or the arm64/Graviton AWS product.

Note: pfSense Plus software cannot run on ".nano" size instances as they lack sufficient RAM for certain key functions to work, such as upgrades.

1.2.2 Sizing the EBS Volume

The Netgate appliance is only compatible with EBS storage. For general purpose firewalls, storage requirements will typically be small and the default **8GB** general purpose SSD volume should be more than enough.

In situations where the appliance may provide web proxying or caching to users, or other advanced features, consider increasing the volume size to something more appropriate, for example **64GB**.

1.3 AWS Service Limits

New services provisioned in a VPC may be assigned IP addresses or other resources, but Amazon puts limits on VPC resources per Region. Before provisioning a new resource, make sure to check these limits.

The following tables list the limits for Amazon VPC resources per Region. Unless indicated otherwise, requests can be made to increase these limits using the Amazon VPC limits form. For some of these limits, the current limit applied can be viewed using the **Limits** page of the Amazon EC2 console.

Note: If a limit increase is requested that applies per resource, AWS increases the limit for all resources in the Region. For example, the limit for security groups per VPC applies to all VPCs in the Region.

1.3.1 VPC and Subnets

Resource	Default limit	Comments
VPCs per Region	5	The limit for Internet gateways per Region is directly correlated to this one. Increasing this limit increases the limit on internet gateways per Region by the same amount.
Subnets per VPC	200	–
IPv4 CIDR blocks per VPC	5	This limit is made up of the primary CIDR block plus 4 secondary CIDR blocks.
IPv6 CIDR blocks per VPC	1	This limit cannot be increased.

1.3.2 DNS

For more information, see DNS Limits.

1.3.3 Elastic IP Addresses (IPv4)

Resource	Default limit	Comments
Elastic IP addresses per Region	5	This is the limit for the number of Elastic IP addresses for use in EC2-VPC. For Elastic IP addresses for use in EC2-Classic, see Amazon EC2 Limits in the Amazon Web Services General Reference.

1.3.4 Flow Logs

Resource	Default limit	Comments
Flow logs per single network interface, single subnet, or single VPC in a Region	2	This limit cannot be increased. There can effectively be 6 flow logs per network interface by creating 2 flow logs for the subnet, and 2 flow logs for the VPC in which the network interface resides.

1.3.5 Gateways

Resource	Default limit	Comments
Customer gateways per Region	50	–
Egress-only internet gateways per Region	5	This limit is directly correlated with the limit on VPCs per Region. To increase this limit, increase the limit on VPCs per Region. Only one egress-only internet gateway can attach to a VPC at a time.
Internet gateways per Region	5	This limit is directly correlated with the limit on VPCs per Region. To increase this limit, increase the limit on VPCs per Region. Only one internet gateway can be attached to a VPC at a time.
NAT gateways per Availability Zone	5	A NAT gateway in the pending, active, or deleting state counts against the limit.

1.4 Creating an IAM User in an AWS Account

A pfSense® Plus AMI uses AWS Identity and Access Management (IAM) accounts for administration. Every AWS account includes at least one user. For security reasons, the root account should not be used for day-to-day administration. This section describes the process of creating and using an IAM user account for administering the pfSense® Plus AMI.

See also:

To find out more about AWS security and credentials read Understanding and Getting Your Security Credentials.

There are multiple methods for creating users in IAM. The recommended method is to use the AWS Management Console. The process of creating a user and enabling that user to perform work tasks consists of the following steps:

1. Create the user.
2. Create credentials for the user.
3. As a best practice, create only the credentials that the user needs. For example, for a user who requires access only through the AWS Management Console, do not create access keys.

   Note: For cloud security the best practice is to limit access for the `root` account, so the `root` account is locked by default.

4. Grant the appropriate permissions to the user to administer the pfSense® Plus AMI.
5. Provide the user with the necessary sign-in information.
6. (Optional) Configure multi-factor authentication (MFA) for the user.

1.4.1 Creating IAM Users (Console)

The AWS Management Console can create IAM users.

To create one or more IAM users (console):

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Users** and then choose **Add user**.
3. Type the username for the new user. This is the name they will use to sign in to AWS. To add up to 10 users at once, choose **Add another user** for each additional user and type their usernames.
4. Usernames can be a combination of up to 64 letters, digits, and these characters: `+=,.@-`
5. Names must be unique within an account and are *not* case-sensitive.
6. Select **AWS Management Console access**. This creates a password for each new user.

   Choose one of the following options for **Console password**:

   Autogenerated password
   Each user gets a randomly generated password that meets the account password policy in effect (if any).

   Note: The **Final** page allows viewing or downloading the passwords.

   Custom password
   Each user is assigned a given password.

Tip: The best practice is to select **Require password reset** to ensure that users are forced to change their password the first time they sign in.

7. Click **Next**. On the **Set permissions** page, specify how to assign permissions to this new user(s). Choose one of the following three options:

   Add user to group
   Choose this option to assign the user(s) to one or more groups that already have permissions policies. IAM displays a list of the groups in the account, along with their attached policies.

   Select one or more existing groups or choose **Create group** to create a new group.

   Copy permissions from existing user
   Choose this option to copy all access rights from an existing user to the new user(s).

   Attach existing policies to user directly
   Choose this option to see a list of the managed policies in the account. Select the policies to attach to the new users or choose **Create policy** to open a new browser tab and create a new policy.

8. Choose **Next: Review** to see all the choices made up to this point. Choose **Create user** to proceed.
9. To view user access keys (access key IDs and secret access keys), choose **Show** next to each password and access key to display. To save the access keys, choose **Download .csv** and then save the file to a secure location.

Danger: This is the **only** opportunity to view or download the secret access keys, and users **must** have this information before they can use the AWS API. Save the user new access key ID and secret access key in a safe and secure place.

There is no way to access the secret keys again after this step.

10. Choose **Send email** next to each user to send a message with account information. This opens a local mail client with a draft that to customize and send. The email template includes the following details to each user:
    - Username
    - URL to the account sign-in page. Use the following example, substituting the correct account ID number or account alias:
    - https://AWS-account-ID or alias.signin.aws.amazon.com/console

Important: The user's password is **not** included in the generated email as email is not a secure communications channel. Provide passwords to the user through a secure method which complies with security policies set by the organization.

1.5 Using IAM Roles

AWS IAM Roles are used to delegate access to users, applications, or services that require controlled access to AWS resources. IAM Roles should be used to manage all Netgate® pfSense® Plus software instances. This unique role can be specified when launching a new instance, or attached to an existing instance.

The AWS Management Console is the recommended method for creating roles for use with pfSense® Plus software. The best practice is to create these roles based on the *principle of least privilege*, also known as the *principle of least authority*, which is the assignment of lowest needed privileges based on necessity. These instructions attempt to follow this principle.

1.5.1 Create Policy for pfSense Plus Software Management IAM Role

Create a custom policy that will be associated with an IAM role allowing access to the pfSense® Plus Management GUI running on an EC2 Instance.

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane of the console, select **Policies** then choose **Create Policy**.
3. Click the **Service** drop-down menu and select **EC2**.
4. In the **Actions** drop-down check the box next to **All EC2 actions (ec2:)**

   Note: If stricter policies are required for the actions that can be performed on the pfSense® Plus EC2 Instance, these can be set here.

   ---

5. Select the **Resources** drop-down arrow and review resulting warnings.
6. Click the **All resources** bubble
7. Select **Review policy**.
8. Populate the **Name** field (e.g. `pfSense_EC2_Access`) and **Description**, if desired.

   Note: Policy names must be unique within the AWS account, and the name of the policy cannot be changed once created.

   ---

9. Select **Create Policy**.

1.5.2 Create IAM Role for pfSense Plus Software Management

Create a role that an IAM user, or users within an IAM Group, can assume and use to connect to and manage pfSense® Plus running on an EC2 Instance.

Note: The administrator of the specified account can grant permission to assume this role to any IAM user in that account. To do this, the administrator attaches a policy to the user or a group that grants permission for the **sts:AssumeRole** action. That policy must specify the role's ARN as the Resource.

---

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane of the console, select **Roles** then choose **Create Role**.

3. Select the **Another AWS account** role type.
4. In the **Account ID** field, type the AWS account ID that will be allowed to access the destination resource.
5. The **Require external ID** checkbox should remain cleared unless granting permissions to users from an account not under the control of this organization. Reference AWS Documentation for External ID Roles in the event this is required.
6. The best practice is to restrict the role to users who sign in with multifactor authentication (MFA). Select **Require MFA** to add a condition to the role's trust policy to require MFA sign-in.
7. Select **Next: Permissions**.
8. Type the name of the previously created Custom policy in the search field. Check the box next to the correct Policy name.
9. Select **Next:Tags**

Note: IAM tags are key-value pairs that can be used to organize, track, or control access for this role. This is an optional step. More information can be found within AWS Documentation for Tagging IAM Entities.

---

10. Select **Next: Review**.
11. Populate the **Role name** field (e.g. `pfSense_Admin`) and Role description if desired.

Note: Role names must be unique within the AWS account, and the name of the role cannot be changed once created.

---

12. Review remaining configured settings then select **Create role**.

This role can now be assigned to an IAM User or all users in an IAM group allowing secure administrative access to the EC2 Instance(s) containing pfSense® Plus.

CHAPTER

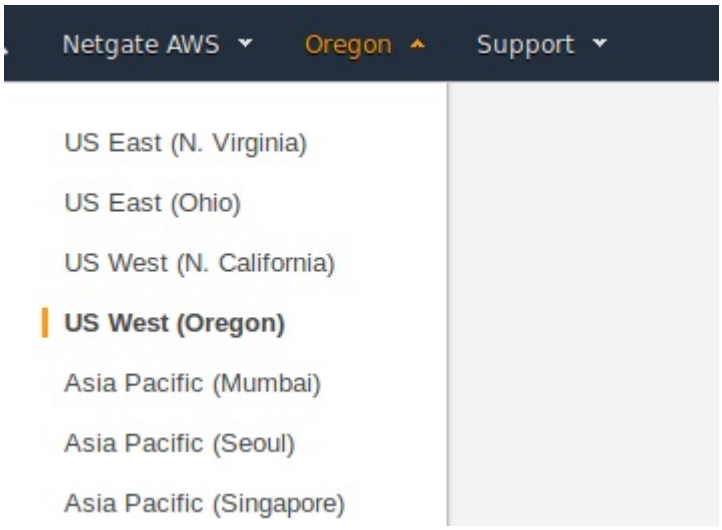
TWO

INSTANCE USAGE

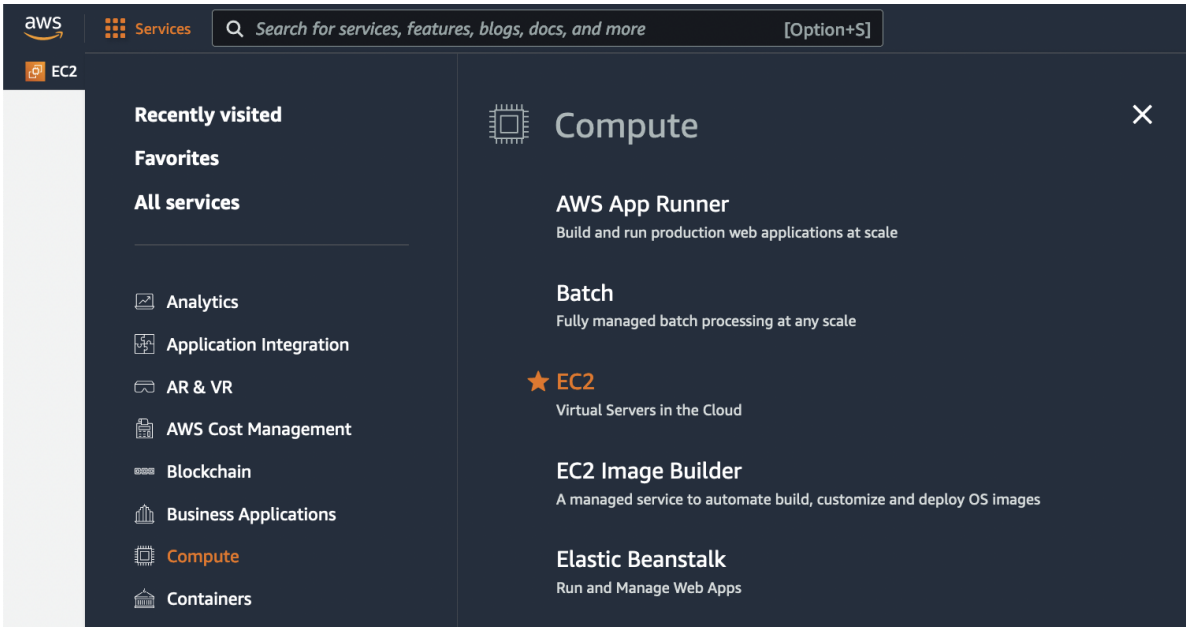
2.1 Launching an Instance

These instructions cover how to launch a new instance of the Netgate® pfSense® Plus firewall/VPN appliance from the Amazon EC2 Management Console.

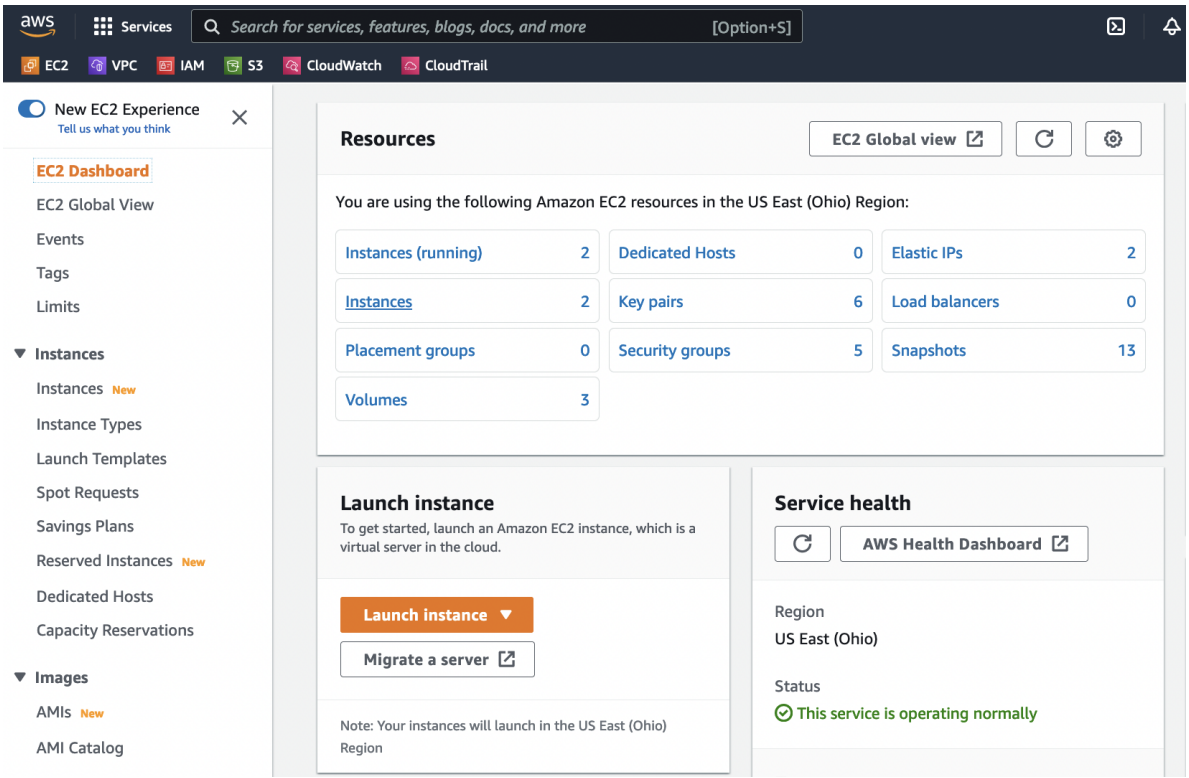
1. Select the region for the instance to run in using the region tab in the upper right corner of the page.



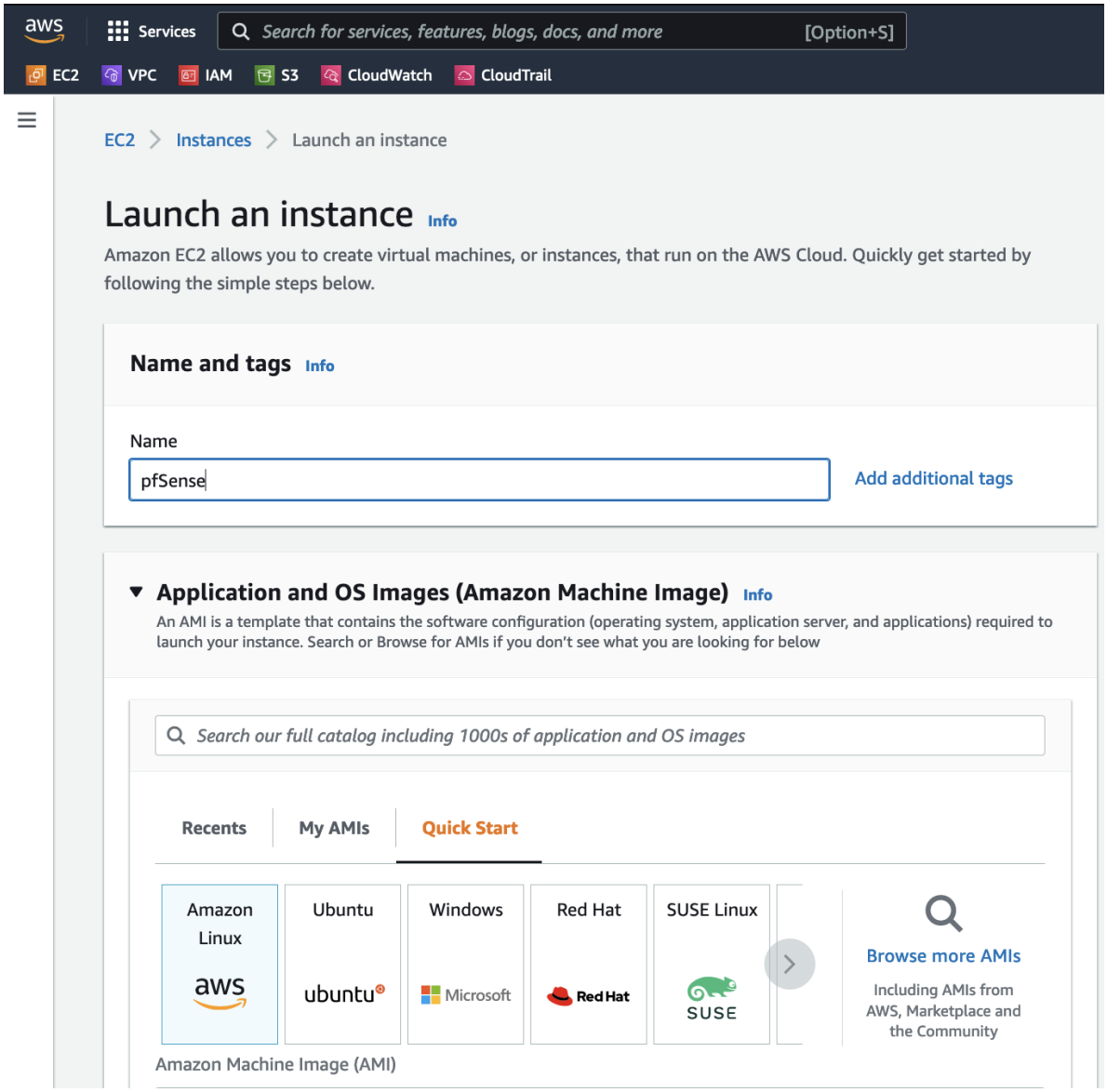
2. Select **Services** from the top navigation, and select **Compute** on the left navigation of the drop-down and then select **EC2** on the main section of the drop-down.



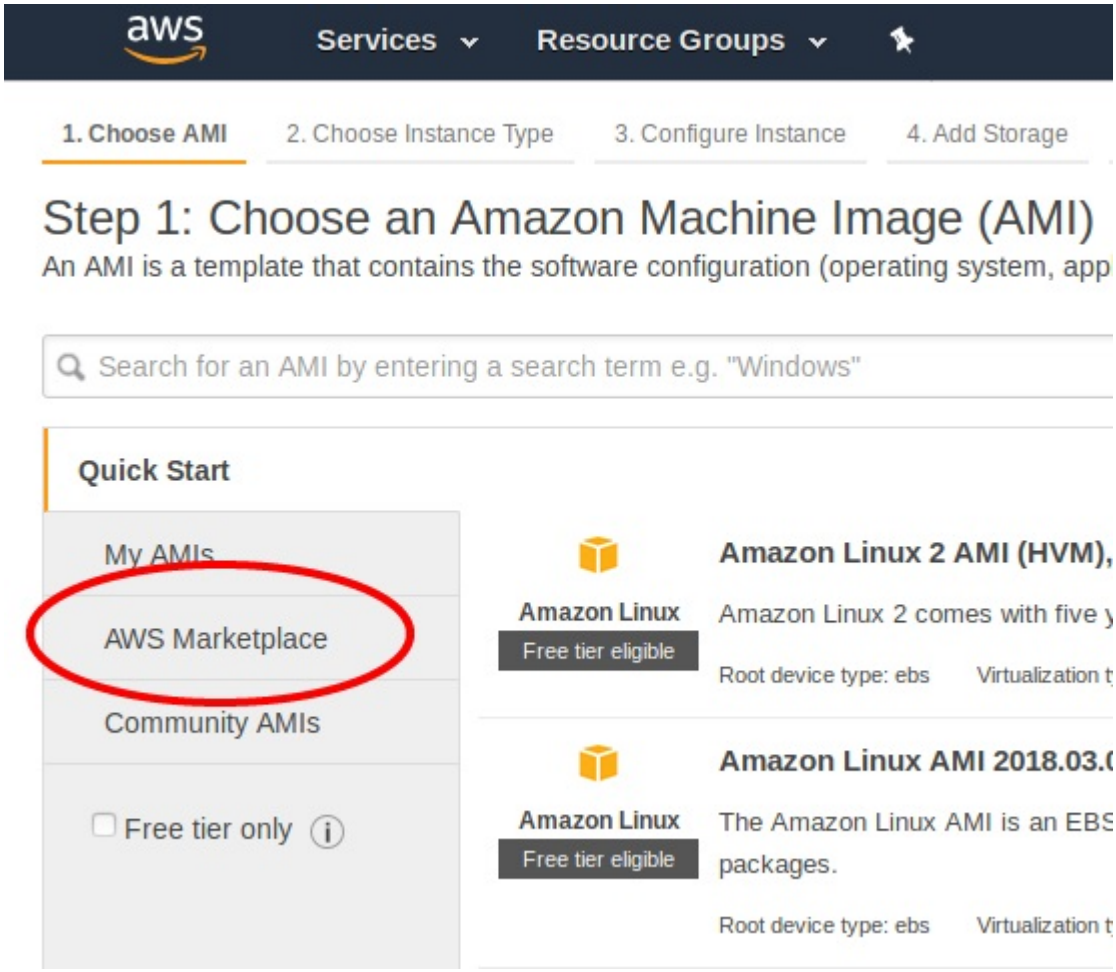
3. Launch a new instance by clicking on the **Launch Instance** button under the **Resources** section of the EC2 dashboard.



