Table of Contents

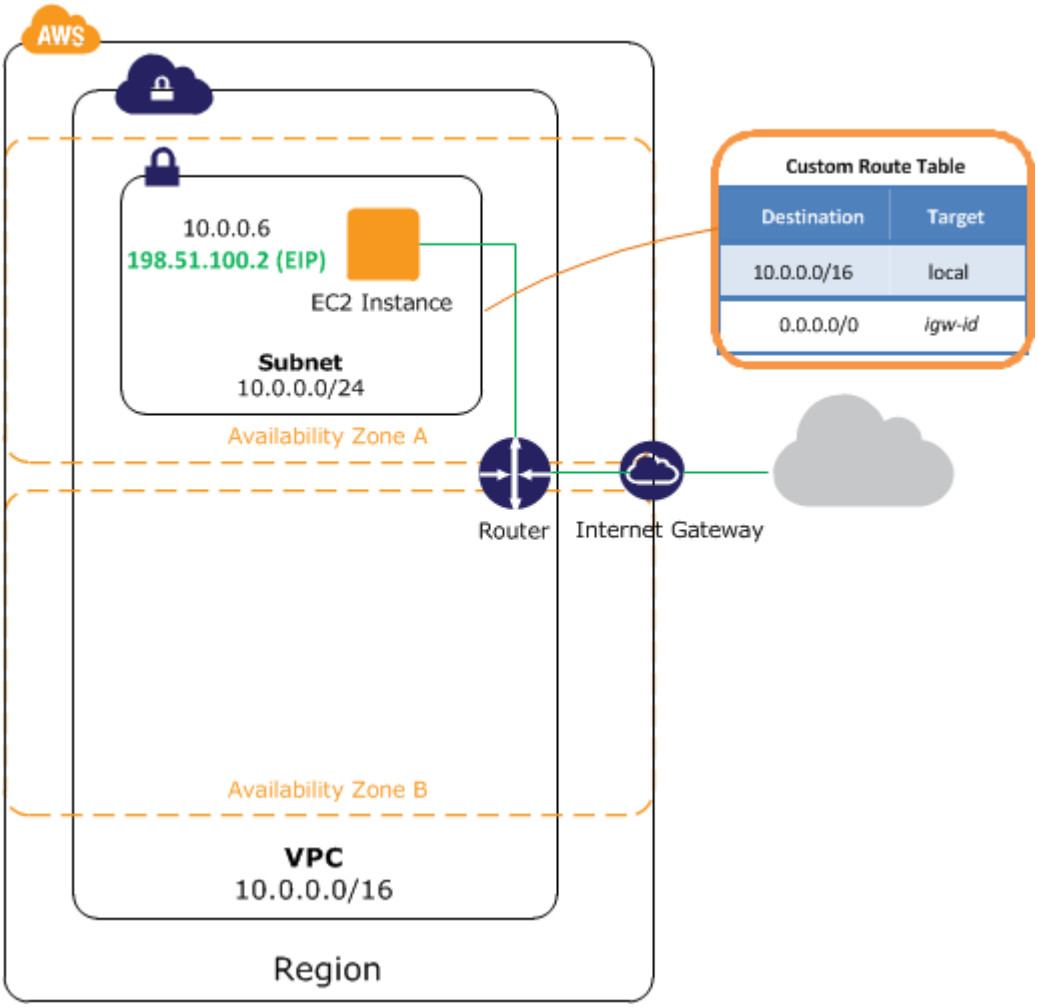
Prepared by

A. N. Anil Kumar
[cloud.b.lab@zoho.com]

This tutorial provides a hands-on introduction to using Amazon VPC through the AWS Management Console.

The exercise in this tutorial walks you through a simple scenario in which you set up a VPC with a single public subnet containing a running EC2 instance with an Elastic IP address.

What to achieve? -VPC with a Public Subnet Only



Topics

- Step 1: Sign up for Amazon VPC
- Step 2: Set Up the VPC and Internet Gateway
- Step 3: Set Up a Security Group for Your VPC
- Step 4: Launch an Instance into Your VPC
- Step 5: Assign an Elastic IP Address to Your Instance

Step 1: Sign up for Amazon VPC

When you create an AWS account, we automatically sign up your account for all AWS services, including Amazon EC2 and Amazon VPC. You pay only for the services that you use. For this example, the charges will be minimal.

If you have an AWS account already, skip to the next step. If you don't have an AWS account, use the following procedure to create one.

To create an AWS account

1. Go to http://aws.amazon.com and click **Sign Up Now**.
2. Follow the on-screen instructions.

   Part of the sign-up process involves receiving a phone call and entering a PIN using the phone keypad.

We'll notify you by email when your account is active and available for you to use.

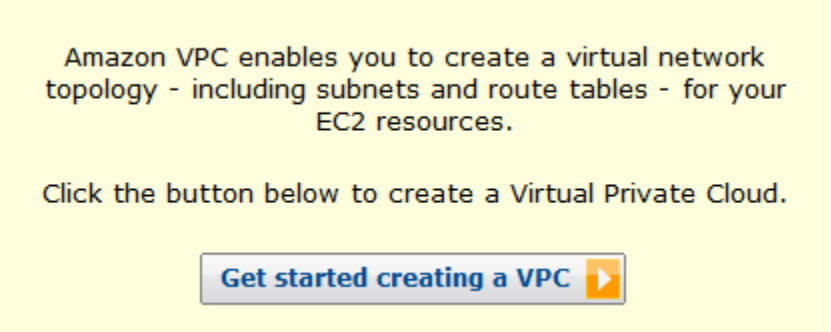
Step 2: Set Up the VPC and Internet Gateway

In this step, we'll use the VPC wizard to create a VPC. The wizard performs the following steps for you:

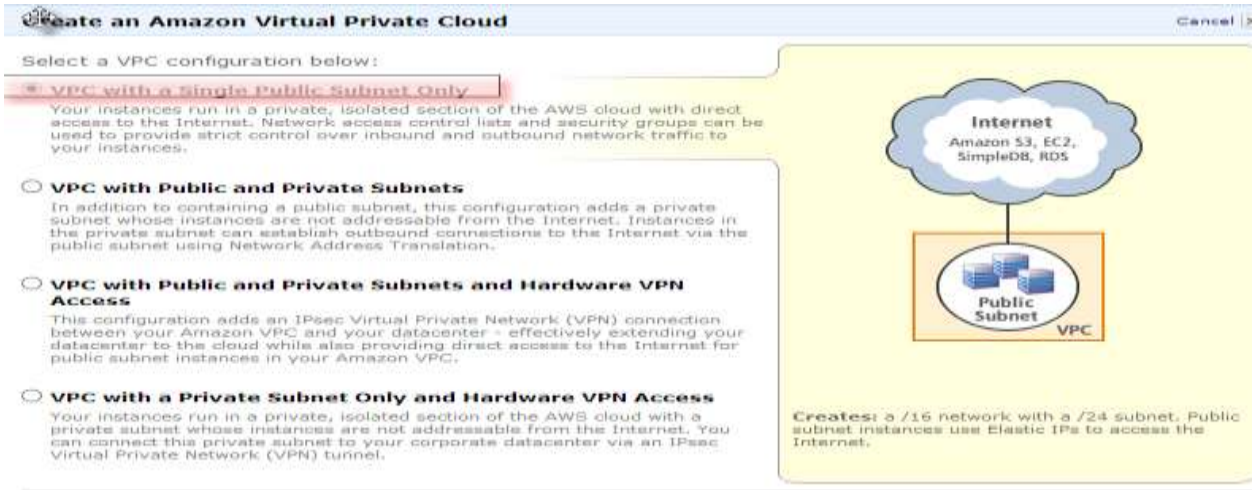
- Create a size /16 VPC (a network with [2^{16}] 65,536 private IP addresses).
- Attach an Internet gateway to the VPC.
- Add a size /24 subnet (a range of 256 private IP addresses).
- Set up routing for your VPC so that traffic can flow between the subnet and the Internet gateway.