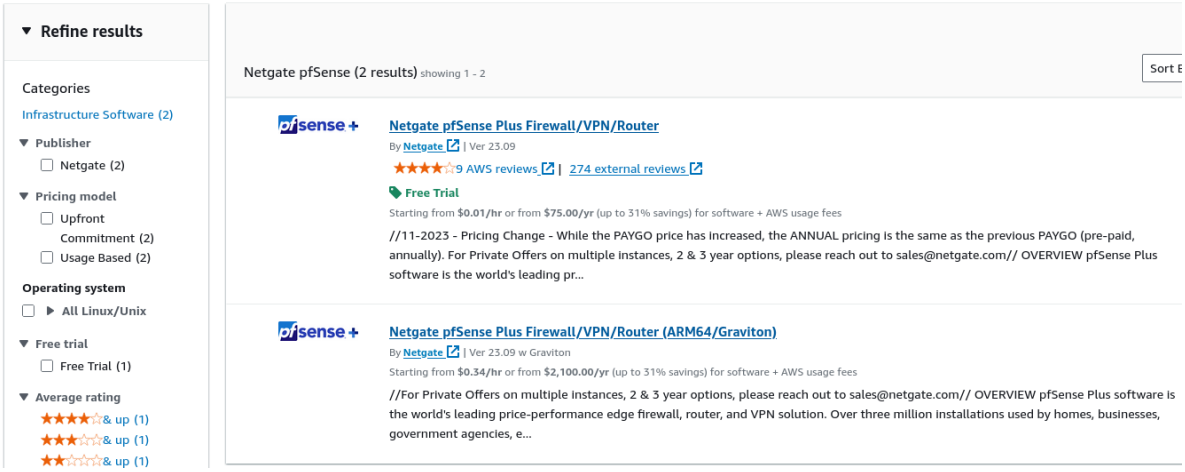
4. Name the instance something like `pfSense` and under the **Application and OS Images**, choose **Browse more AMIs**.



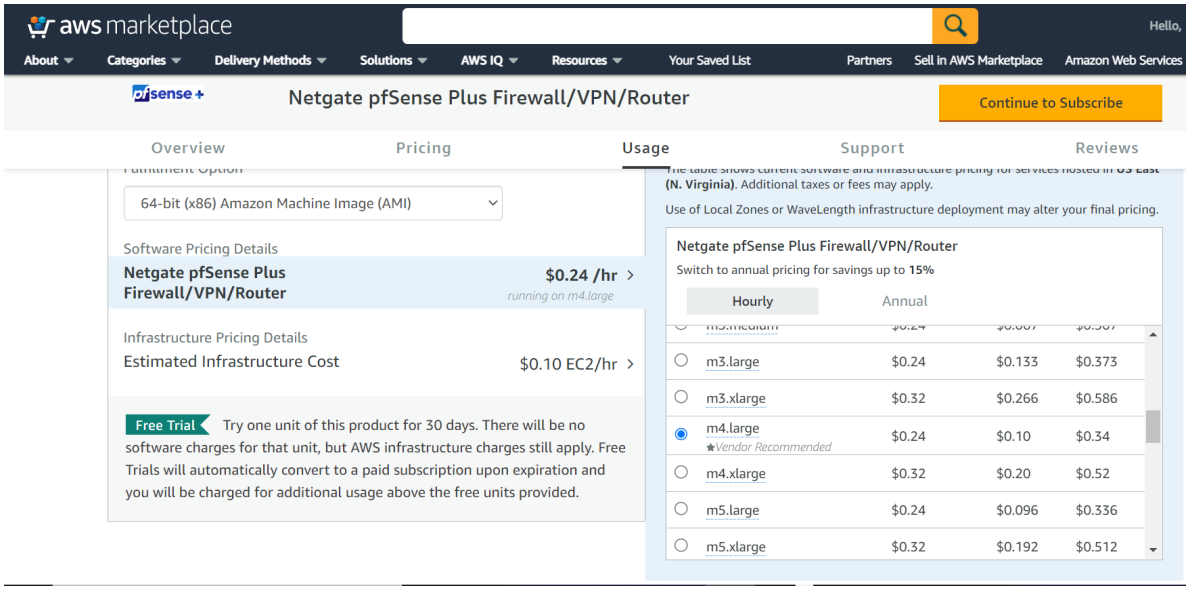
5. Type `Netgate pfSense` in the search box and press `Enter`.



6. Click the **Select** button for the **Netgate pfSense Plus Firewall/VPN/Router** listing in the search result that corresponds to the desired type of instance. This could be either the amd64 AWS product or the arm64/Graviton AWS product depending on the needs of this deployment.

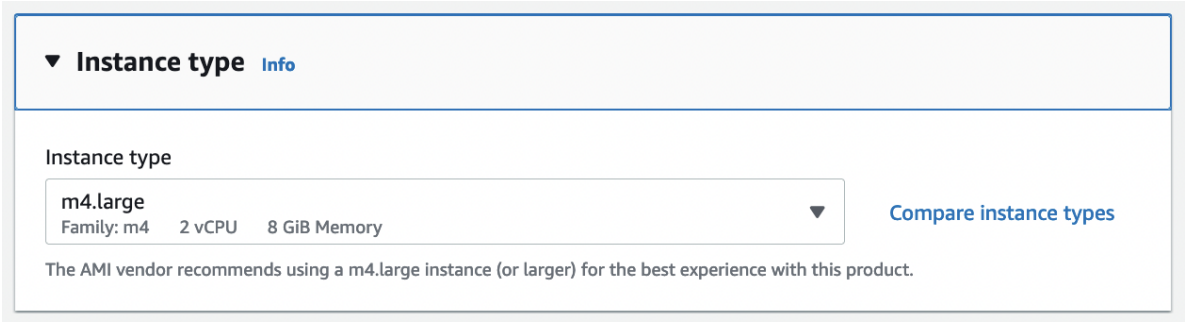


7. Review pricing and other helpful information, then click **Continue**.



Note: There are no optional billable services for the pfSense Plus software. Information about Netgate TAC can be found on the *Assistance Resources* page.

8. Select the desired instance type for the pfSense Plus software from the drop-down.



9. Choose the desired Network and Subnet that the instance will be deployed in. Choose any other instance-specific settings that may be required in the environment. Optionally expand the **Advanced Details** section and set parameters as text in the **User Data** field. The available options are:

password
Setting a value via a directive like `password=abcdefg` will set the password for the administrative account to the given value – `abcdefg` in this example. If no value is set here, a random password will be assigned in order to keep administrative access from being exposed to the Internet with a default password.

Note: A password configured using this method cannot contain the characters `:` or `=`, which are reserved for use as delimiters by the script which handles importing these values.

mgmtnet
Setting a value via a directive like `mgmtnet=10.0.1.0/24` will restrict management access (HTTP, HTTPS, SSH) to the given network – `10.0.1.0/24` in this example. This will cause the firewall rules on the instance (not on access lists in AWS, but on the Netgate pfSense® Plus

appliance firewall rules) to restrict management traffic for the instance to the specified source network. The default behavior is to allow management from any host.

These directives can be set by placing them on a single line in the User Data field and separating them with colons. To specify both parameters, type a statement similar to this one:

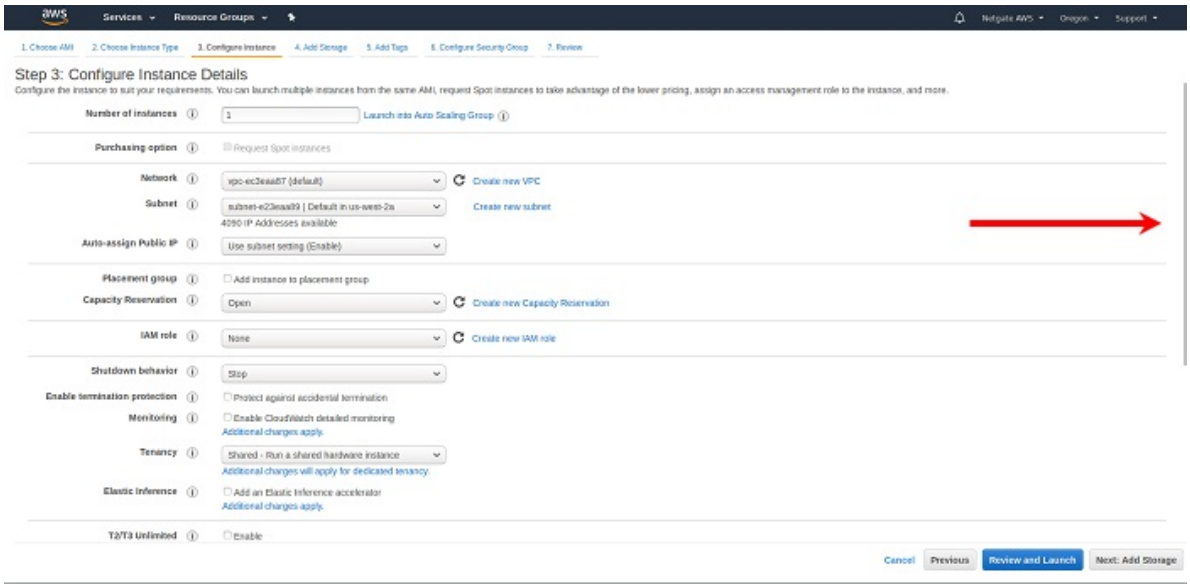
```
password=abcdefg:mgmtnet=10.0.1.0/24
```

Note: If a password is set using the password parameter listed above, the password is retrieved by the instance via an unencrypted HTTP request when the system is configured the first time it boots. The request is made to an Amazon Web Services-operated server on the local LAN that stores metadata about each instance running. The data for an instance is only made available to that instance, but is available to be queried from the instance without providing any authentication credentials.

The best practice is to change the admin password via the pfSense® Plus GUI after the instance comes up to avoid any security risks associated with the unencrypted request. Otherwise, it is possible to choose not to set the password at all and let a random password be set.

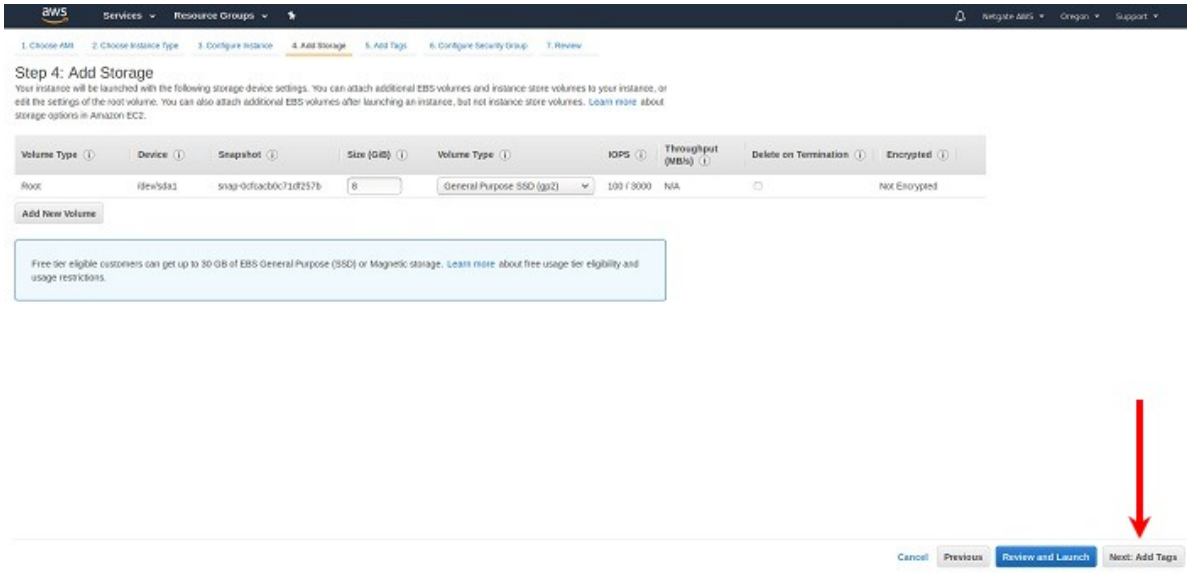
---

10. Choose the desired Network and Subnet to which the Instance will be deployed. Scroll down to configure the network interface(s) with a Static or DHCP-assigned IP address.



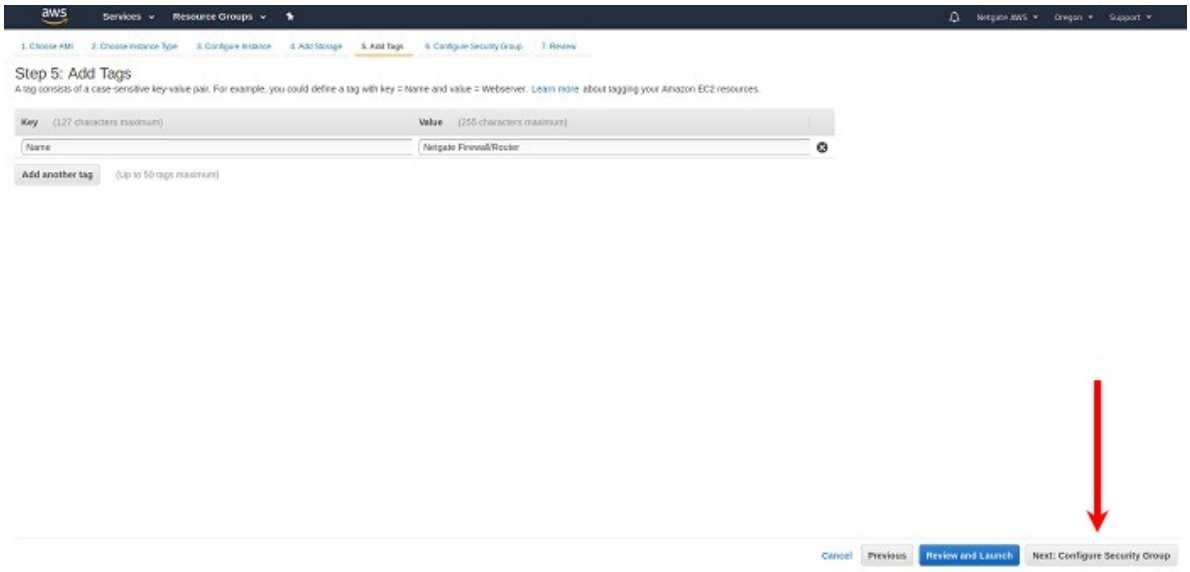
Once the Network Interface(s) are configured, select **Next: Add Storage**.

11. Click **Next: Add Tags** to accept the Storage Device Configuration.



12. The best practice is to set a tag that can be used to differentiate this instance from other instances by entering a value for the **Name** tag. Click **Next: Configure Security Group** after setting any desired tags.

    Press the **Add Tag** button. Input `Name` under the **Key** field and the desired Instance Tag Name under the **Value** field (e.g. `Netgate Firewall/Router`).



13. Select a security group to launch the instance with. The **Security group name** and **Description** fields can be left at the default, or replaced with the desired values.

    The security group should allow at least the following traffic to start with:

    - TCP port `443` from `0.0.0.0/0`

      HTTPS - This is the port that the management GUI listens on.

    - TCP port `22` from `0.0.0.0/0`

      SSH - This port can be used to connect to a command prompt with an ssh client.

    - UDP port `1194` from `0.0.0.0/0`

OpenVPN - The OpenVPN server that is configured by default is bound to this port.

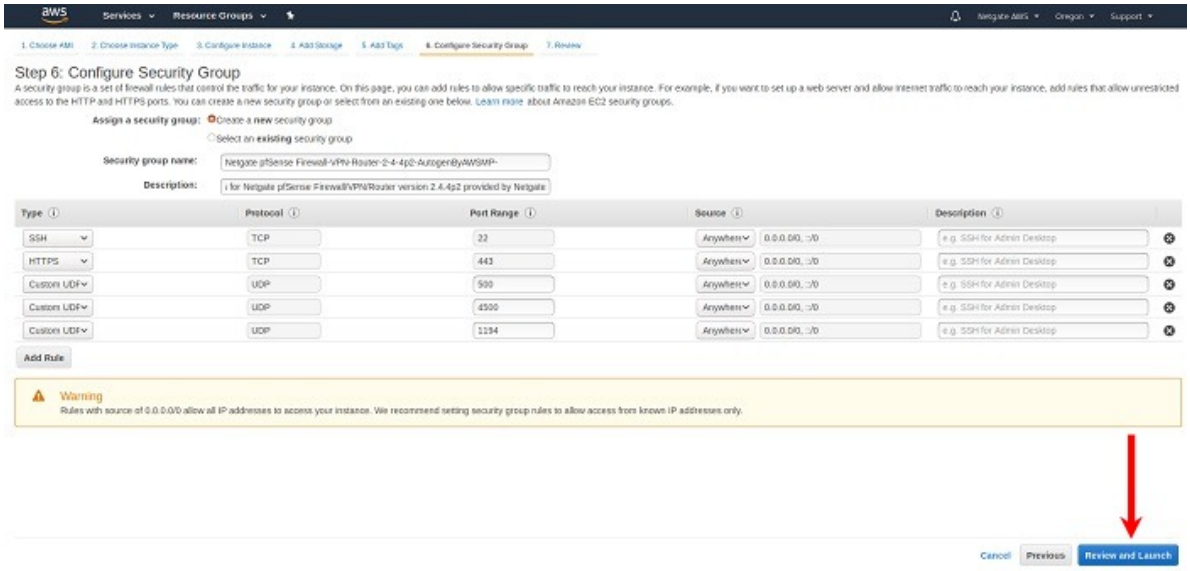
- UDP port `500` from `0.0.0.0/0`

  IKE for IPsec VPN.

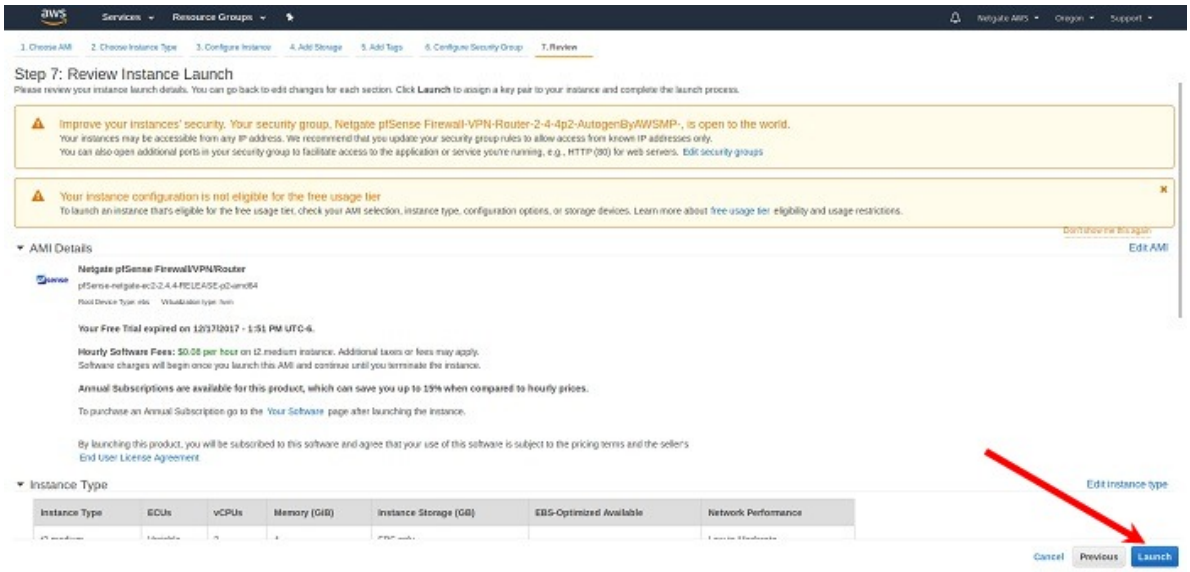
- UDP port `4500` from `0.0.0.0/0`

  IPsec/NAT-T for IPsec VPN.

Note: If there is an existing security group that includes this access, click **Select an existing security group**, then select the desired group(s) to use and click **Continue**. Otherwise, select **Create a new security group**, and add rules for this access by filling in the form for each rule and clicking the **Add Rule** button. When all the rules have been added, click **Review and Launch**.

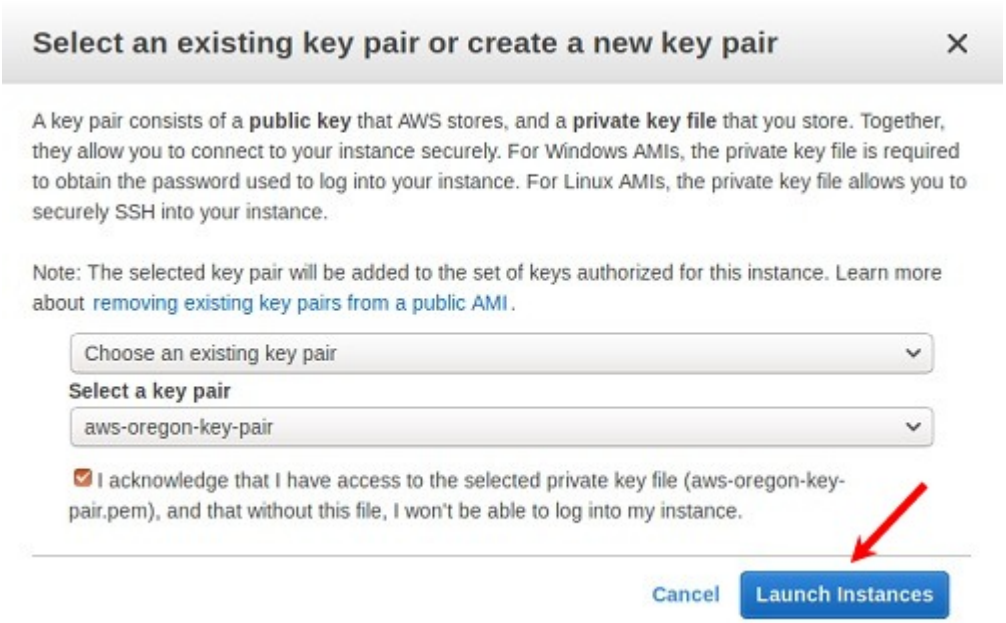


14. Review any AWS warnings and make note of recommendations. Scroll down to review the remaining instance details and click **Launch** after making any needed adjustments.



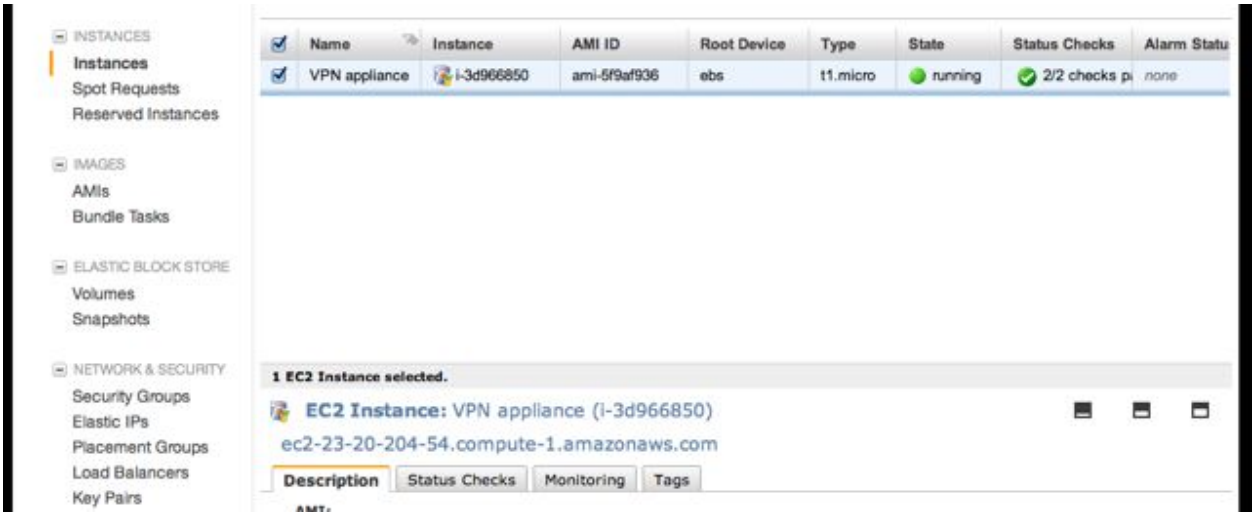
15. Select an existing key pair or create a new key pair to connect to the instance with. Click the checkbox that indicates acknowledgment of access to the selected private key file and then click **Launch Instances**.

Important: Do NOT select the **Proceed Without a Key Pair** option.



2.2 Managing the Configuration of the Instance

Once the instance is launched, monitor its status using the Instances page of the EC2 Management Console. The EC2 Management Console will display whether the instance is up and reachable and will also display its current public IP address and the hostname that resolves to the public IP address. Find the hostname and public IP address in the EC2 console by clicking on the **Instances** heading on the left, finding the instance and checking the checkbox next to it and looking at the details at the bottom of the page.



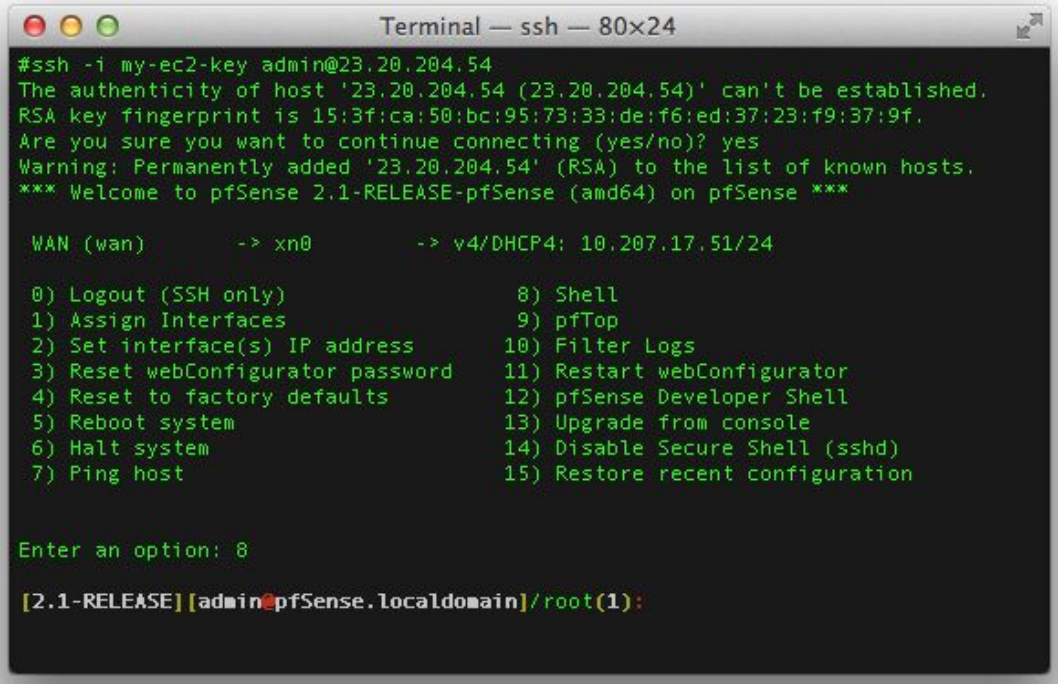
In the example above, the hostname of the instance is `ec2-23-20-204-54.compute-1.amazonaws.com`. The public

IP address is available by putting together the 4 numbers included in the hostname – `23.30.204.54`. It is also possible to obtain the IP address by using a popular DNS lookup tool such as `host`, `dig`, or `nslookup` to resolve the hostname to its IP address.

Note: The hostname and IP address used in this and other examples in this guide are associated at the time of writing with a test instance. This address/hostname will not be the same values used to access the instance, and they will not even be associated with the same test instance by the time this guide is available to the public.

In order to manage the configuration of the instance, connect to it via HTTPS or SSH. To connect via SSH, use the key pair chosen while creating the instance to connect to the `admin` account. From the command line on a Unix/Linux host, use a command similar to `ssh -i my_key_file admin@public_IP`, where the appropriate private key file and public IP address or hostname are substituted. In the example below, the client uses the key file `my_ec2_key` connect to the IP address `23.20.204.54`.

Note: The first time logging into the instance, the SSH key for the instance will not be cached locally, type `yes` when asked whether to continue connecting. This should not be necessary on subsequent sessions.



A limited set of configurations is possible through SSH. The preferred method for managing most of the configurations or viewing data on the status of the Netgate® pfSense® Plus instance is through the HTTPS GUI. To connect via HTTPS, enter an `https://` URL containing the public IP address or hostname of the instance into a web browser. For example, `https://23.20.204.54`.

There will likely be a browser warning indicating that the security certificate of the site is not trusted, because the instance uses a self-signed certificate for HTTPS communication. Click on the option to proceed to the site anyway

and a login screen with the Netgate logo should appear.

The username to log in with is `admin`. The password to use is either a value set in the **User Data** during the creation of the instance or a random password. If a specific password was not set, The value of the random password can be found through one of two different means:

1. Log in over SSH with the key pair selected when the instance was created and examine the contents of the file located at `/etc/motd-passwd`. Select option 8 (**Shell**) from the console menu that is presented after log in and execute `cat /etc/motd-passwd` from the shell.
2. Alternatively, view the **System Log** for the instance in the EC2 Management Console. After the messages that are displayed that show the status of the boot process, a message should appear that indicates the value of the administrative password.

Note: The **System Log** output in the EC2 Management Console is not updated in real time and may take a few minutes to show up. It is preferable to explicitly set a password by passing a value in with the **User Data** field, so the password will be known in advance. To allow a random password to be set, connect via SSH and find the value of the password after the instance is up without any delay.

The message, using either of the methods mentioned, will look like this

```
***
***
*** Admin password changed to: abcdefg
***
***
```

In this example, the password was changed to `abcdefg`.

Once the password has been determined and entered into the login form, the pfSense® Plus GUI should be available.

2.3 Using the remote access IPsec VPN

An IPsec VPN for remote users is preconfigured on the instance when it comes up. Configure the IPsec VPN on client devices to utilize this VPN.

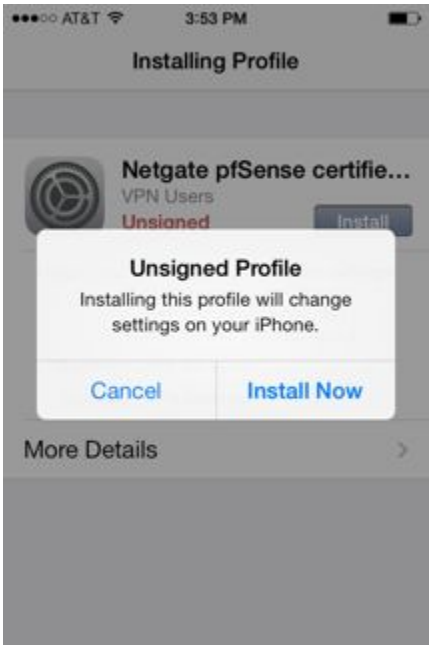
See also:

A guide for manually configuring Android or iOS (iPhone/iPod/iPad) mobile clients to establish an IPsec VPN is located in the pfSense® Documentation.

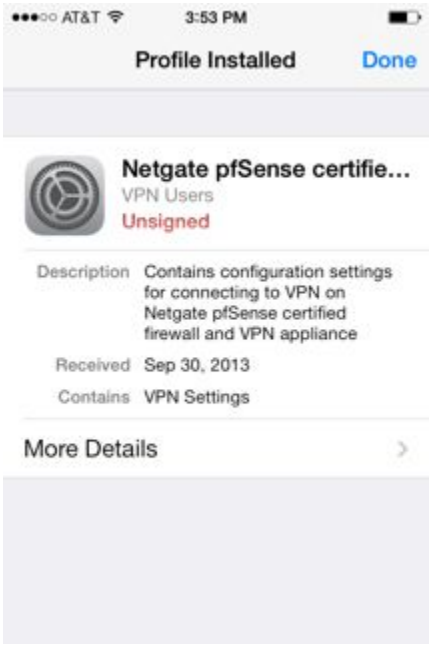
For iOS clients, a profile can be downloaded and installed that will automatically configure an IPsec VPN to the instance. The profile can be downloaded by visiting the page at **VPN > IPsec Export: Apple Profile** or by loading the page directly at `/iphone_ipsec_profile.php` on the instance. If the instance IP address were `23.20.204.54`, the correct URL to visit would be `https://23.20.204.54/iphone_ipsec_profile.php`. Using this page requires authenticating to the web interface by typing the username (`admin`) and password prior to being able to download the profile.

The profile should be downloaded and saved automatically upon opening the page. If the page is visited in a web browser on an iOS device, the device should automatically launch the Settings app and attempt to install the new profile. If the profile is downloaded to another non-iOS device, it can be sent via email as an attachment. If the attachment is opened in the iOS email client, the Settings app new profile installation will also open.

The name and description of the profile being installed will be displayed. Tap the **Install** button. A warning message will be displayed that indicates that the profile is unsigned. Tap on **Install Now** to continue.



Enter the passcode for the iOS device (the one entered when waking the device from sleep) and the password to access the IPsec VPN (the one entered to get access to the GUI) when prompted and the profile will be installed. When the screen shows that the profile was installed, tap **Done**.



When the profile has been installed, the VPN can be enabled in the Settings app. There will be a heading named **VPN** under the main Settings page. If there are more than one VPN configured on the device, tap the VPN heading. The newly installed profile should be selected. It will have a check mark next to it. There will be an on/off switch at the top of the page to enable the VPN. If this is the only VPN configured, the switch to enable the VPN will be next to the VPN heading on the main Settings page. Tap the switch to enable the VPN. The client will prompt for a username and password. The username (`admin`) should already be filled in. Enter the password and tap **OK**. A welcome message should be displayed. Tap **OK** and the VPN is ready to use.

2.4 Using the remote access OpenVPN VPN

An OpenVPN VPN for remote users is automatically configured the first time the instance is booted. To use the VPN, install an OpenVPN client app on a device and import a configuration that specifies how to connect to the instance.

An OpenVPN configuration can be downloaded by visiting the page `/openvpn_connect_profile.php` on the instance. If the instance IP address were `23.30.204.54`, the correct URL to visit would be `https://23.20.204.54/openvpn_connect_profile.php`. Authenticate to the web interface by typing the username (`admin`) and password prior to being allowed to download the configuration.

The profile should be downloaded and saved automatically upon opening the page. The file that it was saved in should be imported into the OpenVPN client on the client device.

2.4.1 Tips for configuring OpenVPN based on platform/client

OpenVPN Connect App on iOS (iPhone/iPad/iPod)

The iOS version of the OpenVPN Connect App allows importing an OpenVPN profile by opening an attachment to an email message. Save the config to a file named `remote-access-vpn.ovpn` and send it to an email account that the iOS device is configured to retrieve mail for. Open the email message and touch the attachment to open it. The device will present **Open in OpenVPN** as one of the available options. Touch the OpenVPN icon to select that option. The OpenVPN Connect App should then open and list the profile under a heading that says **New profiles are available...**. Click on the green ball with the + sign in it to import the profile. Type in the username, `admin`, and password then change the On/Off switch to **On**.

OpenVPN Connect App on Android

The Android version of the OpenVPN Connect App allows importing an OpenVPN profile from an SD card. Save the configuration file to the SD card. Launch the OpenVPN Connect App. From the menu, select Import, then Import Profile from SD card. Browse to the location of the configuration file and select it. Enter the username, `admin`, and password to connect to the VPN. Press **Connect**.

Tunnelblick on macOS

The Tunnelblick App for macOS allows importing an OpenVPN configuration file. Save the configuration to a file on the client device. Click on **VPN Details**. Click on the **+** symbol underneath the existing configurations to add a new configuration. Click on the **I have configuration files** button. Click on the **OpenVPN Configuration(s)** button. Follow the instructions presented by Tunnelblick (copy the config into an empty folder Tunnelblick creates on the Desktop, rename the folder, click on the folder). When the profile is imported successfully, click on its name and then click on **Connect**. Enter the username, `admin`, and password to connect to the VPN.

OpenVPN Connect Client on Windows

The OpenVPN Connect Client on Windows allows importing an OpenVPN configuration file from the local disk. Save the file on the client device. Click the **+** symbol to the right of **Connection Profiles**. Select **Local File** and click on the **Import** button. Find the profile to import in the file browser window and click **Open**. A box with the name of the new profile should appear under **Connection Profiles** now. Click on that box and enter the username, `admin`, and password to connect to the VPN.

2.5 AWS High Availability

The AWS High Availability package enables the use of active/standby pairs of pfSense® Plus instances which modify AWS resources in response to failover events.

The AWS High Availability package builds upon the CARP Virtual IP Address (VIP) functionality in pfSense software to provide an analogous mechanism for High Availability (HA) in the AWS Virtual Private Cloud.

The need for the AWS High Availability package stems from the unique environment of a VPC. These environments do not have a traditional layer 2 network, and thus do not carry broadcast and multicast traffic. This limitation necessitates using the AWS API for configuration of VPC resources to define routes, bind addresses, and make external IP address mappings – actions that CARP cannot do natively due to the nature of the AWS environment.

See also:

- pfSense Software High Availability Documentation

- pfSense Plus Software High Availability Example

2.5.1 How it Works

When the AWS High Availability package is installed, it extends CARP VIP MASTER events on configured CARP VIPs such that user configurable modifications to VPC resources can be made without manual intervention.

An instance be configured to modify Elastic Interface IP address assignments, AWS VPC Route Tables, and Elastic IP Allocations in response to a CARP VIP event. This allows a failover event to modify routes and map subnet IPs and Elastic IPs to the Elastic Network Adapter on the node where a CARP VIP has MASTER status.

The package performs these actions by making contact with the AWS API to make changes dynamically as needed.

2.5.2 Prerequisites

AWS Account Privileges

The package requires access to the AWS API, relying on a correctly configured **EC2 Instance Profile** to assign a **Role** to the instance for authorization. The **Role** must have the following privileges for the associated resources:

`DescribeRegions`

- All resources ("*")

`DescribeRoutetables`

- All resources ("*")

`AssignPrivateIpAddresses`

- All Elastic Network Interfaces which will be configured with IPv4 IPs

`ReplaceRoute`

- All Route Tables which will have routes modified in response to CARP events

`AssociateAddress`

- All Elastic IP Allocations which will be modified in response to CARP events
- All VPC subnets with which the Elastic IP is associated
- All Elastic Network Interfaces with which the Elastic IP will be associated
- All EC2 instances to which the Elastic Network Interfaces are assigned

`CreateTags`

- All Elastic Network Interfaces which will be configured with IPv4 IPs
- All Elastic IP Allocations which will be modified in response to CARP events
- All Route Tables which will have routes modified in response to CARP events

`DescribeTags`

- All resources ("*")

pfSense Plus Software Configuration

Before configuring the AWS High Availability package, the following areas must be configured on pfSense Plus Software.

High Availability Synchronization

State Synchronization may be configured on both primary and secondary instances with **pfsync Synchronize Peer IP** values defined pointing to the other node. When configured in this way, pfsync utilizes unicast instead of the default directed multicast, which does not work on AWS VPC.

CARP Virtual IP Addresses

All AWS High Availability configuration requires at least one available unicast CARP VIP configured on an interface. Any number of IP, Route, and Elastic IP actions may be configured for a given CARP VHID.

Note: Reminder: Unicast CARP is required as the default multicast communication does not work in AWS VPC.

The configuration for a unicast CARP VIP is nearly identical to a traditional multicast CARP VIP. The difference is:

- The **CARP Mode** must be set to *Unicast*
- The **Peer Address** field to the right of the **Unicast** option must be set to the actual interface IP address on the HA peer node.

For example, with a CARP VIP for the WAN interface, the peer address on the VIP on the primary node is set to the WAN interface IP address of the secondary node. If configuration synchronization (XMLRPC Sync) is enabled, the primary node will automatically adjust this value when copying the VIP to the secondary node so that the VIP on the secondary uses the WAN interface IP address of the primary for its peer address.

In configurations where a cluster is synchronizing across availability zones, there is no common address to all subnets, so a dummy address must be chosen for the CARP VIP. In these instances, choose an address from the RFC 5737 documentation networks (`192.0.2.0/24`, `192.51.100.0/24` or `203.0.113.0/24`). The purpose of the CARP VIP is entirely for failover negotiation, and should not be used as a source or destination address.

Note: If peers are in different subnets, unicast CARP traffic will egress through the default route unless another route to the peer has been configured. The route may seem counterintuitive if the selected CARP interface is not the default WAN interface - in which case, absent a route to direct the traffic through a specific gateway, the unicast CARP traffic will route via the WAN interface to the default route.

2.5.3 Package Configuration

To configure the AWS High Availability package, navigate to **System > AWS High Availability**. This displays the **IPs** tab for the package.