To create a VPC using the VPC Wizard in the AWS Management Console

1. Sign in to the AWS Management Console and open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. On the VPC dashboard, click **Get started creating a VPC**.



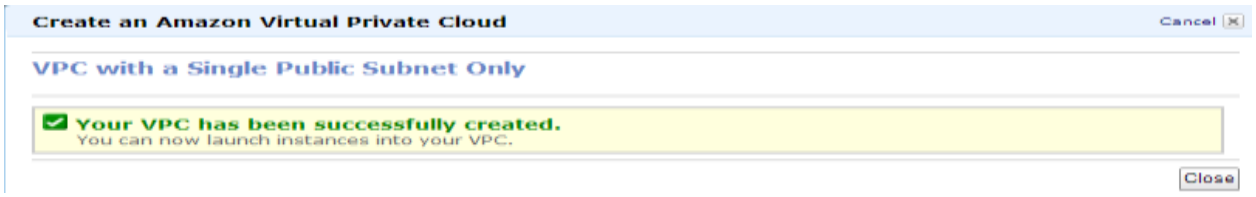
Select the first option, **VPC with a Single Public Subnet Only**, and click **Continue**.



The confirmation page shows the CIDR ranges that we'll use for your VPC and subnet (10.0.0.0/16 and 10.0.0.0/24, respectively), and the hardware tenancy setting. The confirmation page also displays the subnet's

Availability Zone. Make any changes to these settings that you need, and then click **Create VPC** to create your VPC, Internet gateway, subnet, and route table.

A status window shows the work in progress. When the work completes, a status window confirms that your VPC has been successfully created. Click**Close** to close the status window and return to the VPC dashboard.



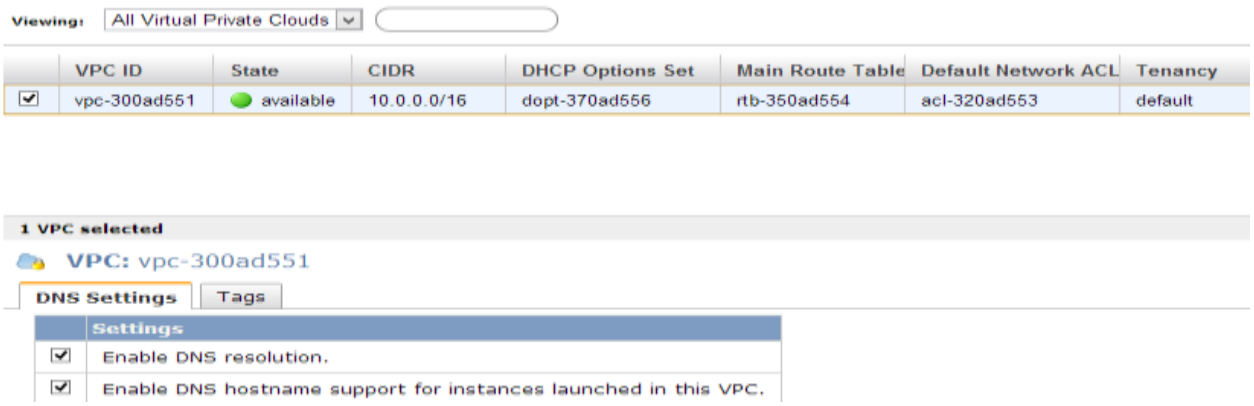
Verify Your VPC's Components

You can use the default settings for the components that the VPC Wizard created for you as you go through the exercise in this guide. This section describes how you can view these components and their settings using the VPC console.

In the navigation pane, click **Your VPCs**, and then select the VPC that you just created (**Default VPC** is `false`).

The console displays your default VPC and the VPC that you just created. Each VPC has a set of DHCP options, a main route table, and default network ACL.

Select the Checkbox and the console displays the DNS settings for the VPC in the details pane.