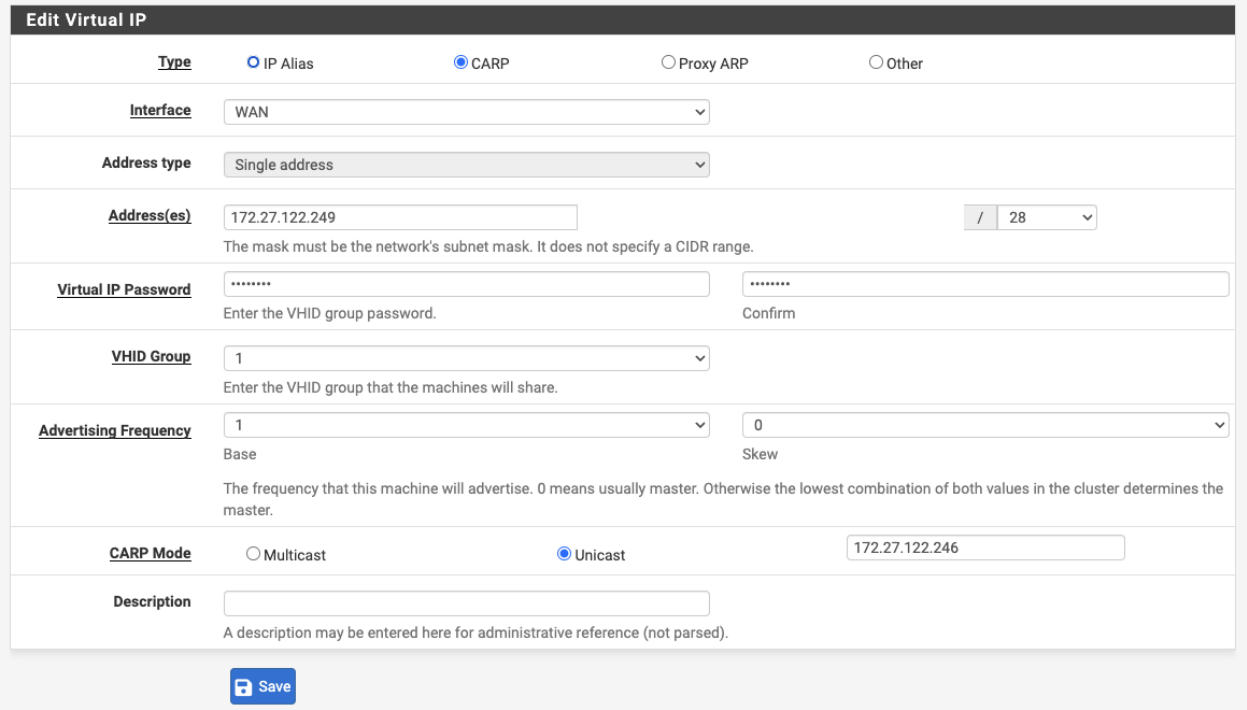


Fig. 1: WAN Interface Unicast CARP Virtual IP Address - Primary

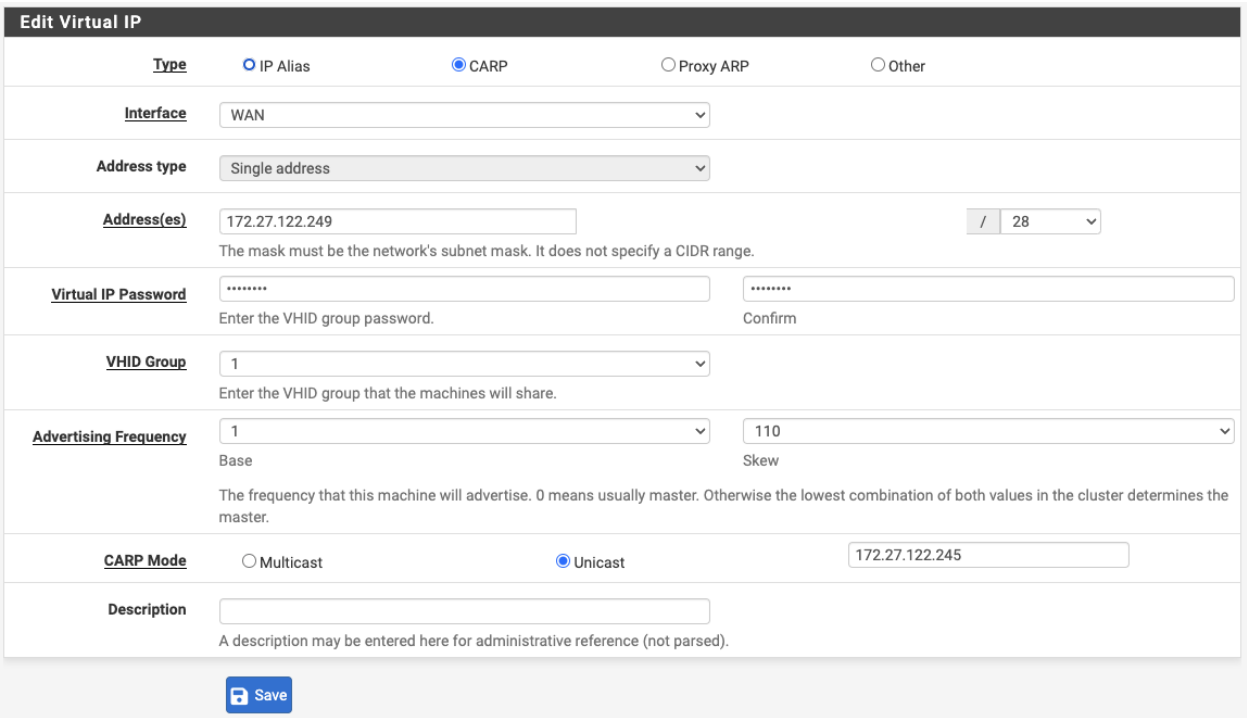


Fig. 2: WAN Interface Unicast CARP Virtual IP Address - Secondary

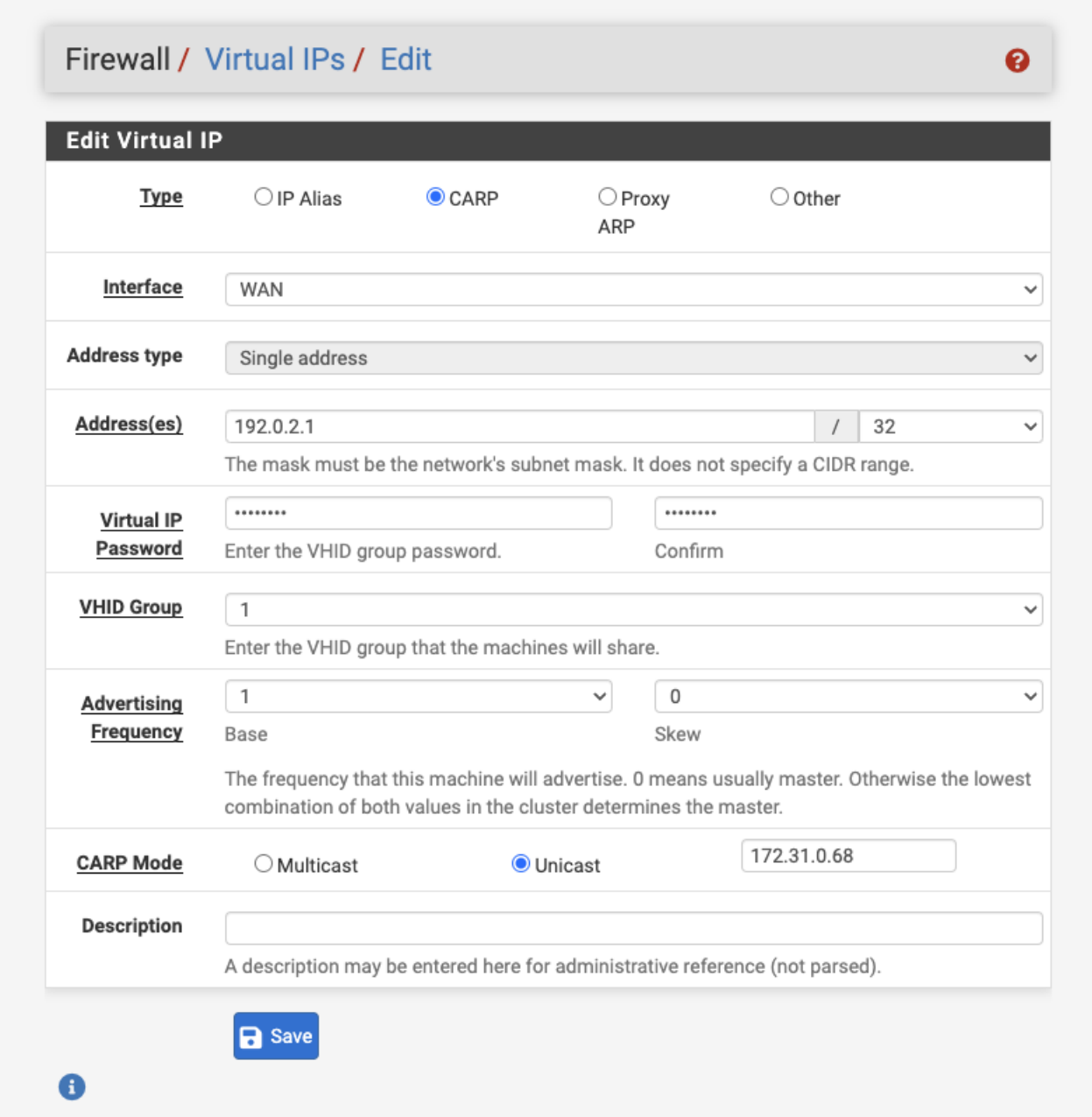


Fig. 3: WAN Interface Unicast CARP Virtual IP Address - Peer in Other AZ

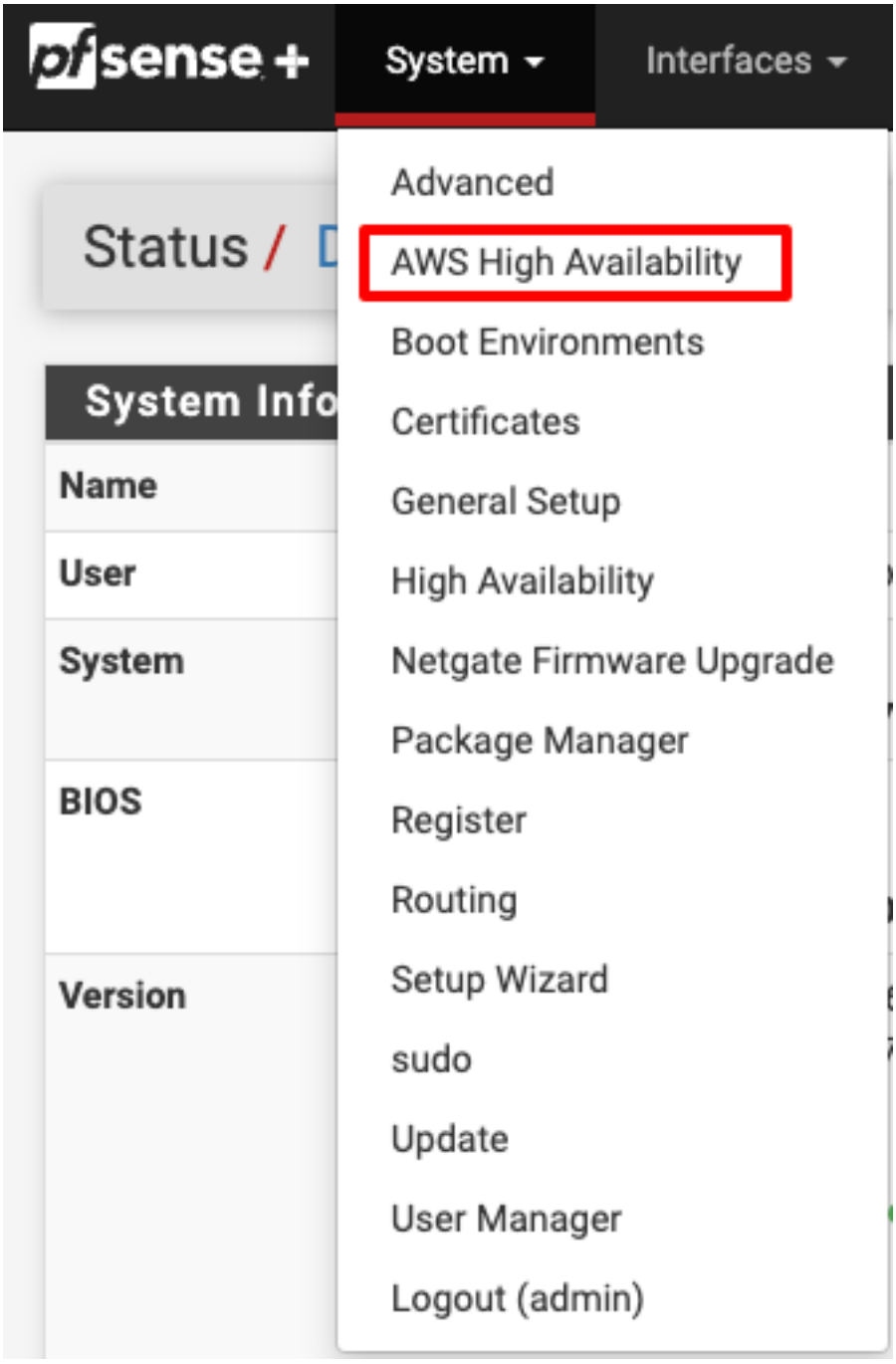


Fig. 4: AWS High Availability System Menu Entry Location

IPs

The **IPs** tab controls how the package updates IP address assignments to Elastic Network Interfaces.

When configuring an IP action in the **IPs** tab, the user is presented with the following options:

CARP VHID
: A drop-down selection to choose the CARP VIP which will trigger this route to be updated when it encounters a failover event and assumes MASTER status. This list is presented with the interface name and VHID (e.g. `wan@1`).

Interface
: A drop-down selection to choose the interface destination of the route. Only Elastic Network Interfaces will be presented.

Private IP Address
: An input field for the IP Address to associate with the Elastic Network Interface. This address must be a valid address in the VPC Subnet associated with the Elastic Network Interface.

XMLRPC Sync
: A checkbox to enable synchronizing this IP action via XMLRPC. The Private IP Address will be mapped to the VPC Subnet of the Elastic Network Interface of the peer automatically, if it differs and the subnet masks are the same.

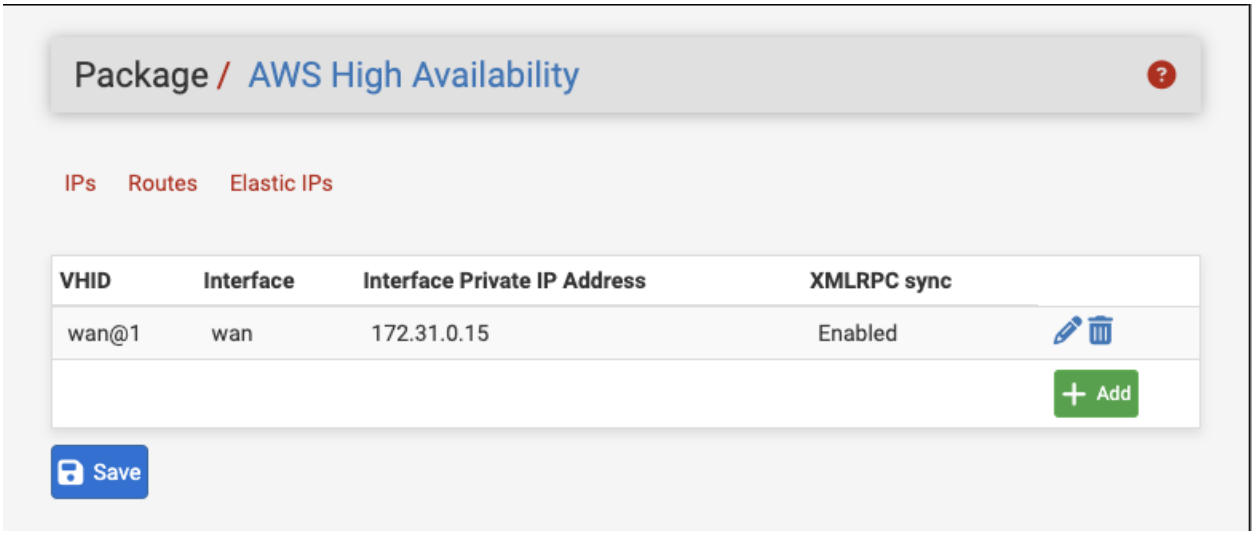


Fig. 5: AWS High Availability IP Address Entry

Routes

The **Routes** tab controls how the package updates route entries in AWS during failover events.

When configuring a route action in the **Routes** tab, the user is presented with the following options:

CARP VHID
: A drop-down selection to choose the CARP VIP which will trigger this route to be updated when it encounters a failover event and assumes MASTER status. This list is presented with the VHID and interface name (e.g. `wan@1`).

Route Table ID
: Input field to specify the AWS resource identifier for the route table to modify (e.g. `rtb-05ed6e7c46531c6c3`).

Route CIDR
An input field for the CIDR of the route destination address or network to be replaced in the routing table (e.g. `0.0.0.0/0` for the default route).

Interface
A drop-down selection to choose the interface destination of the route. Only Elastic Network Interfaces will be presented.

XMLRPC Sync
A checkbox to enable synchronizing this Route action via XMLRPC.

All fields are required.

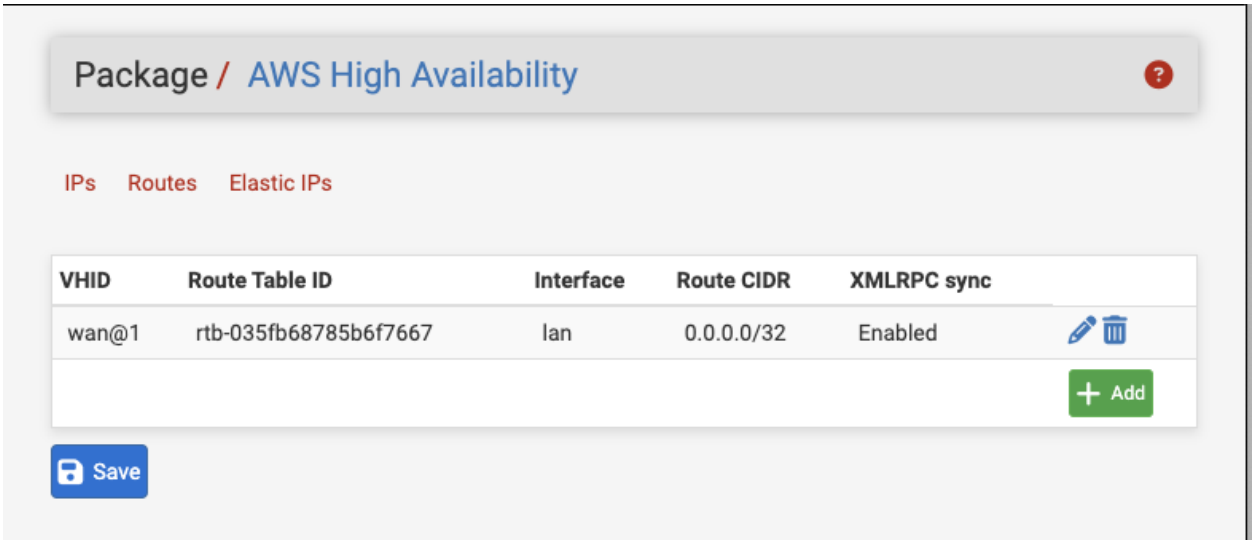


Fig. 6: AWS High Availability Route Entry

Elastic IP Addresses

The **Elastic IPs** tab associates a CARP VIP with an Elastic IP address and interface.

When configuring an Elastic IP action, the user is presented with the following options:

CARP VHID
A drop-down selection to choose the CARP VIP which will trigger this Elastic IP address to be updated when it encounters a failover event and assumes MASTER status. This list is presented with the VHID and interface name (e.g. `wan@1`).

Elastic IP Allocation ID
An input field for the Elastic IP Allocation ID that identifies the Elastic IP Allocation to modify. This value must be filled in by the user.

Interface
A drop-down selection to choose the interface destination of the route. Only Elastic Network Interfaces will be presented.

Private IP Address
A drop-down selection to choose the Private IP Address of the interface to which the Elastic IP address will point. Private IP Addresses presented will include addresses that are known to be associated with the Elastic Network Interface and addresses configured on the IP tab for the interface.

This value is populated automatically with the IP address of the selected CARP VIP.

XMLRPC Sync
A checkbox to enable synchronizing this Elastic IP action via XMLRPC. The Private IP Address will be mapped to the VPC Subnet of the Elastic Network Interface of the peer automatically, if it differs and the subnet masks are the same.

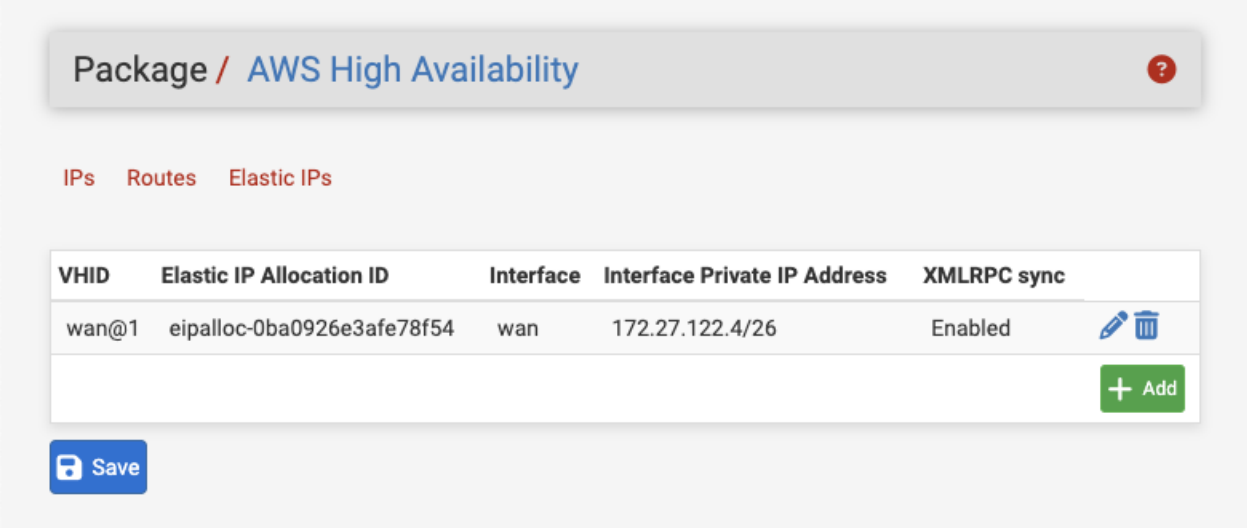


Fig. 7: AWS High Availability Elastic IP Address Options Tab

2.5.4 Behavior

When an instance assumes the MASTER status for a CARP VHID, the package takes the following actions in order:

- Assigns all VIPs for that VHID to the Elastic Network Interface associated with the VHID.
- Performs all configured route replacements configured for the VHID sequentially.
- Performs all configured Elastic IP re-associations configured for the VHID sequentially.

When an action is completed on a resource, it is also updated with tags to record the time of modification and the CARP advertising frequency base and skew of the member that acted upon it.

If the package fails to complete any of these tasks via AWS API calls, it generates a system notice.

2.5.5 Background Service

AWS High Availability installs a small background service in the form of a script scheduled to run by `minicron` every 60 seconds. This script cycles through the configured actions on the system and checks that a lower priority CARP member has not modified the resources, indicating that the CARP link is not configured correctly. If this is found to be the case, a notification is generated and the resource actions are executed again to correct it.

2.5.6 EC2 Service Endpoints

AWS High Availability uses the AWS PHP SDK to access AWS API regional endpoints. Because these endpoints resolve to addresses outside a VPC subnet, without additional action each pfSense® Plus instance will require a valid route through an AWS Internet Gateway or NAT Gateway that remains intact across resource actions.

AWS PrivateLink allows you to map AWS API endpoints to addresses in your VPC subnets, which avoids this scenario and keeps API endpoint traffic local to your VPC. Follow the guides at https://docs.aws.amazon.com/vpc/latest/privatelink/create-interface-endpoint.html and https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/interface-vpc-endpoints.html to create VPC Service Endpoints for EC2 in a VPC Subnet.

Note: When configuring AWS High Availability across availability zones, a subnet in each availability zone will require its own EC2 VPC Service Endpoint.

2.5.7 Limitations

Due to limitations in AWS, established TCP streams that are traversing through pfSense® Plus instances must be re-established after failover. AWS is not currently capable of redirecting traffic for established streams in response to changes to Route Tables, Elastic Network Interfaces, and Elastic IP Addresses.

2.6 Advanced Usage

2.6.1 Protecting a private network in VPC

An instance of the Netgate® pfSense® Plus appliance can be used as a firewall for a VPC subnet. This will generally require more manual configuration than using an instance to host a remote access VPN does. See the *VPC User Guide* for a more detailed explanation of how to configure a VPC and a Netgate pfSense® Plus appliance instance to implement this.

2.6.2 Connecting a local Netgate device running pfSense® Plus software

In addition to connecting remote devices as clients, a device running pfSense® Plus software as a firewall/router can be connected as a peer to a Netgate® appliance.

See also:

Read Configuring a Site-to-Site Static Key OpenVPN Instance in the pfSense software documentation to see the process of configuring this setup.

When implementing the configuration changes detailed in the document, the best practice is to use the Netgate appliance instance on AWS as the "server" end of the connection and the local Netgate device as the client "end". Additionally, make sure that the server is using a unique port. The default remote access OpenVPN server is configured to use UDP port `1194`. When adding a site-to-site tunnel, the best practice is to use a port between `1195` and `2000`. Whichever port the site-to-site tunnel uses will need to be opened both in the firewall rules on the Netgate appliance instance and in the Security Group in the EC2 Management Console.

To route all traffic from a home/office network through the OpenVPN tunnel to the Netgate appliance instance, add this statement to the advanced options for the OpenVPN Client on the home/office Netgate device:

```
redirect-gateway def1;
```

This will cause a default route to be set that sends all locally originated traffic from the home/office network over the OpenVPN tunnel when it is established. When using this configuration to send all traffic from a local network through the OpenVPN tunnel, the outgoing traffic also needs NAT applied on the Netgate appliance instance on AWS for traffic from the home/office network to the internet. This can be accomplished by adding the CIDR block for the home/office network to the preconfigured Alias called `Networks_to_NAT`. This is done by navigating to **Firewall > Aliases** in the GUI, then clicking on the edit icon to the right of `Networks_to_NAT`. Add the new network address and mask to the list of Networks and click the **Save** button. Then click the **Apply Changes** button. Add the network used for the tunnel endpoints (**IPv4 Tunnel Network**) to the `Networks_to_NAT` alias as well using the same procedure that was used to add the home/office network.

Connecting multiple pfSense Plus gateways to a Netgate appliance

Multiple home/office networks can be connected to a single Netgate appliance instance. This could be used to allow clients at different office locations to communicate without requiring tunnels between each individual location. It could also be used as a way to apply policies on traffic to/from the internet in one place and have them take effect across multiple locations.

Each site would need to have the instructions above for connecting an individual device repeated to add an OpenVPN server on the Netgate appliance instance and an OpenVPN client on the local Netgate device. Each OpenVPN Server that is configured must use a unique port and a unique network for **IPv4 Tunnel Network**. It is recommended to use a name that uniquely identifies each location connected in this manner in the Description field when adding an OpenVPN Server for a site in the Netgate appliance.

2.6.3 Detect and Recover EC2 Instance Failure

It is also possible to create an Amazon CloudWatch alarm that monitors an Amazon EC2 instance and automatically recovers the instance if it becomes impaired due to an underlying issue.

For more information about instance recovery, see Recover Your Instance.

2.7 Frequently Asked Questions

2.7.1 How can the GUI password for an instance be located?

The first time the instance boots it looks for a user-defined password set in the **User Data** box when the instance was created. If there is no custom password, it chooses one randomly so that the instance is not accessible via a default password to malicious users.

The random password can be located by choosing **Get System Log** from the **Actions** Menu for the instance in the EC2 Management Console.

A message should appear after the system boot messages that looks like the following:

```
***
***
*** Admin password changed to: abcdefg
***
***
```

It may take 5-10 minutes after the instance boots for this message to appear in the system log. To find out the password sooner, log in via SSH using the SSH key selected when the instance was created. The same message that will be written to the system log will be written to the file `/etc/motd-passwd`. Running the command `cat /etc/motd-passwd` will show the password.

Note: If the output of **Get System Log** is empty or does not contain the expected output, try **Get Instance Screenshot** instead.

2.7.2 How can the random password selected during provisioning be changed?

The password can be changed via the GUI:

- Log in with the username `admin` and the existing random password
- Navigate to **System > User Manager** in the menu
- Locate the `admin` account in the list of accounts
- Click the  icon on the row for the `admin` account to edit the account
- Enter a new secure **Password** in both boxes to confirm the new value
- Click the **Save** button at the bottom of the screen

2.7.3 How can an instance be accessed?

In order to manage the configuration of the instance, connect to it via **HTTPS** or **SSH**. A limited set of configurations is possible through the SSH interface, the preferred method for managing most of the configurations or viewing data on the status of the Netgate® pfSense® Plus instance is through the HTTPS GUI.

Connecting via SSH

Connecting via SSH requires knowing the password of the admin account and logging in with that account or using the SSH key selected when the instance was created. Here is a sample command line to log in with an SSH key from a Unix or Linux host:

```
ssh -i ~/.ssh/my-ec2-key admin@ec2-A-B-C-D.compute-1.amazonaws.com
```

Substitute the actual location of the SSH private key for `~/.ssh/my-ec2-key` and the real hostname, which can be retrieved from the EC2 Management Console by looking at the data for the instance, for example `ec2-A-B-C-D.compute-1.amazonaws.com`.

Note: To log in with a known password for the `admin` account, use a command similar to the one above, but omit the `-i ~/.ssh/my-ec2-key`.

Connecting via HTTPS

Connecting via HTTPS requires the password for the instance, either by setting it explicitly in the **User Data** when the instance is created or by retrieving it from the instance. Connect to the instance with any web browser by typing in the hostname of the instance to the URL field and login using the `admin` account and the password.

2.7.4 How can a VPN client connect to an instance?

See the section in the user guide on Using the remote access *IPsec* or *OpenVPN* VPN.

2.7.5 Why does the GUI Dashboard say the WAN address is 10.X.Y.Z?

Amazon AWS instances use DHCP to assign private addresses to the public-facing interfaces of an instance. Amazon applies NAT between the publicly routable IP address clients use to access the instance and the private address configured on the WAN interface of the instance.

2.7.6 Why do packets not arrive at the firewall even with custom firewall rules in place to allow the traffic?

Amazon AWS provides packet filtering in addition to the Netgate® pfSense® Plus Appliance itself being a stateful firewall. If the Netgate Appliance allows traffic, but there is a security group configuration in the AWS settings for the instance that is restricting traffic, then the security group in the EC2 Management Console must also be configured with rules similar to those on the Netgate Appliance.

Given that the Netgate pfSense® Plus Appliance is a fully functional firewall, it is generally safe to assign an AWS security group which allows all traffic so that the Netgate Appliance can perform any necessary filtering of inbound traffic.

2.7.7 How can a VPN client route all traffic from an entire home network over a VPN?

If a client home gateway/router has an OpenVPN client, it can connect using a site-to-site tunnel between the home network and the Netgate VPN Appliance. The VPN can then route all Internet traffic over the encrypted tunnel. See the user guide section on *Connecting a local Netgate device running pfSense® Plus software*.

This may provide for simpler administration at home, but any mobile devices and laptops that get used outside the home should have an OpenVPN client installed and configured anyway so that they can always receive the benefits of sending traffic through a VPN.

2.7.8 Backup & Recovery

Backing up and restoring the config directions are available in the backup section of the pfSense documentation.

2.7.9 Monitoring

The pfSense Plus software offers a wide range of different monitoring and metrics, see the monitoring section of the pfSense documentation for more information.

2.7.10 Upgrading

Information on upgrading the pfSense Plus software is available in the upgrading section of the pfSense documentation.

2.7.11 Further troubleshooting

More information about troubleshooting pfSense Plus software can be found in the troubleshooting section of the pfSense documentation.

CHAPTER

THREE

VIRTUAL PRIVATE CLOUD (VPC)

3.1 AWS VPC User Guide

The Netgate® pfSense® Plus Firewall/VPN/Router appliance for Amazon EC2 is a stateful firewall and VPN appliance. In addition to its capabilities as a VPN gateway and firewall for users and offices, it is capable of acting as a firewall to protect instances providing services in Amazon's Virtual Private Cloud or VPC service. This service differs from the classic EC2 service in that it allows for management of instances on private subnets.

This guide will explain how to launch, manage, and use an instance of the appliance to act as a gateway for other instances in a VPC subnet.

3.1.1 Preparing a VPC

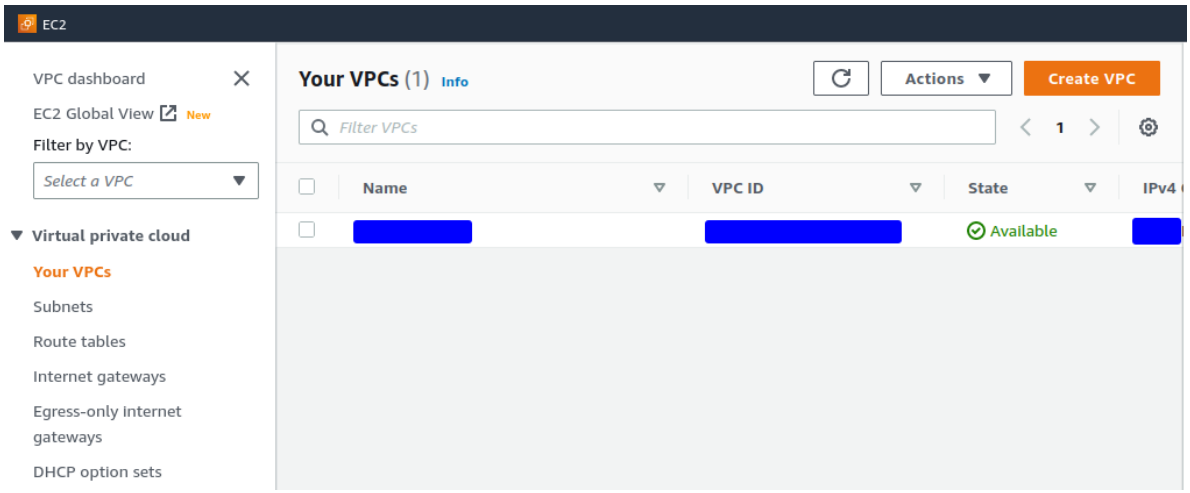
Using a Netgate appliance instance to protect VPC subnets requires the following:

- One internet-facing subnet, to which the Netgate appliance instance will have its primary/WAN interface connected.
- One or more private subnets, to which the Netgate appliance instance will have its secondary/LAN interface (and possibly additional optional interfaces) connected.
- Separate routing tables for the internet-facing subnet and the private subnet(s).
- If all of these are already in place with an existing VPC, feel free to skip ahead to *Launching an Instance*.

These instructions demonstrate how to create a single private subnet and set it up behind an instance of the Netgate® pfSense® Plus Firewall/VPN/Router appliance.

In the Amazon VPC Management Console, create a new VPC, subnets, and routing table(s).

1. Navigate to **Your VPCs**
    - Open the VPC Management Console
    - Click **Your VPCs** in the menu on the left side under the **Virtual private cloud** grouping
    - Click the **Create VPC** button

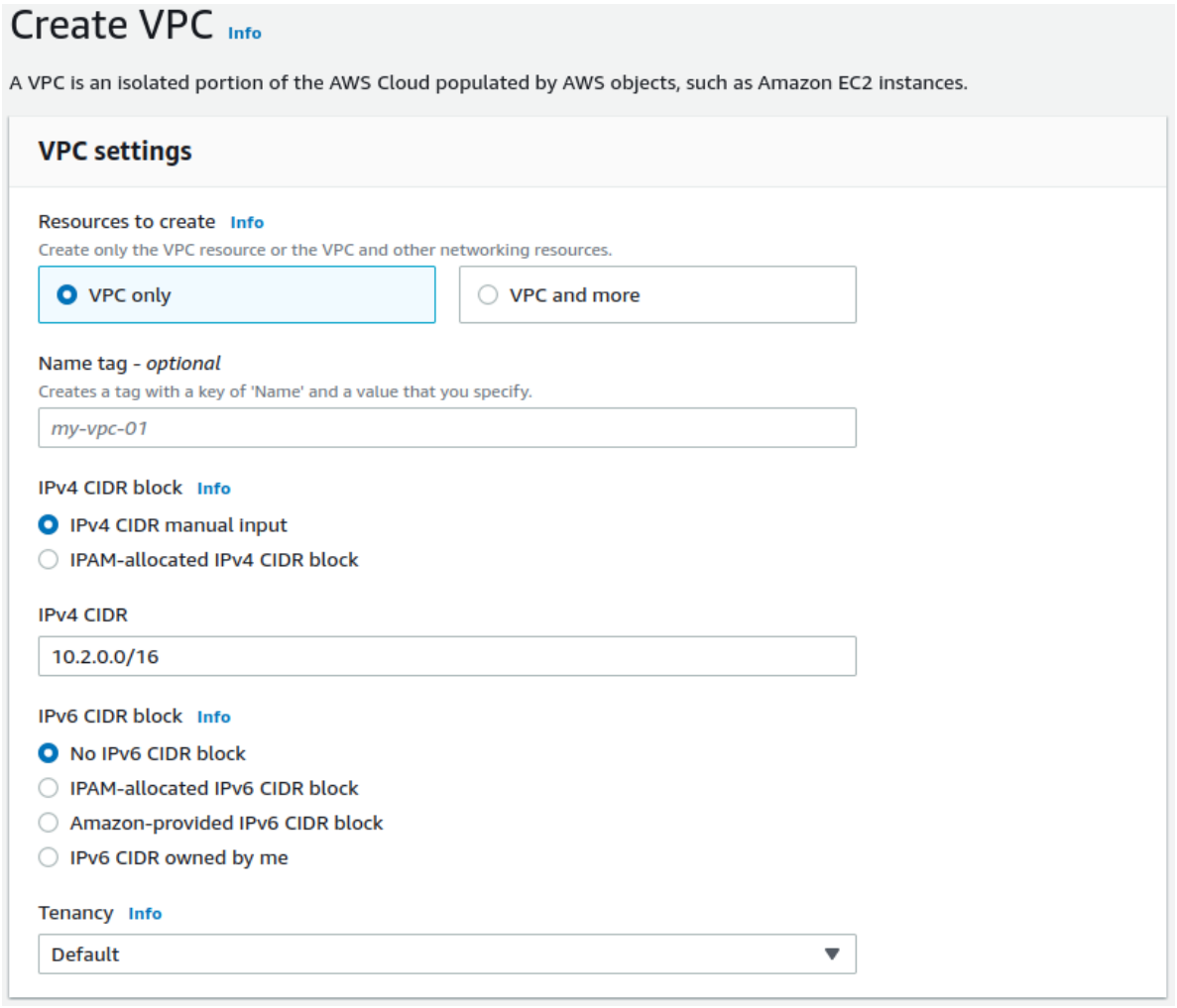


2. Configure the new VPC
   - Optionally enter a **Name tag**
   - Enter the **IPv4 CIDR** network to use

     If connecting to hosts in the VPC using a VPN from hosts at other sites in an organization's infrastructure, be sure to select address space that does not conflict with the private address space used elsewhere by the organization.

     Make sure the block is large enough to contain all subnets to include within it, optionally providing for future expansion. e.g. To use a `/24` for an internet-facing subnet and a `/24` for a private network, the minimum CIDR block must be at least a `/23` to hold those two subnets. The maximum size block is a `/16`.

     For the purposes of this example, use `10.2.0.0/16`.
   - Leave the value of **Tenancy** set to *Default*
   - Click the **Create VPC** button

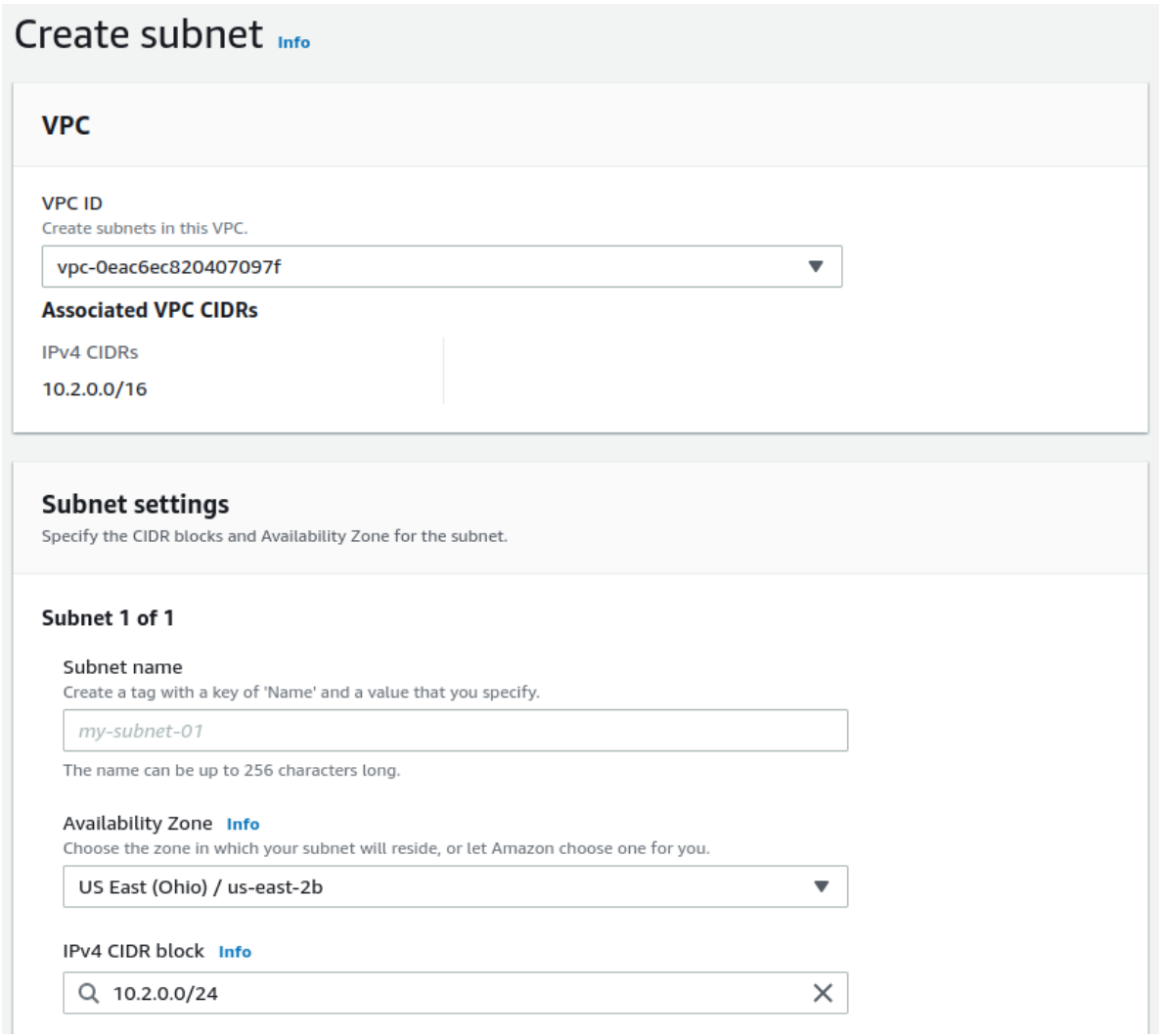


3. Create the public subnet(s)
   - Navigate to the **Subnets** view in the menu on the left side of the VPC Management Console
   - Click the **Create Subnet** button
   - Select the newly created VPC from the **VPC ID** drop-down
   - Optionally enter a **Subnet name**
   - Optionally choose the desired **Availability Zone**
   - Enter the subnet to use for the internet-facing hosts in the **IPv4 CIDR Block** field

     This subnet **must** be a block that is within the address space assigned to the VPC.

     This is the subnet to which the WAN interface of the Netgate appliance instance is attached and could include any other hosts or appliances that should be available directly from the Internet and not protected behind the Netgate appliance.

     For this example, use `10.2.0.0/24`.
   - Click the **Create subnet** button



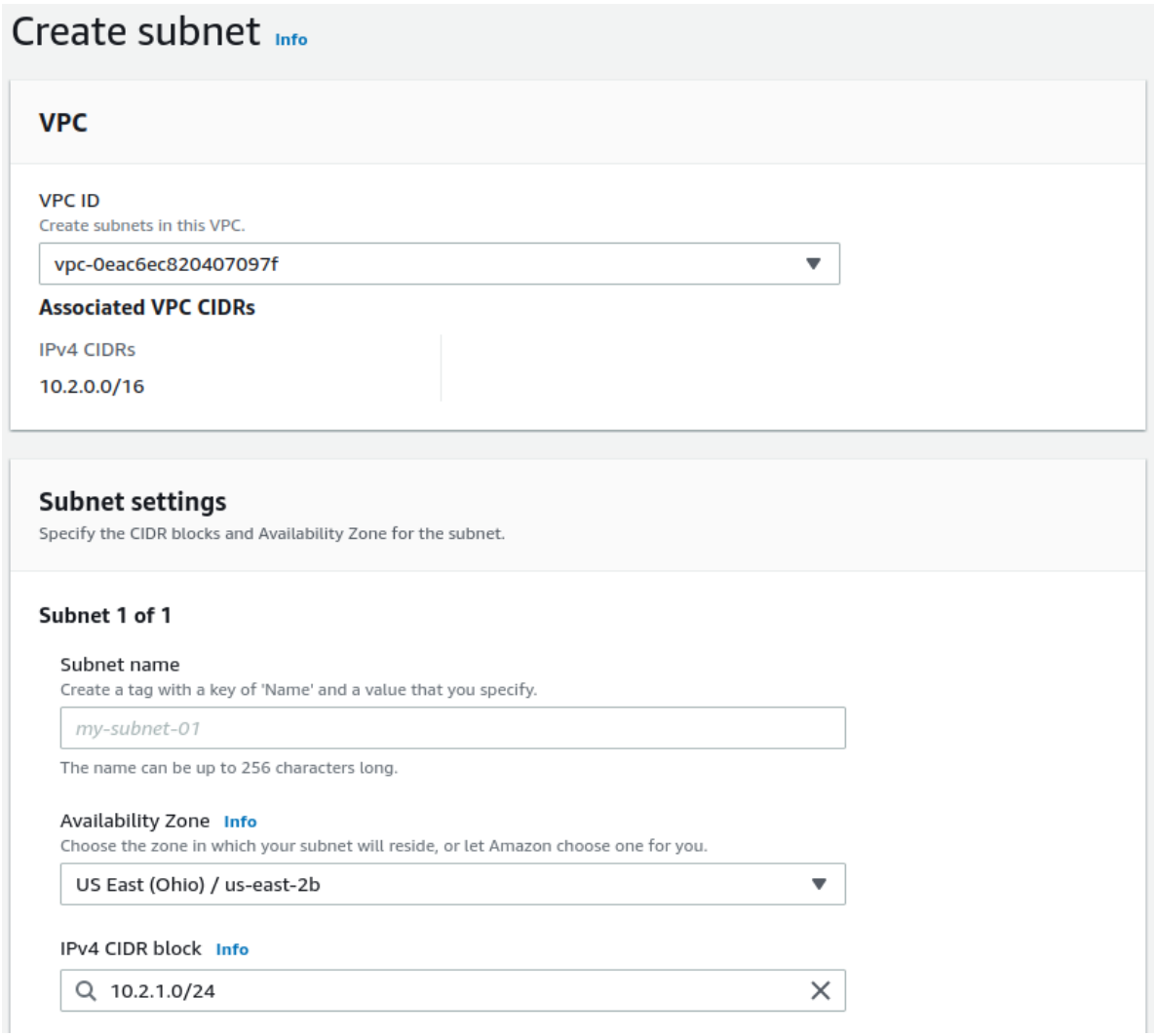
4. Create the private subnet(s).
    - Navigate to the **Subnets** view in the menu on the left side of the VPC Management Console

      The browser may still be in this view after completing the previous task.
    - Click the **Create Subnet** button again
    - Select the same VPC from the **VPC ID** drop-down
    - Optionally enter a **Subnet name**
    - Optionally choose the same **Availability Zone** as the previous subnet
    - Enter the subnet to use for the private network in the **IPv4 CIDR Block** field

      This subnet **must** be a block that is within the address space assigned to the VPC, and it must be distinct from the public subnet.

      This is the subnet to which the LAN interface of the Netgate appliance instance is attached and could include any other hosts or appliances that should be protected behind the Netgate appliance.

      For this example, use `10.2.1.0/24`.
    - Click the **Create subnet** button



5. Configure the route table

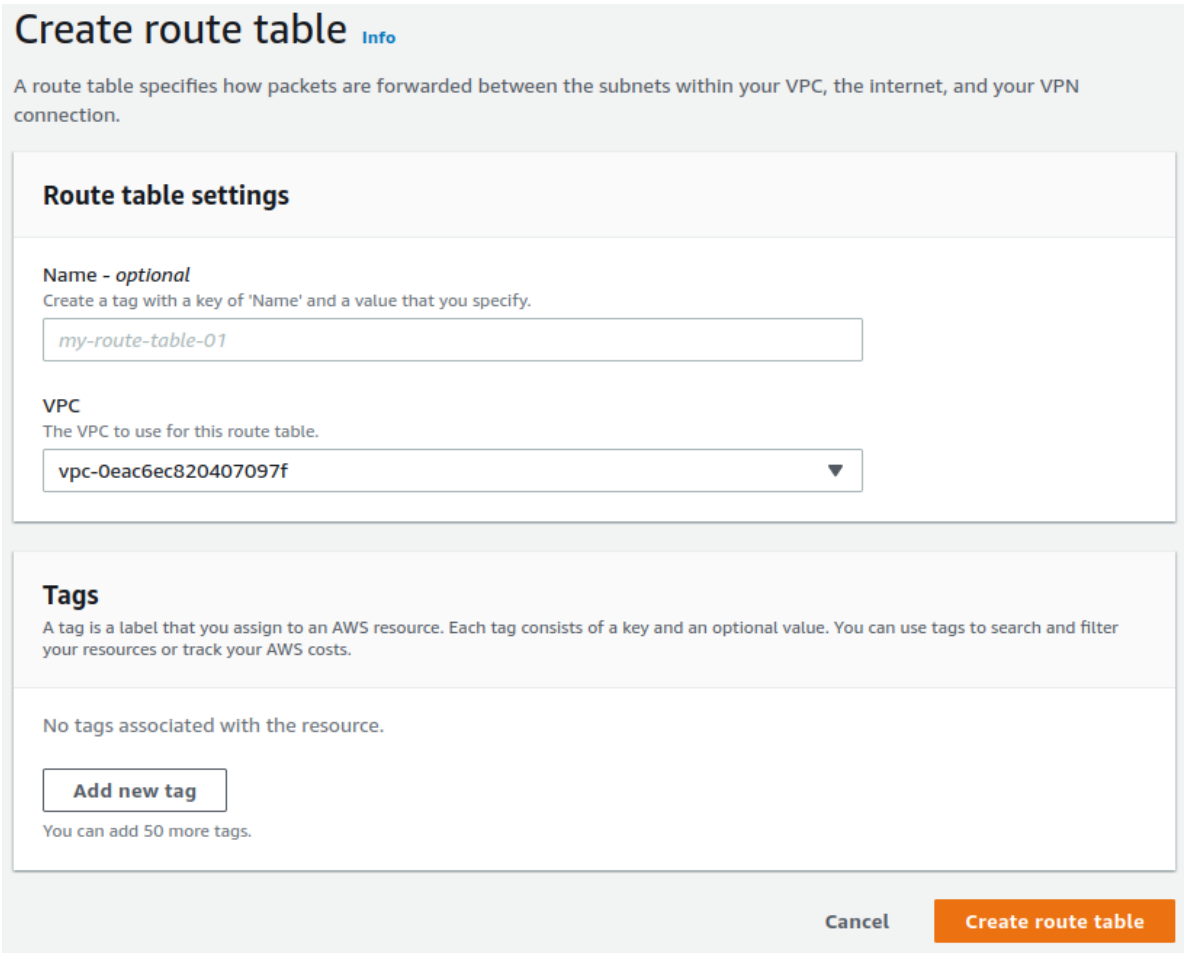
   Both new subnets start out set to use a default route table automatically created for the VPC by AWS. The private subnet can continue to use that default table.

   Create a new route table for the public subnet to override this behavior:

   - Navigate to the **Route Tables** view in the menu on the left side of the VPC Management Console

     This view will contain all the existing route tables, and will at least contain the route table created by AWS for the VPN.