For your extra knowledge

Using DNS with Your VPC

Amazon EC2 instances need IP addresses to communicate. Public IP addresses enable communication over the Internet, while private IP addresses enable communication within the network of the instance (EC2-Classic or a VPC). To enable an EC2 instance to be publicly accessible, it must have a public IP address, a DNS hostname, and DNS resolution.

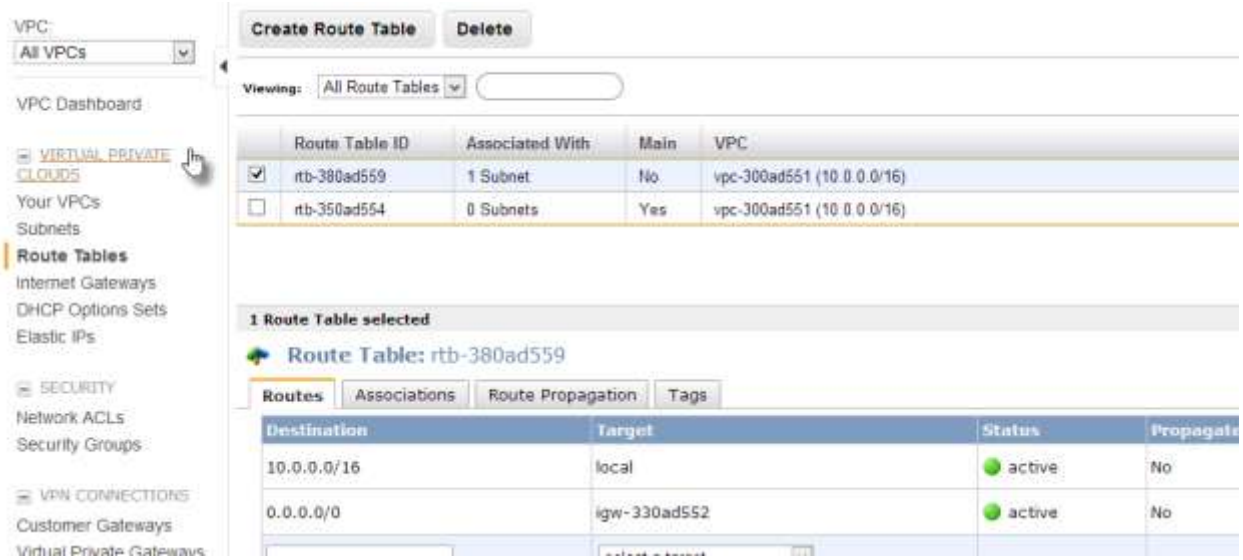
Domain Name System (DNS) is a standard by which names used on the Internet are resolved to their corresponding IP addresses. A DNS hostname is a name that uniquely and absolutely names a computer; it's composed of a host name and a domain name. DNS servers resolve DNS hostnames to their corresponding IP addresses.

To display information about your Internet gateways, click **Internet Gateways** in the navigation pane. You have one Internet gateway for your default VPC, and another for the VPC that you just created.

The VPC that you just created has two route tables. The VPC came with a main route table by default, and the VPC Wizard created a custom route table in addition. Your subnet is associated with the custom route table, which means that we use the routes in that table to determine how the traffic for the subnet flows. If you add a new subnet to your VPC, it uses the main route table by default.