   - Click the **Create route table** button
   - Optionally enter a **Name** for the route table
   - Select the **VPC**
   - Click the **Create route table** button

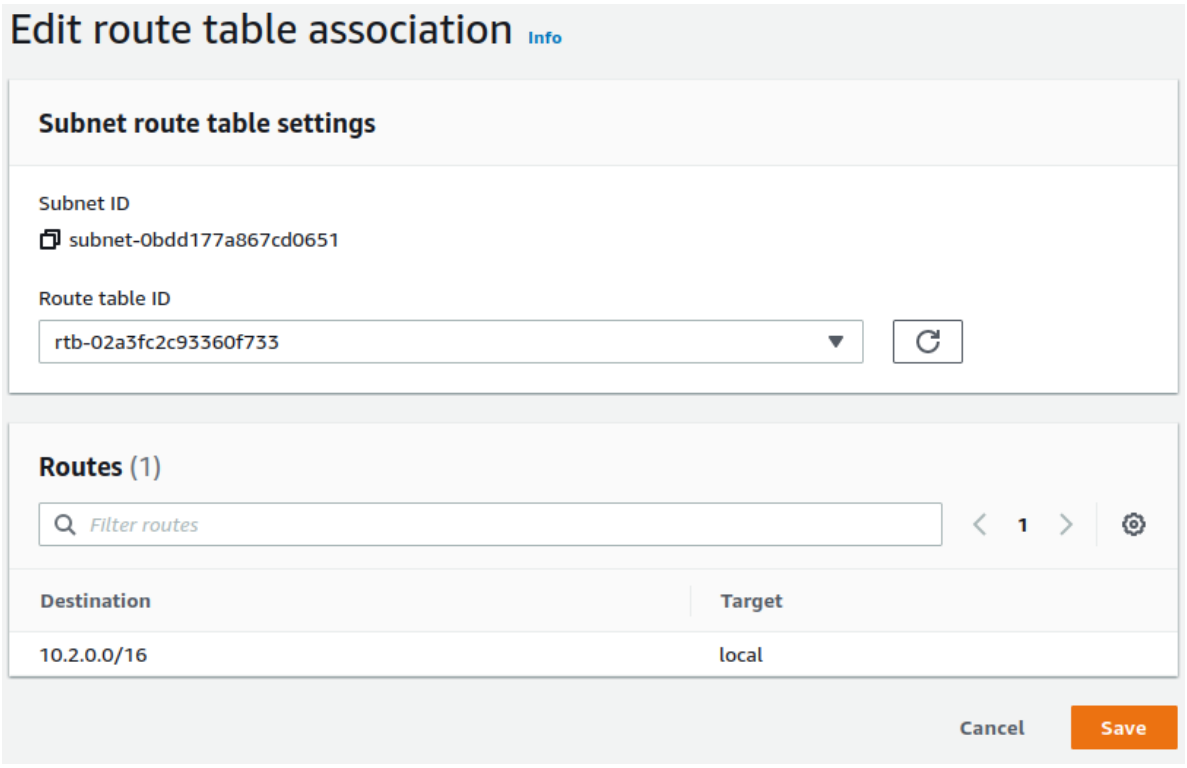


6. Associate the public subnet with the newly created routing table

   The public subnet is `10.2.0.0/24` in this example.

   - Navigate to the **Subnets** view on the left-hand side of the VPC Management Console.
   - Check the checkbox next to the public subnet
   - Scroll down to the information displayed for the selected subnet

     The **Details** tab contains the **CIDR block**, **VPC**, and **Availability Zone** among other information.
   - Click the **Route table** tab
   - Click **Edit route table association**
   - Select the **Route table ID** corresponding to the route table created for the public subnet
   - Click the **Save** button

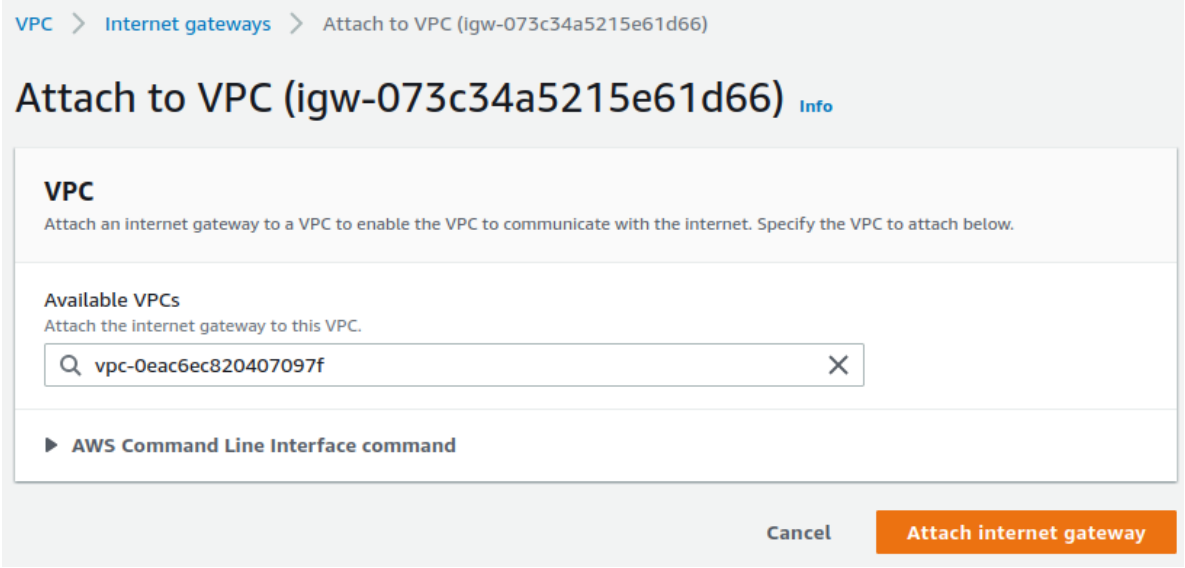


7. Create an Internet Gateway and attach it to the VPC

   To send traffic from the public subnet to the Internet, the VPC requires a default route to an Internet Gateway. To create this gateway:

   - Navigate to the **Internet gateways** view in the menu on the left side of the VPC Management Console
   - Click the **Create Internet Gateway** button
   - Enter a **Name tag** for the gateway
   - Click the **Create Internet Gateway** button

     The view should be filtered to display the new gateway. If it is not, then click the checkbox next to the newly created gateway entry in the list.
   - Click **Actions**
   - Click **Attach to VPC**
   - Select the VPC in the **Available VPCs** box
   - Click the **Attach internet gateway** button



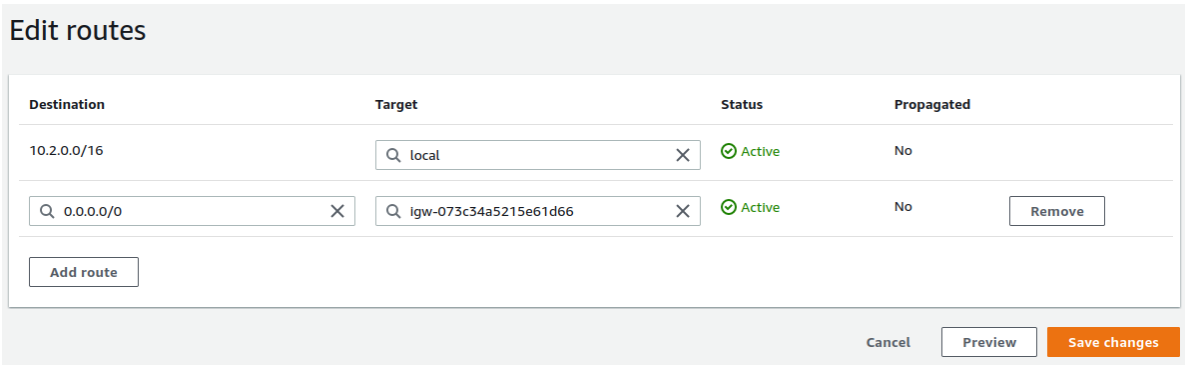
8. Associate the gateway with the public route table

   The route table for the public subnet must be updated so that it has a default route to the Internet Gateway.

   - Navigate to the **Route Tables** view on the left-hand side of the VPC Management Console.
   - Check the checkbox next to the route table for the public subnet
   - Scroll down to the details of the route table
   - Click the **Routes** tab

     The **Routes** tab for this route table should contain a single route for the CIDR block of the VPC (`10.2.0.0/16` in this example) that has a target of **local**.
   - Click **Edit routes**
   - Click **Add route**
   - Click the **Destination** text box in the new row
   - Enter a **Destination** of `0.0.0.0/0`
   - Click the **Target** text box in the same row
   - Click **Internet gateway** from the list
   - Select the Internet Gateway created previously

     The entry should be formatted similar to `igw-XXXXXXXX` and will also have the name configured for the gateway.
   - Click **Save Changes**



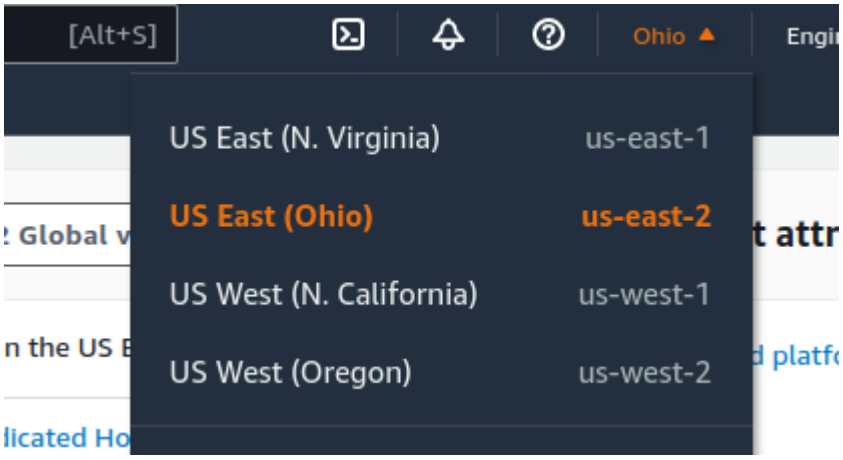
There are a few more VPC configuration changes that will be required later, but the next step is to launch a Netgate appliance instance.

3.1.2 Launching an Instance in a VPC

In the **Amazon EC2 Management Console**, launch a new instance of the Netgate® pfSense® Plus software firewall and VPN appliance.

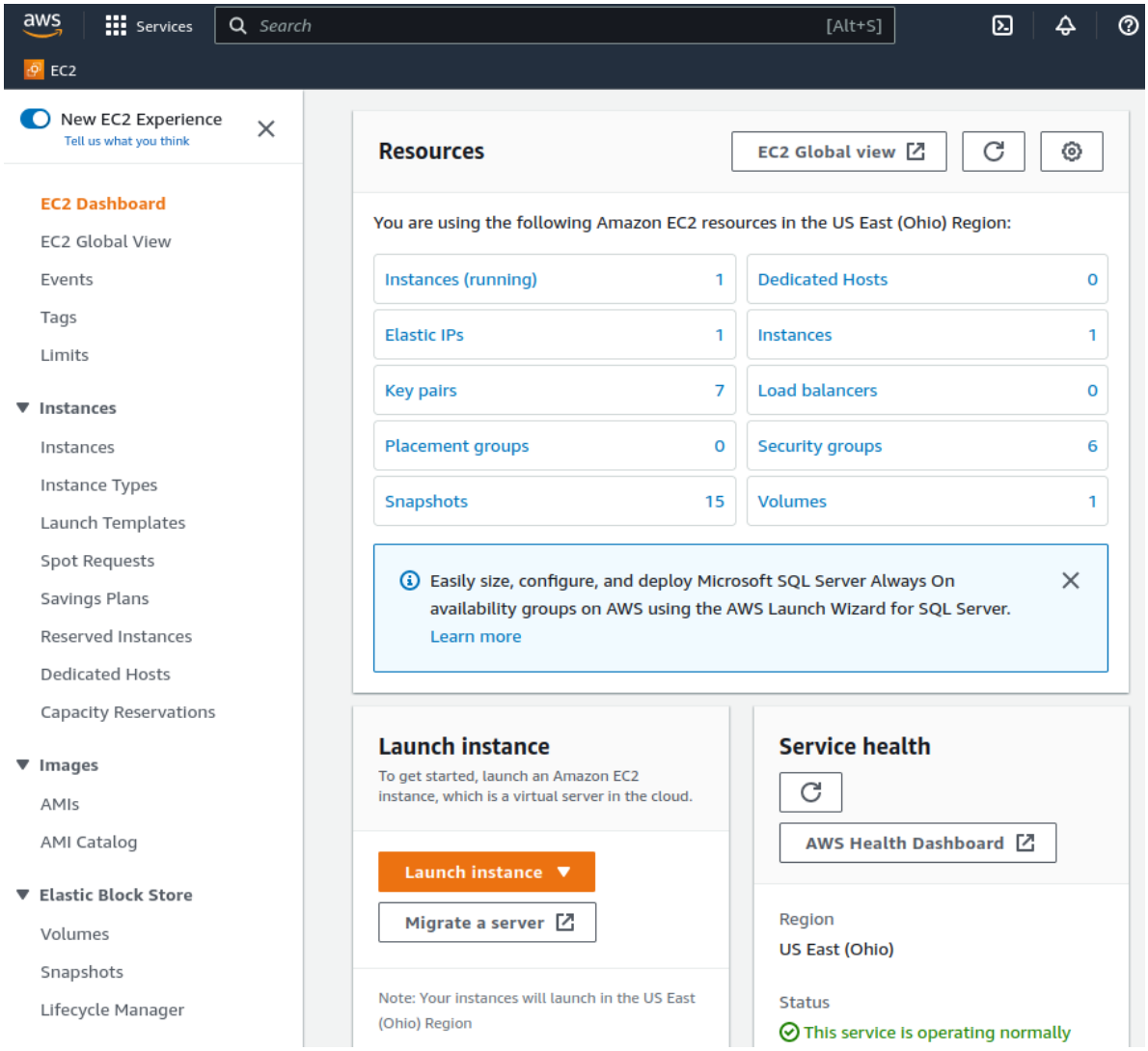
This process is the same as the one for launching an EC2 (non-VPC) instance, up until the **Network Settings** in order to specify the instance should be created in the VPC.

1. Select the region in which the instance will run
   - Click the current **Region** name near the upper right corner of the page
   - Select a new region if necessary



2. Enter the **Launch Instance Wizard**
   - Click the **Launch Instance** button to open the **Launch Instance** menu

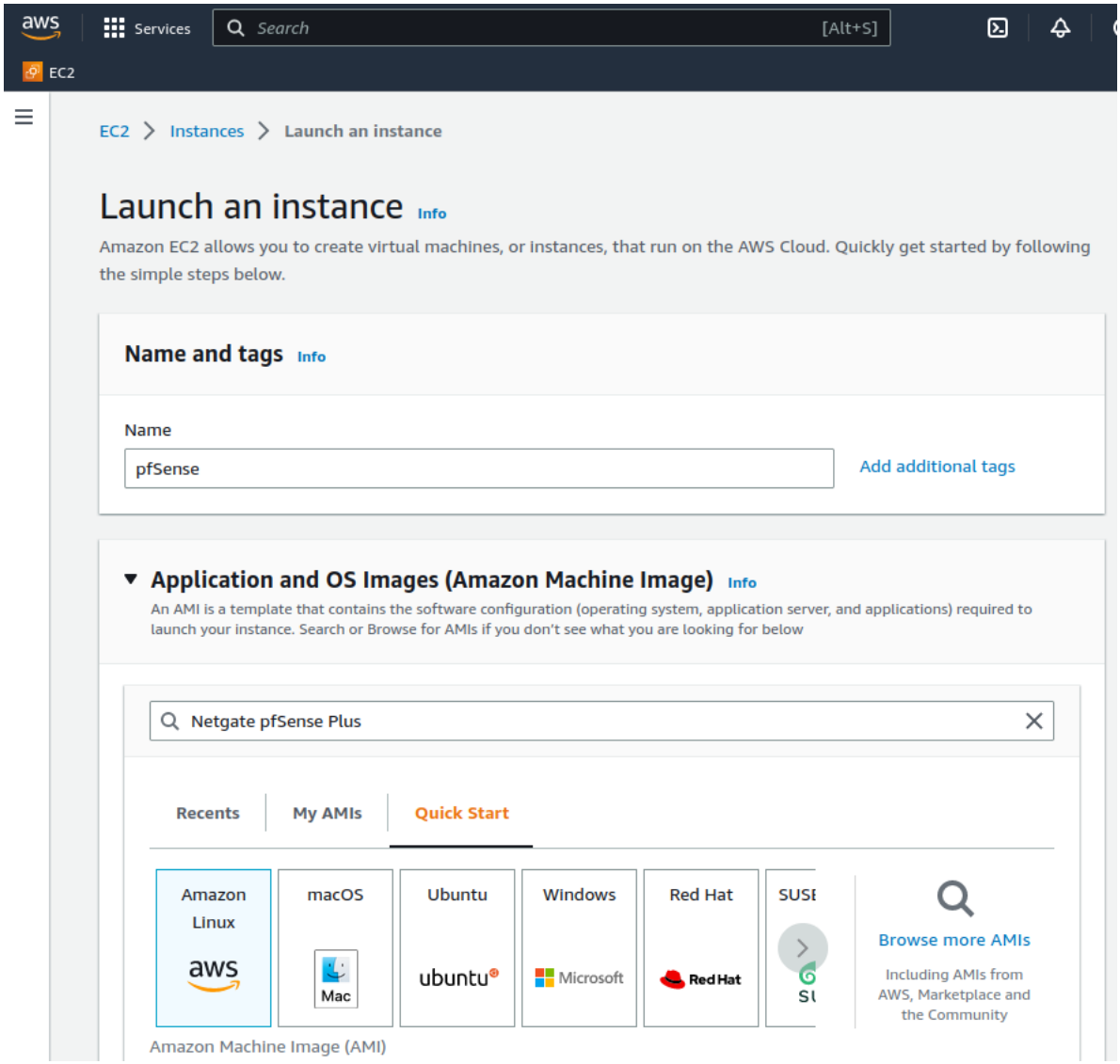
     This button is in the **Launch Instance** section which is located under the **Resources** section of the EC2 dashboard.
   - Click **Launch Instance** from the menu



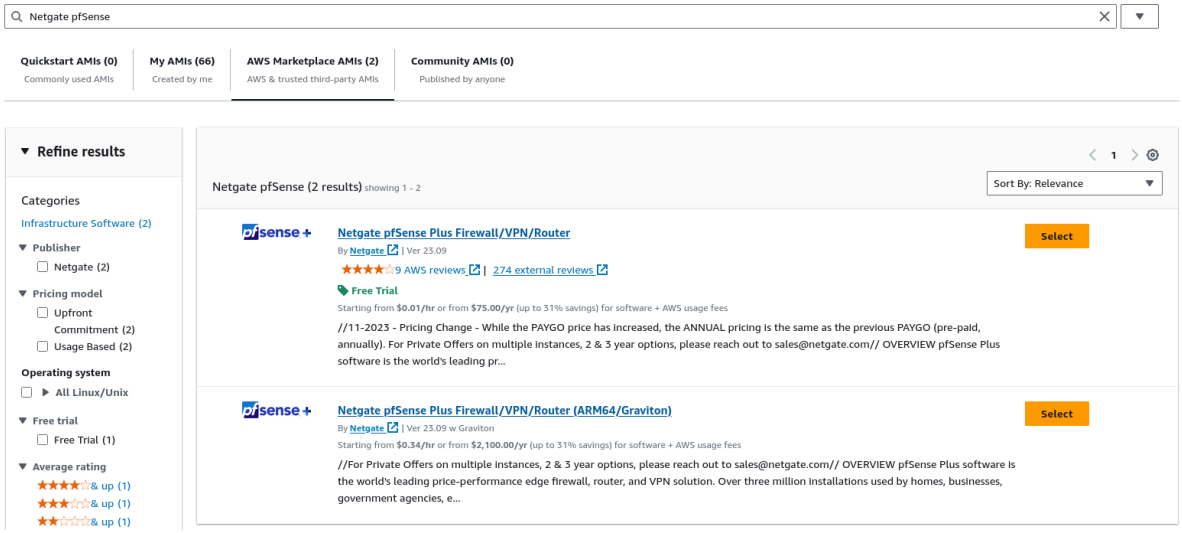
3. Give the new instance a **Name**, such as `pfSense`

   Optionally, click **Add Additional Tags** to create more tags which can be used to identify and locate this instance.

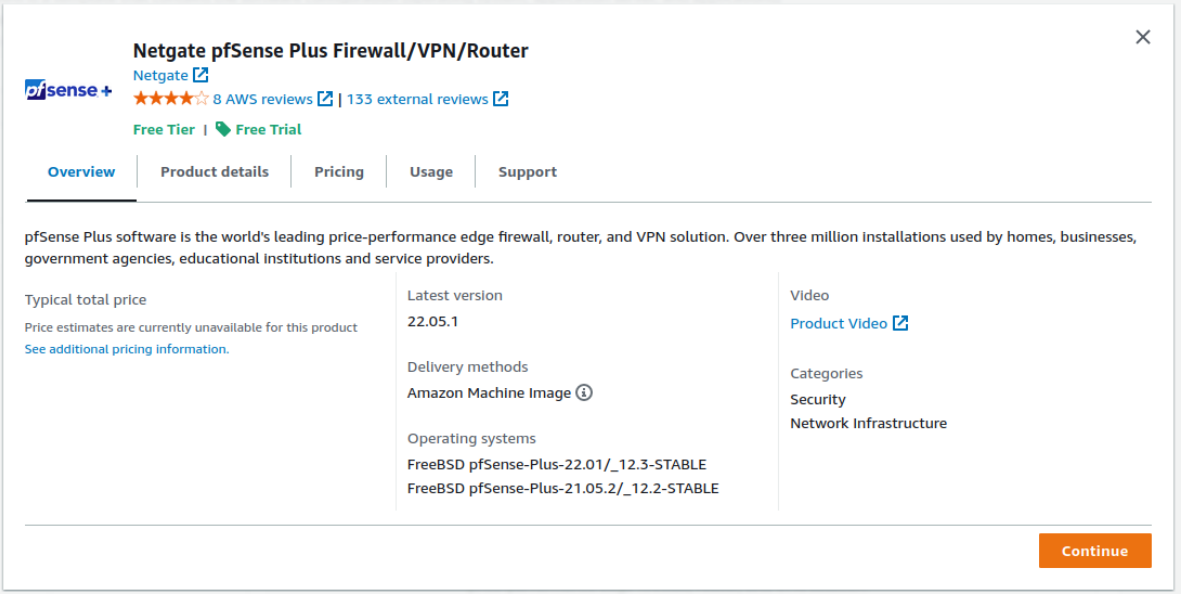
4. Type `Netgate pfSense Plus` in the search box and press `Enter`.



5. Select **AWS Marketplace AMIs** if it is not automatically highlighted
6. Click the **Select** button for the **Netgate pfSense Plus Firewall/VPN/Router** listing in the search result that corresponds to the desired type of instance. This could be either the amd64 AWS product or the arm64/Graviton AWS product depending on the needs of this deployment.



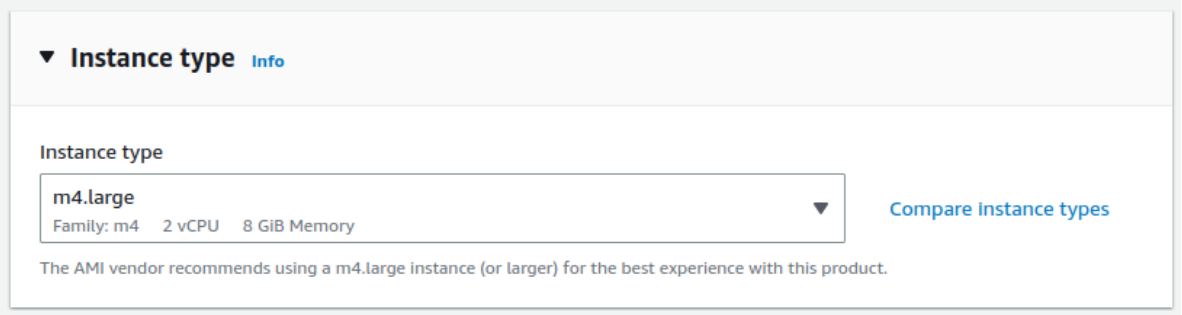
7. Review pricing and other helpful information, then click **Continue**.