To view your route tables

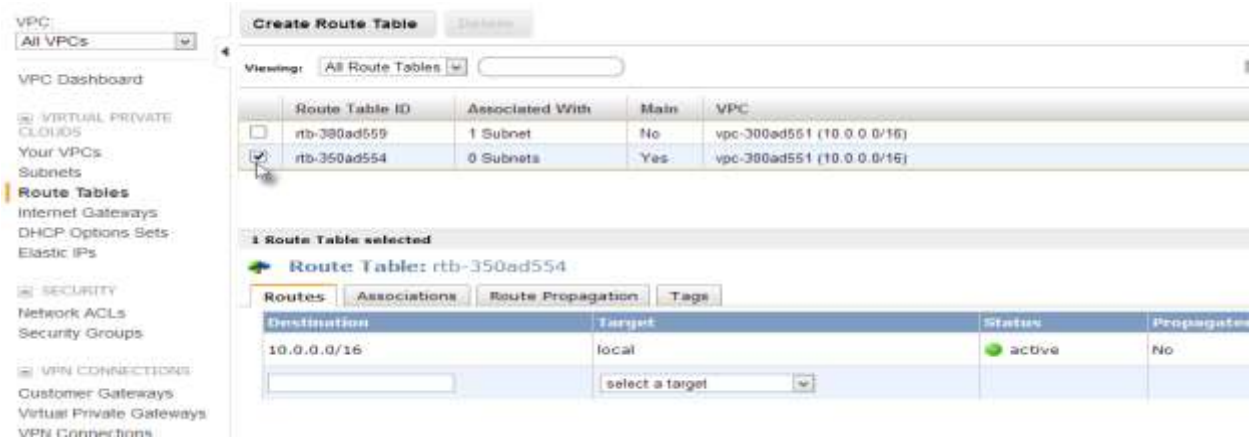
1. Click **Route Tables** in the navigation pane.
2. Select the custom route table (the **Main** column has **No**) to display the route information in the details pane.



The first row in the above table is the local route, which enables instances within the VPC to communicate. This route is present in the every route table by default, and you can't remove it.

The second row shows the route that the VPC wizard added to enable traffic destined for an IP address outside the VPC (0.0.0.0/0) to flow from the subnet to the Internet gateway. We refer to this subnet as a *public subnet* because all traffic from the subnet goes to the Internet gateway.

Select the main route table(the **Main** column has). The main route table has a local route, but no other routes. Therefore, any subnet you create is not exposed to the Internet initially, it's a *private subnet*. To expose a new subnet as a public subnet, you can either change the routing in the main route table, or associate the subnet with a custom route table.



Step 3: Set Up a Security Group for Your VPC

A *security group* acts as a virtual firewall to control the traffic allowed into its associated instances. To use security groups, you create a group, add the inbound and outbound rules that you want to use, and then associate your instances with the security group when you launch them. If you add and remove rules from the security group, we apply those changes to the instances associated with the security group automatically.

Your VPC comes with a *default security group*. Any instance not associated with another security group is associated with the default security group. Although we could use the default security group for this exercise, we've chosen to create a security group, `WebServerSG`, instead. You'll specify this security group when you launch an instance into your VPC.

Rules for the WebServerSG Security Group

Inbound rules regulate the traffic that is allowed to reach the instances associated with the security group (the source of the traffic and the listening port on the instance). All return traffic is automatically allowed to reach the instances. For example, if a client on the Internet sends a request to a web server in your VPC associated with `WebServerSG`, the instance can respond, regardless of any outbound rules on the group. In this way, security groups are stateful.

Outbound rules control which destinations the instances associated with the security group can send traffic to (the destination of the traffic and the destination port). All return traffic (such as a response from the host that received the traffic) is automatically allowed to reach the instances, regardless of the inbound rules set on the security group.

The following table describes the inbound rules for the `WebServerSG` security group. Because the web server doesn't initiate outbound communication, we'll remove the default outbound rule.

Note

If your company uses only Linux or only Windows, you don't have to add access for both SSH and RDP.

Inbound			
Source IP	**Protocol**	**Port Range**	**Comments**
0.0.0.0/0	TCP	80	Allow inbound HTTP access from anywhere
0.0.0.0/0	TCP	443	Allow inbound HTTPS access from anywhere
Public IP address range of your home network	TCP	22	Allow inbound SSH access from your home network (Linux/UNIX only)
Public IP address range of your home network	TCP	3389	Allow inbound RDP access from your home network (Windows only)