8. Choose an **Instance Type** from the drop-down

See also:

For guidance on which instance type to choose, see *Choosing Instance Type and Sizing*.



9. Configure an SSH **Key Pair**

   The **Key Pair** section of the form sets the SSH key pair used by an SSH client when it connects to the TNSR instance for management.

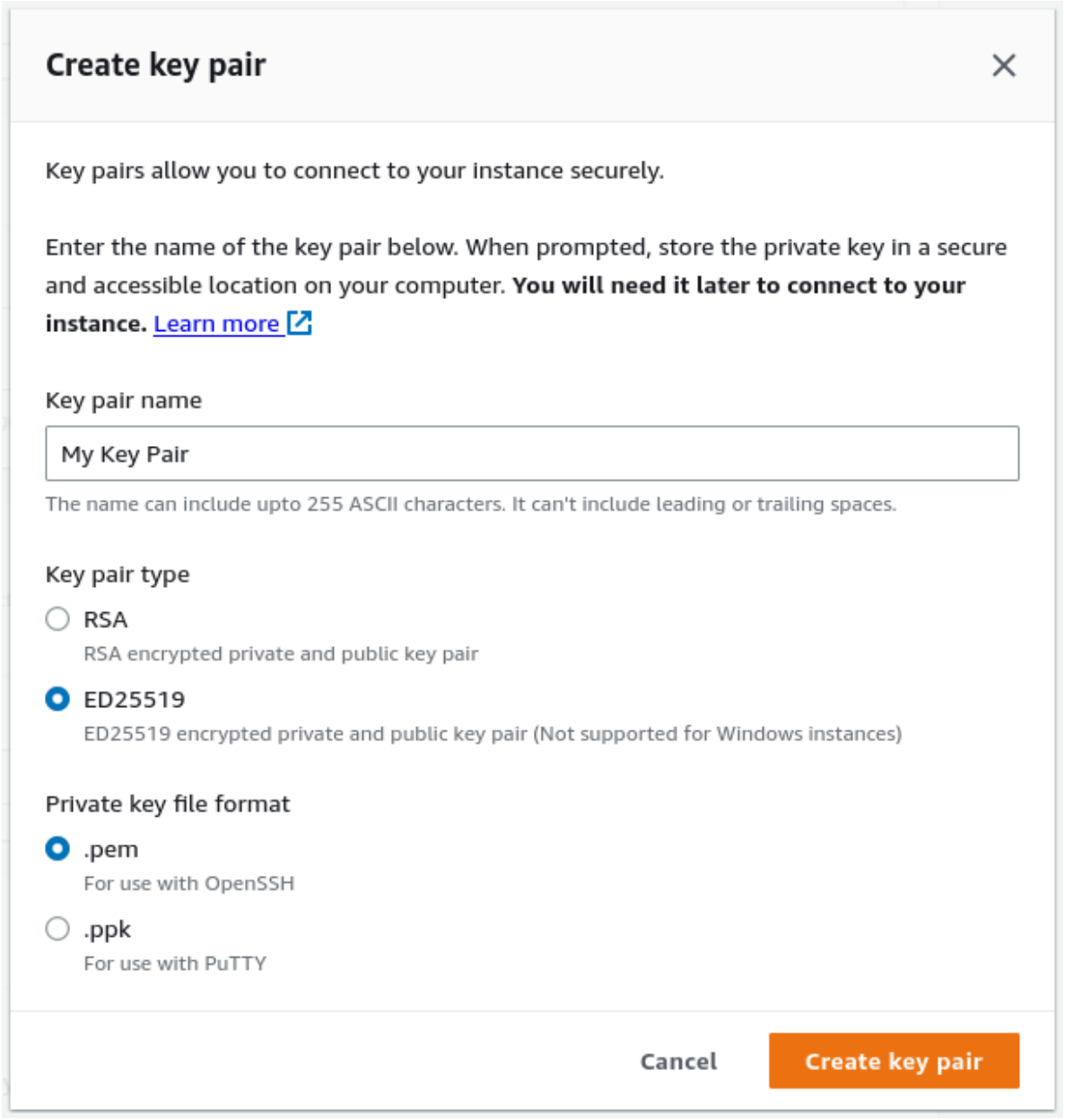
   For an existing key pair:

   - Click **Key pair name**
   - Search for and select an existing key pair entry

   To create a new key pair:

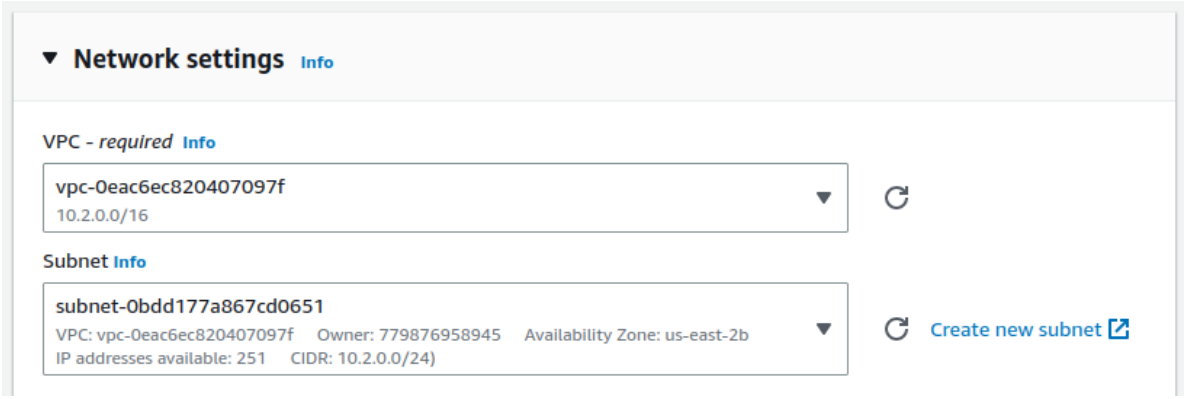
   - Click **Create new Key Pair**
   - Enter a **Key pair name**, such as **TNSR SSH Key**
   - Select a **Key Pair Type** and **Private Key Format**

     The chosen type and format must be compatible with whichever local SSH client will be used by TNSR administrators
   - Click the **Create key pair**
   - Select a location to save the key pair locally



10. Click **Edit** under **Network Settings** to allow making changes for the next few steps.

11. Configure the **Network Settings**
    - Select the **VPC**
    - Select the public **Subnet** (e.g. `10.2.0.0/24`)



12. Configure **Security Groups**

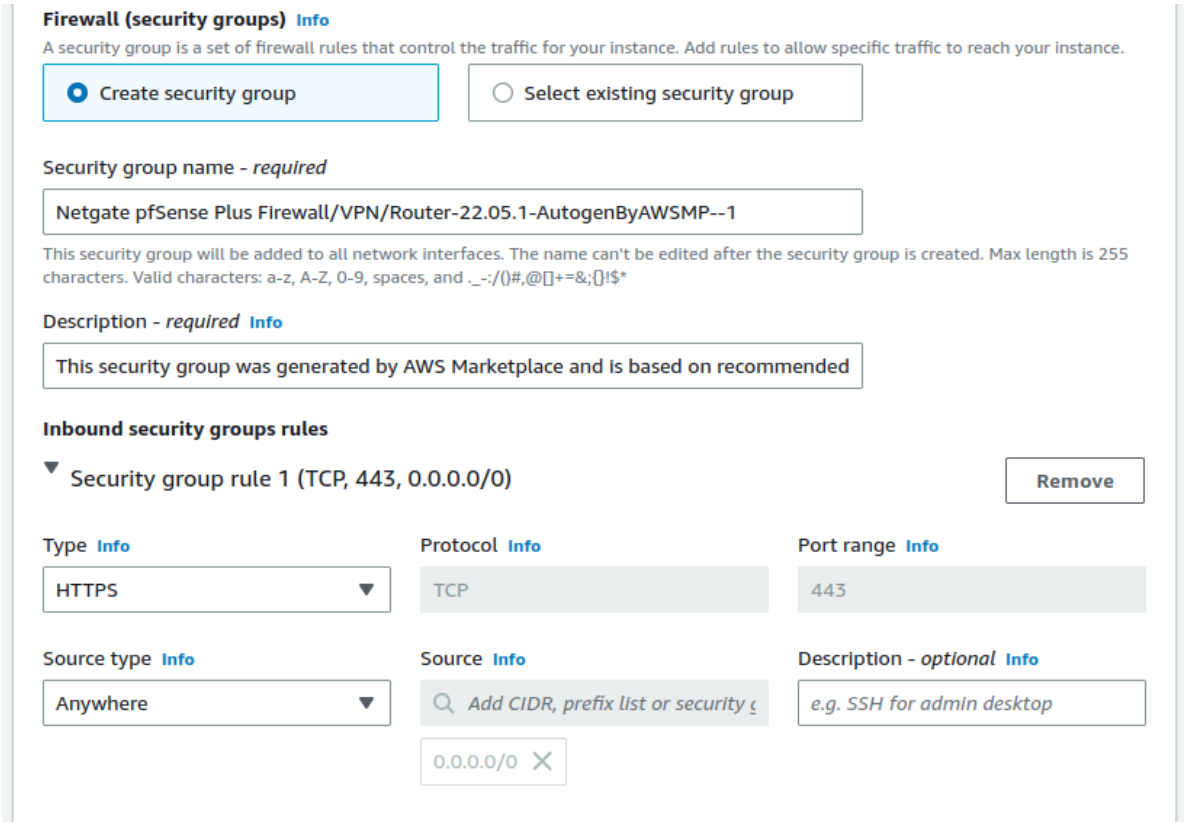
    Select a security group to launch the instance with. This controls which traffic AWS allows to reach the instance.

    If there is an existing security group which includes this access:

    - Click **Select an existing security group**
    - Select the group(s) to use.

    If there is no existing group:

    - Click **Create security group**
    - Optionally, change the **Security group name** and **Description**
    - Add/Remove/Change individual rules as necessary to allow only the traffic required by the instance.



By default, the instance suggests a configuration to allow access to the firewall management ports (TCP/80, TCP/443, TCP/22) but there are several other common things which may need to be passed.

Security group rules may be configured by selecting pre-defined **Types** or by entering custom values.

The best practice settings for the security group rules should allow at least the following traffic:

- HTTPS (TCP port `443`) from `0.0.0.0/0`

  This is the port that the management GUI listens on.

- SSH (TCP port `22`) from `0.0.0.0/0`

  This port can be used to connect to a command prompt with an ssh client.

- Custom UDP, Port `1194` from `0.0.0.0/0`

  OpenVPN - The OpenVPN server that is configured by default is bound to this port.

- Custom UDP, Port `500` from `0.0.0.0/0`

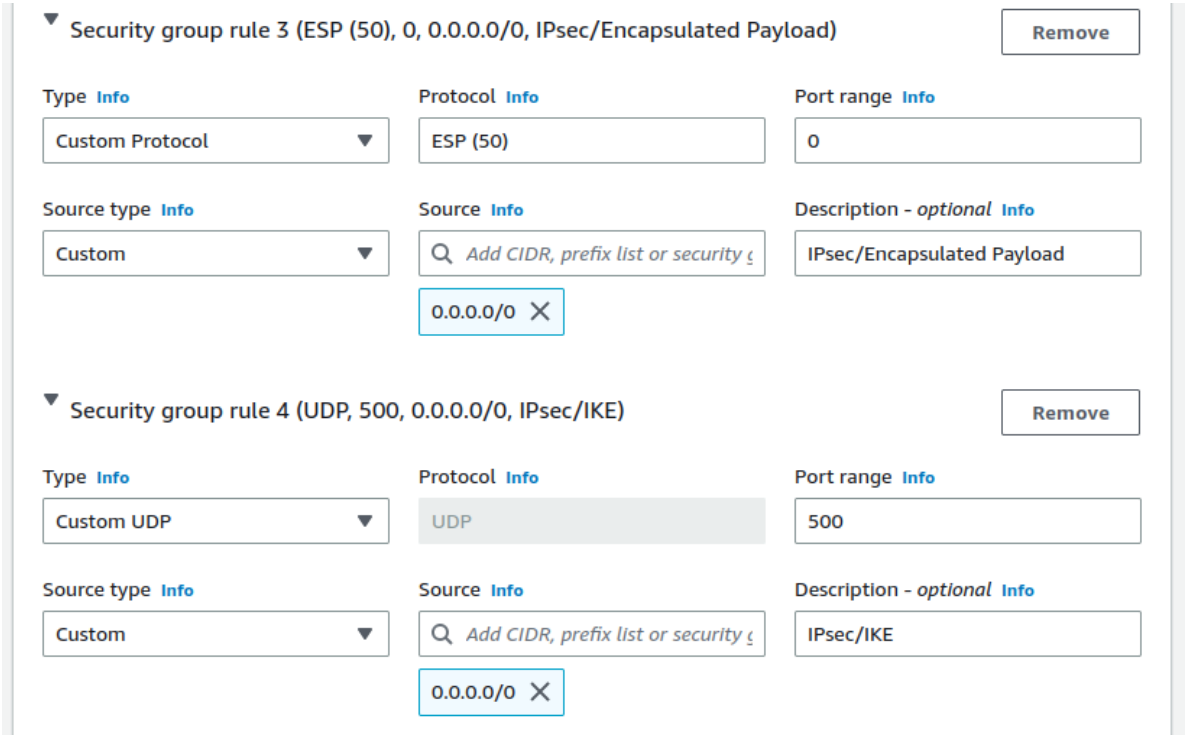
  IKE for IPsec VPN.

- Custom Protocol, `ESP` (50)

  Encapsulated IPsec traffic

- Custom UDP, Port `4500` from `0.0.0.0/0`

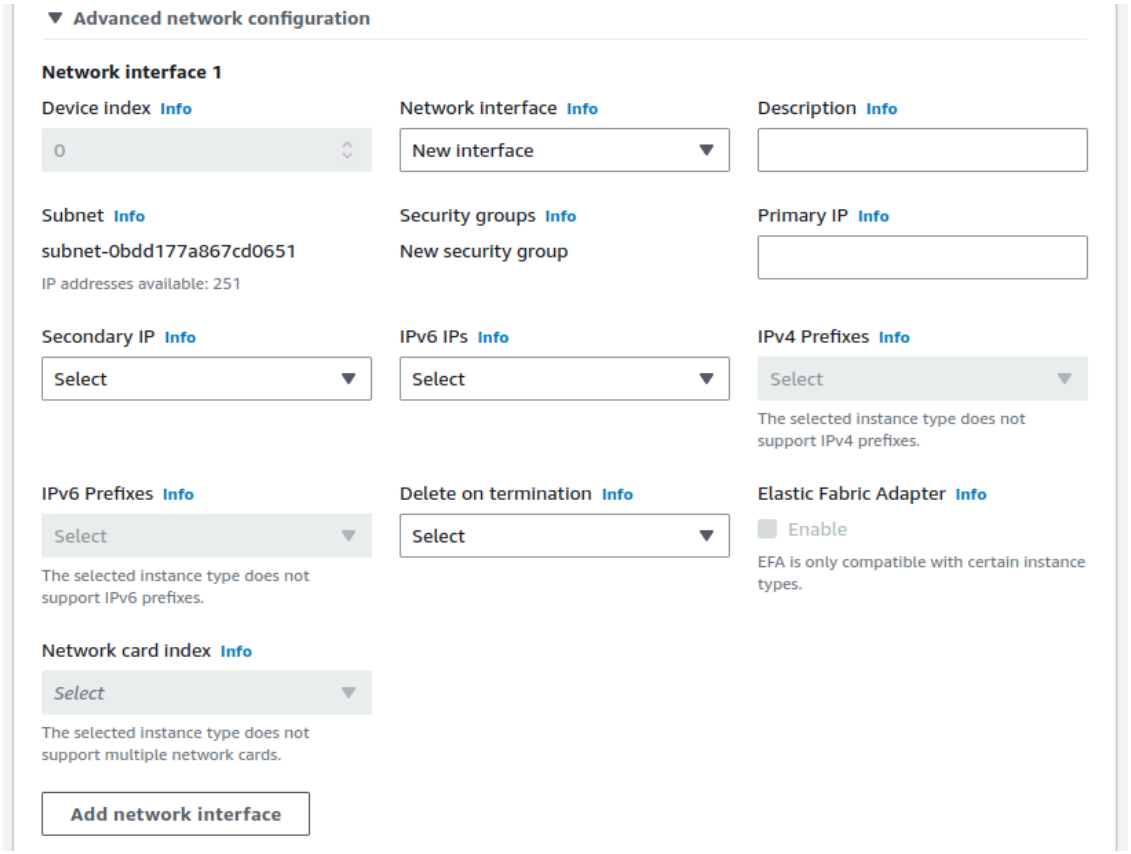
  IPsec/NAT-T for IPsec VPN.



13. Add the LAN interface

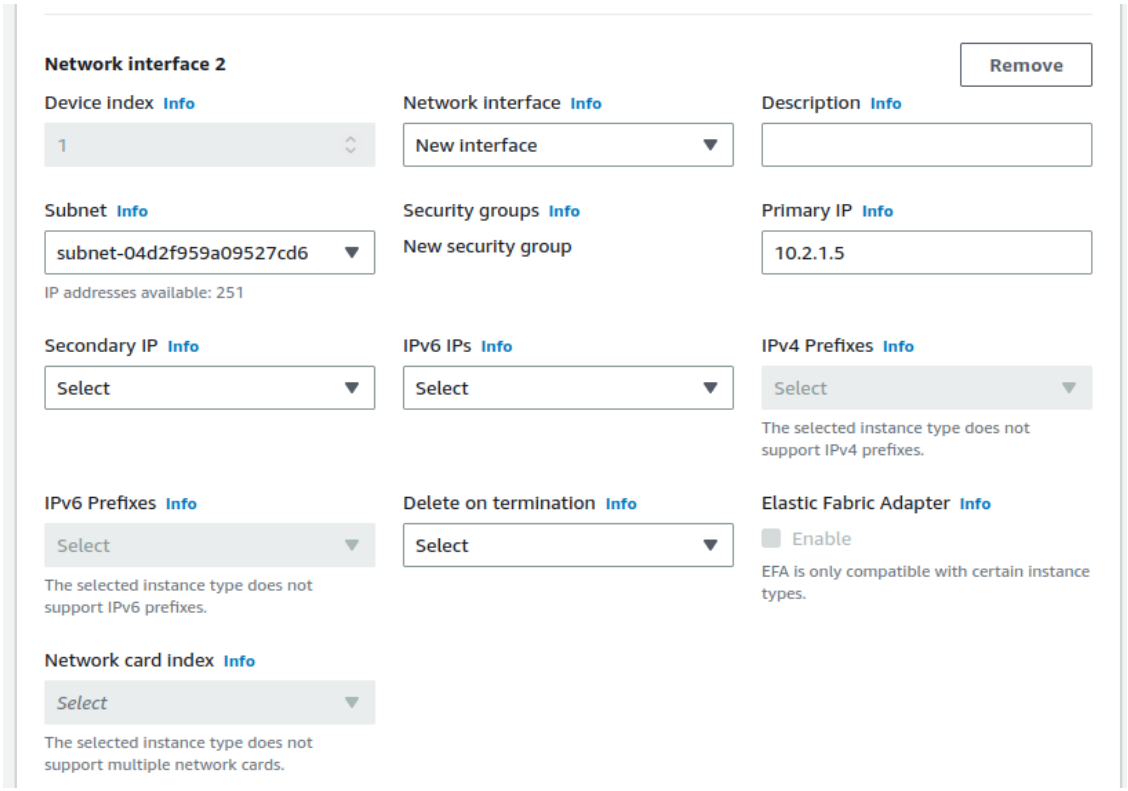
    - Scroll down to **Advanced network configuration**
    - Click **Advanced network configuration** to expand the section

      This section contains a single interface by default, which corresponds to the WAN / public subnet.



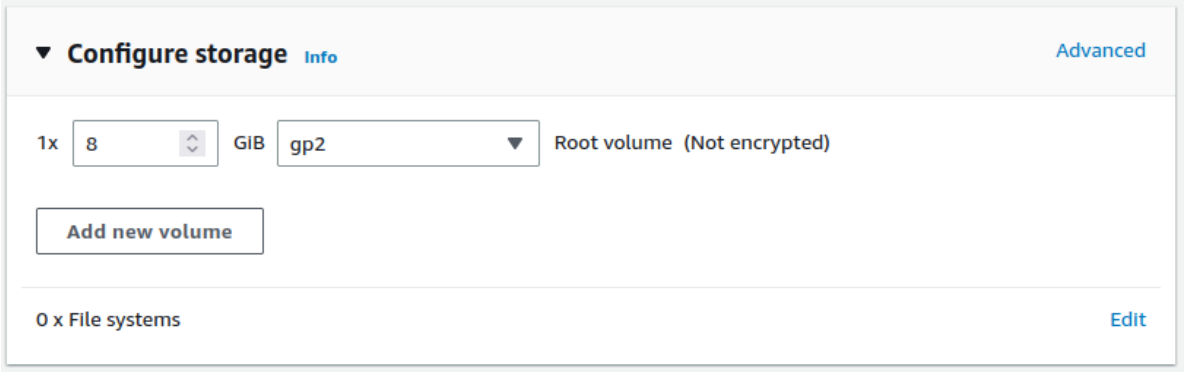
- Click **Add network interface**
- Select the private subnet that was created (e.g. `10.2.1.0/24`)
- Enter an IP address from the private subnet in the **Primary IP** field

  Keep in mind that the first 3 or 4 IP addresses are reserved. For this example, use `10.2.1.5`.



14. Configure storage

    If this instance will require more than the default 8 GiB disk, increase the value in the **Configure Storage** section



15. Configure **Advanced details**

    There are a couple optional parameters which can influence how the instance will start the first time.

    - Click to expand the **Advanced Details** section
    - Set parameters as text in the **User Data** field.

    The available options are:

    password
    : Setting a value via a directive like `password=abcdefg` will set the password for the administrative account to the specified value – `abcdefg` in this example. If no value is set here, a random password will be assigned in order to keep administrative access from being exposed to the internet with a default password.

> **Note:** A password configured using this method cannot contain the characters `:` or =, which are reserved for use as delimiters by the script which handles importing these values.

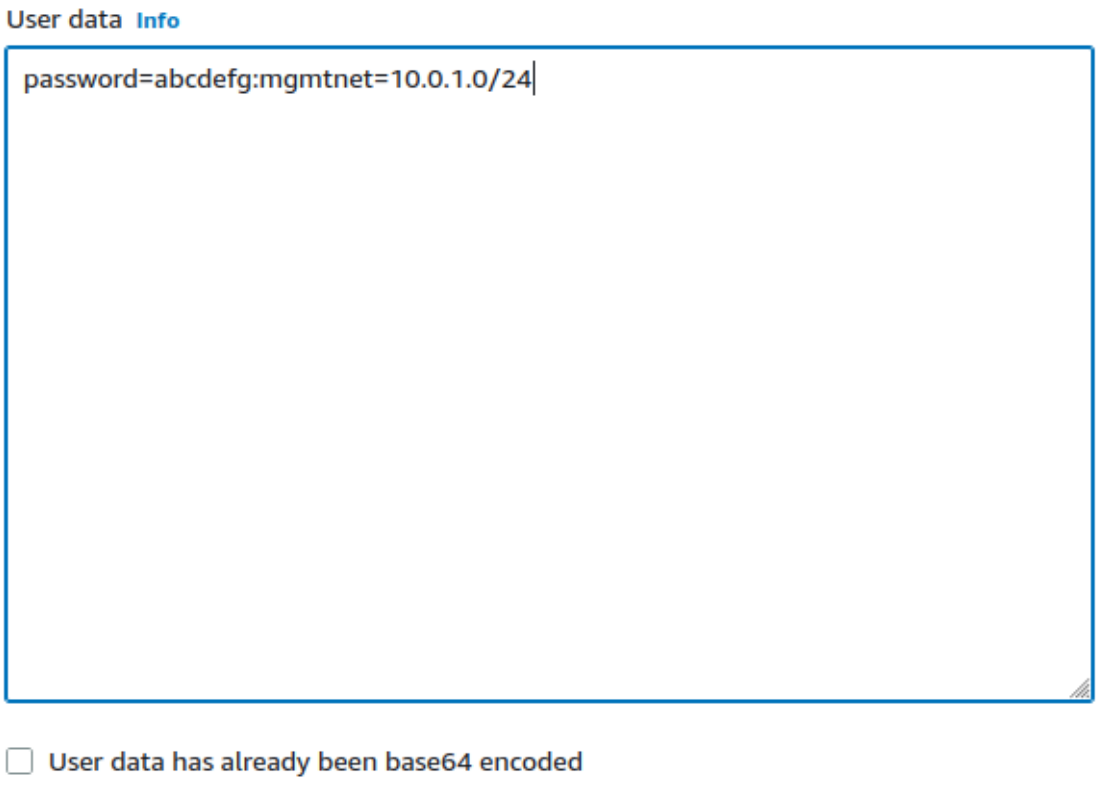
mgmtnet
Setting a value via a directive like `mgmtnet=10.0.1.0/24` will restrict management access (HTTP, HTTPS, SSH) to the specified network – `10.0.1.0/24` in this example. This will cause the firewall rule on the instance (not on Amazons access lists, but on the Netgate appliance's own firewall) to restrict management traffic for the instance to the specified source network. The default behavior is to allow management from any host.