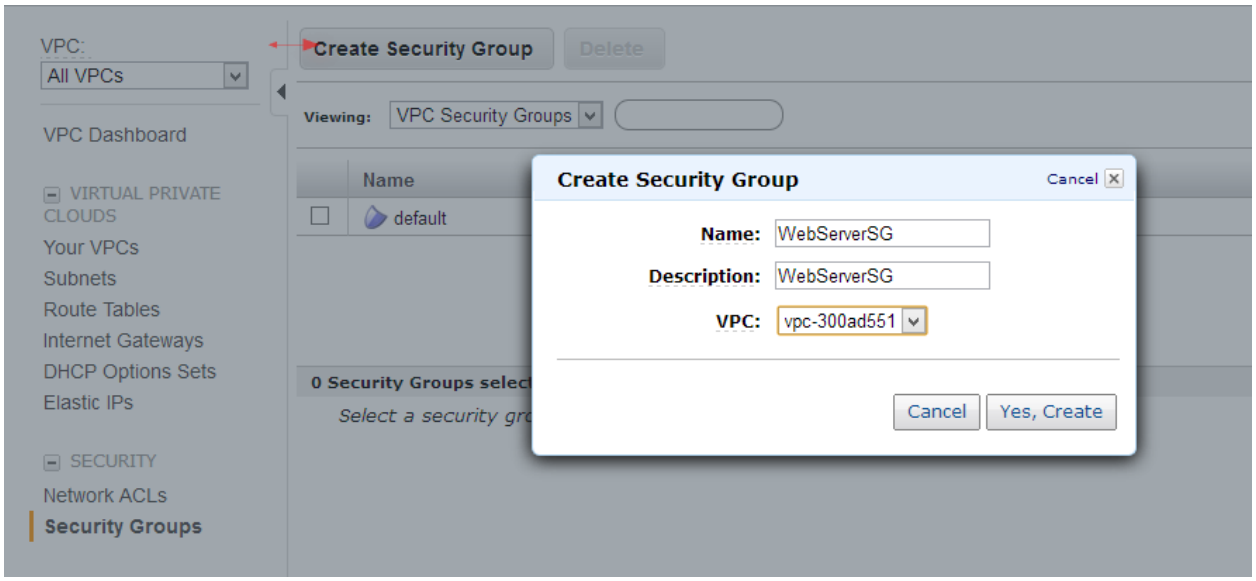
In this exercise, you won't add a rule to enable instances associated with the security group to talk to each other. To enable this type of communication, you must add a rule to the security group for this purpose.

Creating Your WebServerSG Security Group

To create the WebServerSG security group

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. Click **Security Groups** in the navigation pane.
3. Click the **Create Security Group** button.
4. Enter `WebServerSG` as the name of the security group, and provide a description. Select the ID of your VPC from the **VPC** menu, and then click **Yes, Create**.

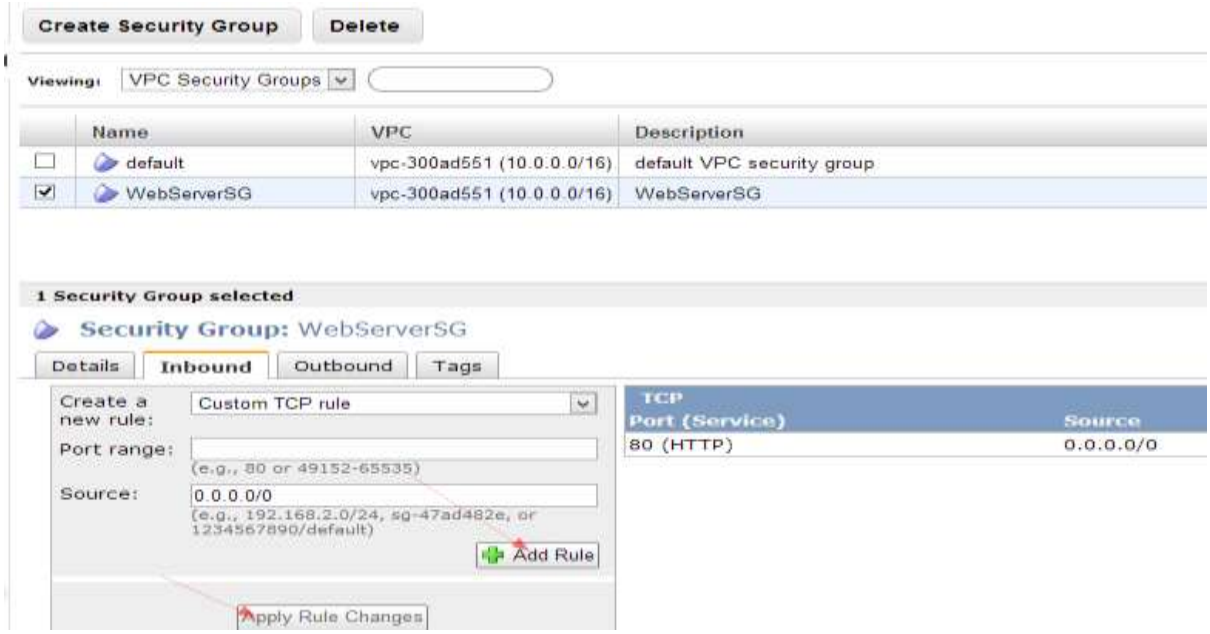
By default, new security groups start with only an outbound rule that allows all traffic to leave the instances. You must add rules to enable any inbound traffic or to restrict the outbound traffic.



Adding Rules to Your WebServerSG Security Group

To add rules to the WebServerSG security group

1. Click **Security Groups** in the navigation pane to display your security groups.
2. Select the `WebServerSG` security group that you just created. The details pane include a tab for information about the security group, plus tabs for working with its inbound rules and outbound rules.
3. Add rules for inbound HTTP and HTTPS access from anywhere:
   a. On the **Inbound** tab, select `HTTP` from the **Create a new rule** drop-down list and make sure that **Source** is `0.0.0.0/0`.
   b. Click **Add Rule**. This adds a rule to allow HTTP access from anywhere. Notice that the **Apply Rule Changes** button is enabled, and the text "Your changes have not been applied yet" appears above the button. We'll click this button to apply the rule changes after we've added all the inbound rules.



Similarway:

a. Select `HTTPS` from the **Create a new rule** drop-down list and make sure that **Source** is `0.0.0.0/0`.

b. Click **Add Rule**. This adds a rule to allow HTTPS access from anywhere.

Add rules for inbound SSH and Remote Desktop (RDP) access to the group from your network's public IP address range:

Caution

If you use `0.0.0.0/0`, you enable all IP addresses to access your instance using SSH or RDP. This is acceptable for the short exercise, but it's unsafe for production environments. In production, you'll authorize only a specific IP address or range of addresses to access your instance.

Tip

You can also get the public IP address of your local computer using a service. To locate a service that provides your IP address, use the search phrase "what is my IP address". If you are connecting through an ISP or from behind a firewall without a static IP address, you need to find the range of IP addresses used by client computers.

a. On the **Inbound** tab, select `SSH` from the **Create a new rule** drop-down list.

b. In the **Source** field, enter your network's public IP address range (for example, `192.0.2.0/24`). If you don't know this address range, you can use`0.0.0.0/0` for this exercise (see the Caution and Tip for this step).

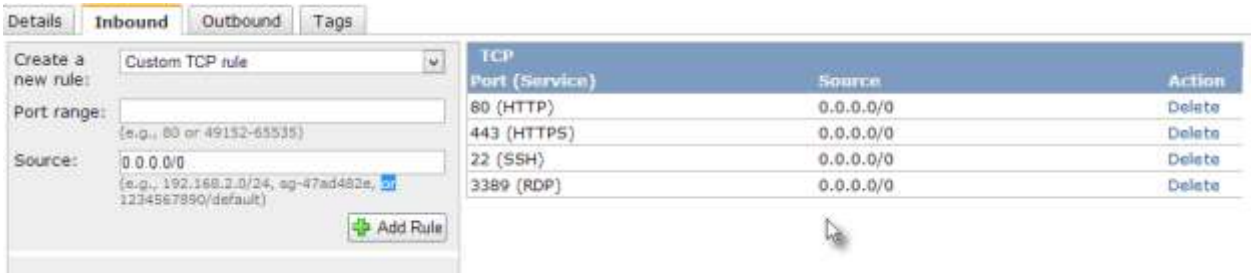
c. Click **Add Rule**.