These directives can be set by placing them on a single line in the **User Data** field and separating them with colons. Specify both parameters, by typing a statement similar to:

```
password=abcdefg:mgmtnet=10.0.1.0/24
```

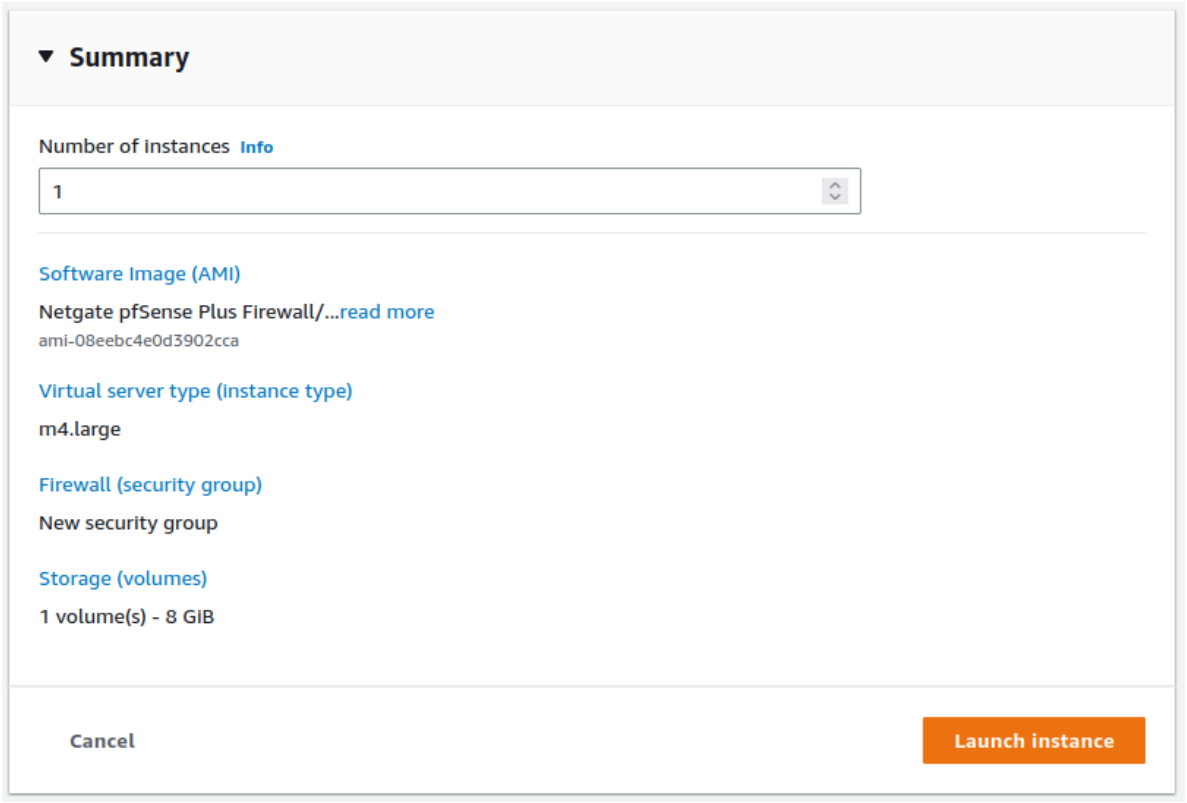
Note: If setting a password using the password parameter listed above, the password is retrieved by the instance via an unencrypted HTTP request when the system is configured the first time it boots. The request is made to an Amazon Web Services-operated server on the local LAN that stores metadata about each instance running. The data for an instance is only made available to that instance, but is available to be queried from the instance without providing any authentication credentials.

It is advised to change the admin password via the pfSense® Plus GUI after the instance comes up, or choose not to set the password at all and let a random password be set.



16. Verify the settings selected in earlier steps and review any errors or recommendations displayed by AWS

17. Click **Launch instance** in the **Summary** box on the right side



18. Allocate an Elastic IP address

    To reach the instance from the Internet, associate an Elastic IP with the WAN interface of the instance.

    For each interface that needs a public Elastic IP Address, allocate one by following the instructions at https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/elastic-ip-addresses-eip.html#using-instance-addressing-eips-allocating

    Before associating an Elastic IP Address to a **Network Interface**, make a note of the ID of the **Network Interface** to use. To find the **Network Interface ID**:

    - Navigate to https://console.aws.amazon.com/ec2/
    - Click **Instances**
    - Click the checkbox next to the pfSense Plus instance to select it
    - Look at the bottom of the page, under the **Networking** tab to see **Network Interfaces**
    - Click on the interface names to display information about the **Network Interface**

      Determine which interface is which by their private IP address, which will match either the public (WAN) or private (LAN) subnet.

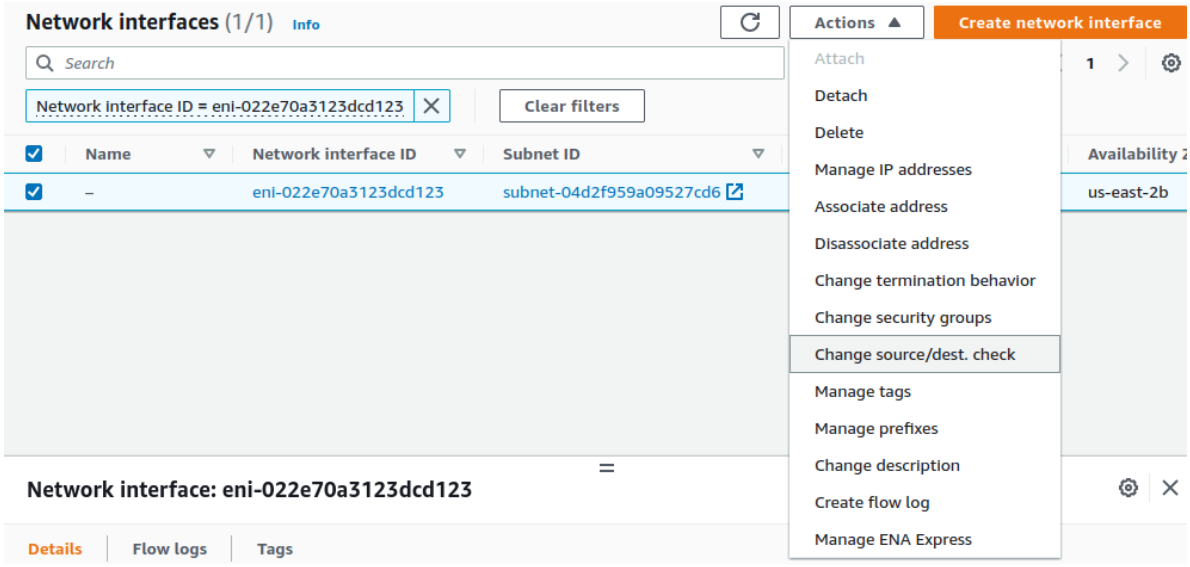
    - Write down the **Interface ID** for the WAN interface

    After allocating the Elastic IP Address and finding the Network Interface ID for WAN, associate the Elastic IP Address to the Network Interfaces by following the instructions at https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-eni.html#associate_eip

19. Disable Source/Destination address checking

To allow the firewall to route traffic from the private subnet through the public interface of the instance, the **Source/Dest Address Check** on the private interfaces needs to be disabled:

- Navigate to https://console.aws.amazon.com/ec2/
- Click **Instances**
- Click the checkbox next to the pfSense Plus instance to select it
- Look at the bottom of the page, under the **Networking** tab to see **Network Interfaces**
- Click on the LAN interface name to filter the view to only that interface
- Click the checkbox to the left of the interface
- Click **Actions** at the top of the page
- Select **Change Source/Desk Check** from the popup menu
- Uncheck **Enable**
- Click **Save**



Non-local traffic from the private subnet should now be sent through the private/LAN interface on the Netgate appliance instance.

3.1.3 Managing a VPC Instance

Once the instance is launched, connect to it via the Elastic IP address attached to the primary interface during the provisioning phase.

In order to manage the configuration of the instance, connect to it via **HTTPS** or **SSH**. A limited set of configurations is possible through the SSH interface, the preferred method for managing most of the configurations or viewing data on the status of the Netgate® pfSense® Plus instance is through the HTTPS GUI.

Connecting via SSH

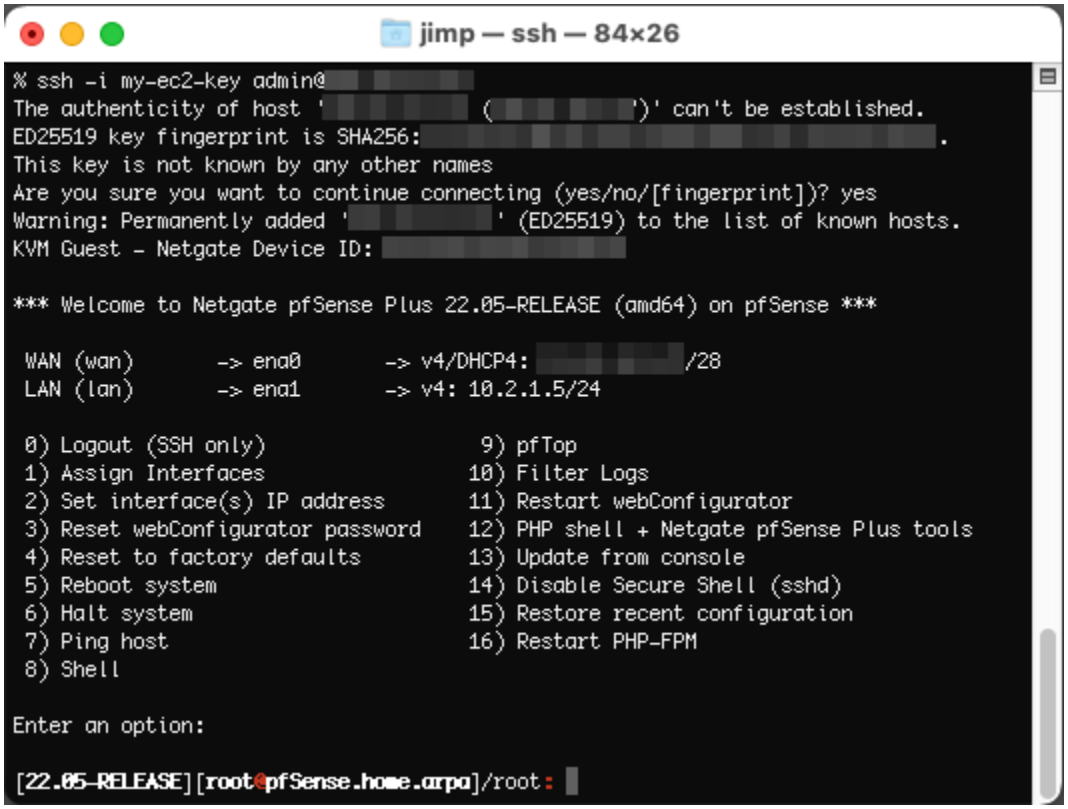
To connect via SSH, use the key pair chosen while creating the instance to connect to the `admin` account. From the command line on a Unix/Linux host, use a command similar to:

```
ssh -i my_ec2_key admin@213.0.113.54
```

Where the appropriate private key file and public IP address or hostname are substituted.

Note: The first time logging into the instance, the SSH key for the instance will not be cached locally, type `yes` when asked whether to continue connecting. This should not be necessary on subsequent sessions.

Once logged in, the client will display the console menu similar to the following:



Connecting via HTTPS

To connect via HTTPS, enter an `https://` URL containing the public IP address or hostname of the instance into a web browser. For example, `https://23.20.204.54`.

There will likely be a browser warning indicating that the security certificate of the site is not trusted, because the instance uses a self-signed certificate for HTTPS communication. Click on the option to proceed to the site anyway and a login screen with the Netgate logo should appear.

The username to log in with is `admin`. The password to use is either a value set in the **User Data** during the creation of the instance or a random password.

Tip: The best practice is to explicitly set a password by passing a value in with the **User Data** field, so the password will be known in advance, and then to change it after logging in the first time.

If a specific password was not set, The value of the random password can be found through one of two different means:

1. The first method is to log in over SSH with the key pair selected when the instance was created and examine the contents of the file located at `/etc/motd-passwd`. Do this by selecting option `8) Shell` in the console menu that is presented when connecting via SSH, then run this command in the shell:

```
cat /etc/motd-passwd
```

2. Alternatively, view the **System Log** for the instance in the EC2 Management Console. After the messages that are displayed that show the status of the boot process, a message should appear that indicates the value of the administrative password.

   Note: The **System Log** output in the EC2 Management Console is not updated in real time and may take a few minutes to show up. It is preferable to explicitly set a password by passing a value in with the **User Data** field, so the password will be known in advance. To allow a random password to be set, connect via SSH and find the value of the password after the instance is up without any delay.

The message, using either of the methods mentioned, will look like this

```
***
***
*** Admin password changed to: abcdefg
***
***
```

In this example, the password was changed to `abcdefg`.

Once the password has been determined and entered into the login form, the pfSense® Plus GUI should be available.

3.1.4 Forwarding traffic from VPC subnets through the instance

Some additional configuration is required within the VPC instance pfSense® Plus GUI before the instance can manage traffic from the private subnet.

1. Log into the pfSense® Plus GUI for the instance.
2. Click on the **Interfaces** heading on the left and then click the **Assign** link
3. Click on the **+** icon to add a new Interface under the **Interface assignments** tab. A LAN interface should automatically be added with the next available network interface (`xn1`)
4. Click on the **Interfaces** heading on the left again and then click on **LAN**. Click the checkbox to enable the LAN interface. Set the **IPv4 Configuration Type** to **Static IPv4** and enter the IP address assigned to the second interface during the provisioning phase. Click the **Save** button.

Now is it possible to create instances attached to the private subnet and protect them with the firewall on the pfSense® Plus instance.

Common ways to manage private hosts

Allowing private hosts to connect to the Internet

To allow private hosts to be able to connect to the Internet, one method is to allow any traffic from the LAN in the firewall rules. There should be a rule like this in place by default.

Next, set up NAT rules, so the firewall will apply NAT to addresses in the private subnet using the IP address of the WAN interface:

1. Navigate to **Firewall > NAT**, **Outbound** tab
2. Select the radio button for **Hybrid Outbound NAT**
3. Click the **Save** button
4. Navigate to **Firewall > Aliases**
5. Add the private subnet to the `Networks_to_NAT` alias.

> **Note:** There is an existing NAT rule configured by default that uses the alias `Networks_to_NAT`.

Allow private hosts to connect to each other

If hosts should only contact each other and a private network segment elsewhere, configure an IPsec or OpenVPN tunnel from the remote networks to the Netgate® pfSense® Plus appliance instance and set up the appropriate firewall rules, routes, and security policies to allow access to the private subnet through a VPN tunnel.

Allow direct inbound access from the internet to hosts

To enable direct inbound access from the internet to hosts on the private subnet, set up port forwarding on the WAN interface to direct traffic to particular hosts in the private subnet.

3.1.5 Establishing a VPN connection to a VPC in another region

The Netgate appliance has a VPC configuration wizard which configures the Netgate appliance and VPC configuration elements to establish a VPN that allows instances on the VPC subnet(s) that sit behind a Netgate® appliance to communicate with instances that reside in a VPC in another region. Without the wizard, these settings must be set manually through the AWS Management Console

For detailed instructions, see the *AWS VPC User Guide*.

3.1.6 Upgrading a VPC Instance

Periodically, new releases of the Netgate® pfSense® Plus AMI are issued that may provide new functionality, bug fixes, and security updates. In most cases it is recommended to update via the pfSense® Plus GUI.

> **Tip:** Consult the Upgrade Guide before proceeding with any upgrade.

Warning: pfSense Plus software can no longer run on AWS ".nano" size instances as they lack sufficient RAM to upgrade properly. Attempting to upgrade a ".nano" instance will fail before the upgrade is performed. Migrate the instance to a ".micro" or larger size **before** attempting to upgrade, or redeploy instead.

Before upgrading, back up the configuration of the existing instance by navigating to **Diagnostics > Backup/Restore** in the GUI. Click the **Download configuration** button under the **Backup Configuration** heading and save the config file to a local system.

Next, navigate to **System > Update** to perform the update.

Note: If issues arise with the upgrade process, or there is a need to bring up a new instance alongside the existing one to execute a cutover, follow the instructions below. These instructions detail the procedure for moving an existing instance to one running the latest version.

1. Save a backup configuration, as mentioned above, and write down the NDI located on the pfSense® Plus dashboard, of the current pfSense® Plus AMI.
2. Bring up a new instance of the pfSense® Plus AMI running the latest version.
3. When creating the instance, make sure the interfaces match the interfaces on the existing instance. Make sure that the new instance is in the same VPC as the existing instance and that it has the same number of interfaces attached and that the interfaces are connected to the same Subnets.
4. Make sure any interfaces on the new instance that will communicate with private Subnets have the Source/Destination check disabled.
5. Allocate a new Elastic IP and associate it to the WAN interface of the new instance to allow management access.
6. Restore the backed up configuration file to the new instance. Navigate to **Diagnostics > Backup/Restore** in the GUI. Under the **Restore Configuration** heading, click the **Choose File** button and browse for the configuration file backed up from the existing instance earlier. Once that file is selected, click the **Restore configuration** button. The configuration file will be uploaded, and the instance will reboot automatically.
7. If the old instance had packages installed, navigate to **Packages** under the **System** menu in the pfSense® Plus GUI and install the same packages.
8. If there was any external dependency on the public IP address of the existing instance, remove the Elastic IP Address from the upgraded instance and move the Elastic IP Address from the existing instance to the upgraded instance. External dependencies that might necessitate this include things like VPNs configured on external devices that rely on the existing instance Elastic IP address, or access lists on external devices that allow access to traffic from the existing instance's IP address. There may be other reasons to keep the existing address as well (to preserve existing bookmarks to the GUI, reduce the need for updates to existing internal documentation, etc.). The process for moving the old Elastic IP address to the new instance is as follows:
    - Disassociate the Elastic IP address from the new instance. In the EC2 Management Console, click on **Elastic IPs** under the **Network & Security** heading. Check the box next to the Elastic IP address assigned to the new instance and click on the **Disassociate Address** button.
    - Disassociate the Elastic IP address from the old instance. The procedure is the same as in the previous step, repeated for the Elastic IP address of the old instance this time.
    - Associate the Elastic IP address that was previously associated to the old instance to public interface of the new instance. In the EC2 Management Console, click on **Elastic IPs**. Check the box next to the Elastic IP address being moved and click the **Associate Address** button. Fill in the correct value for the Instance or Network Interface and select the Private IP Address of the public interface on the new instance. Click the **Associate** button. The management interface of the new instance should now be accessible.

9. Move any default routes that pointed to an interface on the old instance to point to the equivalent interface on the new instance. In the VPC Management Console, click on **Route Tables** under the **Virtual Private Cloud** heading. Check the box next to a Route Table associated with the VPC that the instances is located in.
    - In the detail pane that appears at the bottom of the screen, click on the **Routes** tab.
    - If a route exists for `0.0.0.0/0` with a Target that is an interface ID of an interface on the old instance, click the **Edit** button above the table displaying the routes.
    - Click the red **X** next to the row for `0.0.0.0/0` to remove the existing route.
    - There should be a blank row with empty fields for a new route. Enter `0.0.0.0/0` in the **Destination** field and the Network Interface ID of the interface on the new instance in the Target field. Click on the **Save** button.
    - If there were multiple private subnets in the VPC which were pointed to interfaces on the pfSense® Plus instance, repeat this process for the other Route Tables associated with the VPC.
    - The new instance should now be functioning as the old one did.

CHAPTER

FOUR

REFERENCES

4.1 Regional Market Availability

The tables below represent the current availability by regional market. If the desired regional market is not listed, refer to the AWS Regions availability or submit a support ticket directly to AWS.

Table 1: AWS Available Regions

Market	Availability
us-east-1 N. Virginia	Available
us-east-2 Ohio	Available
us-west-1 N. California	Available
us-west-2 Oregon	Available
ca-central-1 Quebec	Available
eu-central-1 Frankfurt	Available
eu-west-1 Ireland	Available
eu-west-2 London	Available
eu-west-3 Paris	Available
eu-north-1 Stockholm	Available
eu-south-1 Milan	Available
ap-east-1 Hong Kong	Available
ap-southeast-1 Singapore	Available
ap-southeast-2 Sydney	Available
ap-northeast-2 Seoul	Available
ap-northeast-1 Tokyo	Available
ap-northeast-3 Osaka	Available
ap-south-1 Mumbai	Available
sa-east-1 São Paulo	Available
af-south-1 Cape Town	Available
us-gov-east-1 GovCloud East	Available
us-gov-west-1 GovCloud West	Available

4.2 Assistance Resources

4.2.1 Commercial Assistance

To keep prices low, the software is not bundled with a TAC subscription.

Netgate TAC options:

	TAC Pro	TAC Enterprise
TAC Availability	24/7	24/7
Target Initial Response SLA	24 Hour	4 Hour
Email / TAC Portal	Yes	Yes
Telephone Assistance	No	Yes

For more information and purchasing, see: https://www.netgate.com/support.

4.2.2 Community Assistance

Community assistance is available through the Netgate Forum.

4.3 Additional Resources

4.3.1 Netgate Training

Netgate training offers training courses for increasing your knowledge of pfSense® Plus products and services. Whether you need to maintain or improve the security skills of your staff or offer highly specialized assistance and improve your customer satisfaction; Netgate training has got you covered.

https://www.netgate.com/training

4.3.2 Resource Library

To learn more about how to use Netgate appliances and for other helpful resources, make sure to browse the Netgate Resource Library.

https://www.netgate.com/resources

4.3.3 Professional Services

Netgate TAC does not cover more complex tasks such as CARP configuration for redundancy on multiple firewalls or circuits, network design, and conversion from other firewalls to pfSense® Plus software. These items are offered as professional services and can be purchased and scheduled accordingly.

https://www.netgate.com/our-services/professional-services.html

4.3.4 Community Options

Customers who elected not to get a Netgate TAC plan, can seek assistance from the active and knowledgeable pfSense software community on the Netgate forum.

https://forum.netgate.com/