d. Select `RDP` from the **Create a new rule** drop-down list.

e. In the **Source** field, enter your home network's public IP address range. If you don't know this address range, you can use `0.0.0.0/0` for this exercise (see the Caution and Tip for this step).

f. Click **Add Rule**.

g. Click **Apply Rule Changes** to apply these inbound rules.



Limit outbound access to responses by removing the default outbound rule:

a. On the **Outbound** tab, locate the default rule that enables all outbound traffic, and click **Delete**. The rule is marked for deletion.

b. Click **Apply Rule Changes**.

Step 4: Launch an Instance into Your VPC

When you launch an EC2 instance into a VPC, you must specify the ID of a subnet in the VPC.

To launch an EC2 instance into a VPC

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.

2. From the navigation bar, select the region for the instance. For this exercise, you can use the default region. For more information about regions and Availability Zones, see Regions and Availability Zones in the *Amazon Elastic Compute Cloud User Guide*.

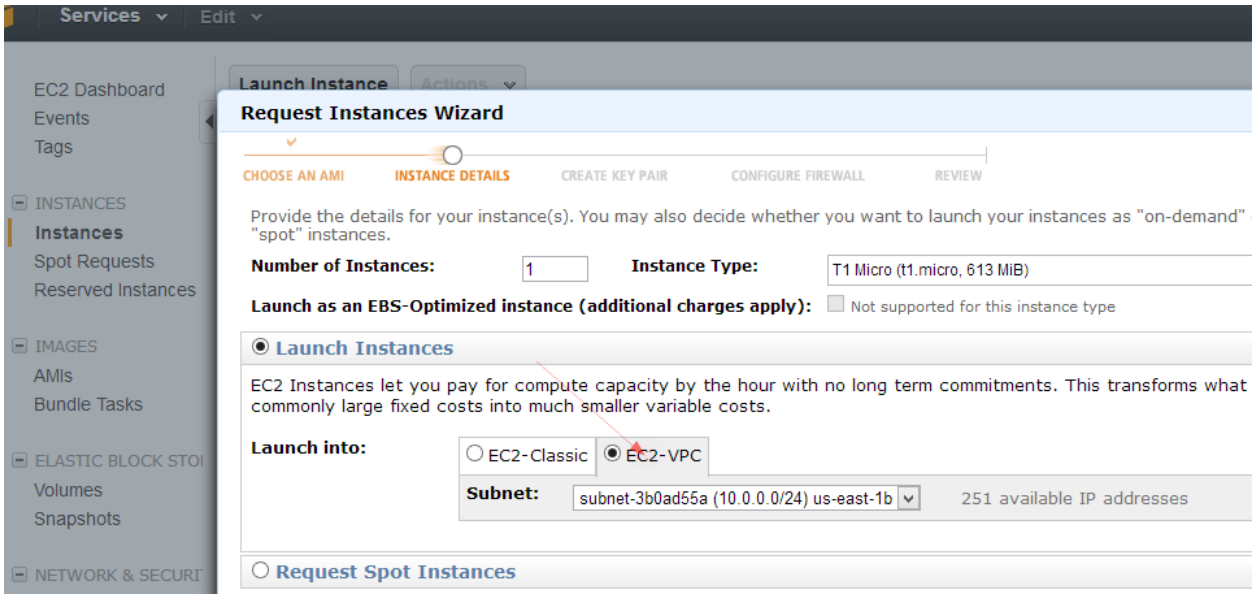
3. From the dashboard, click the **Launch Instance** button.

4. On the **Create a New Instance** page, select **Classic Wizard**, and then click **Continue**.

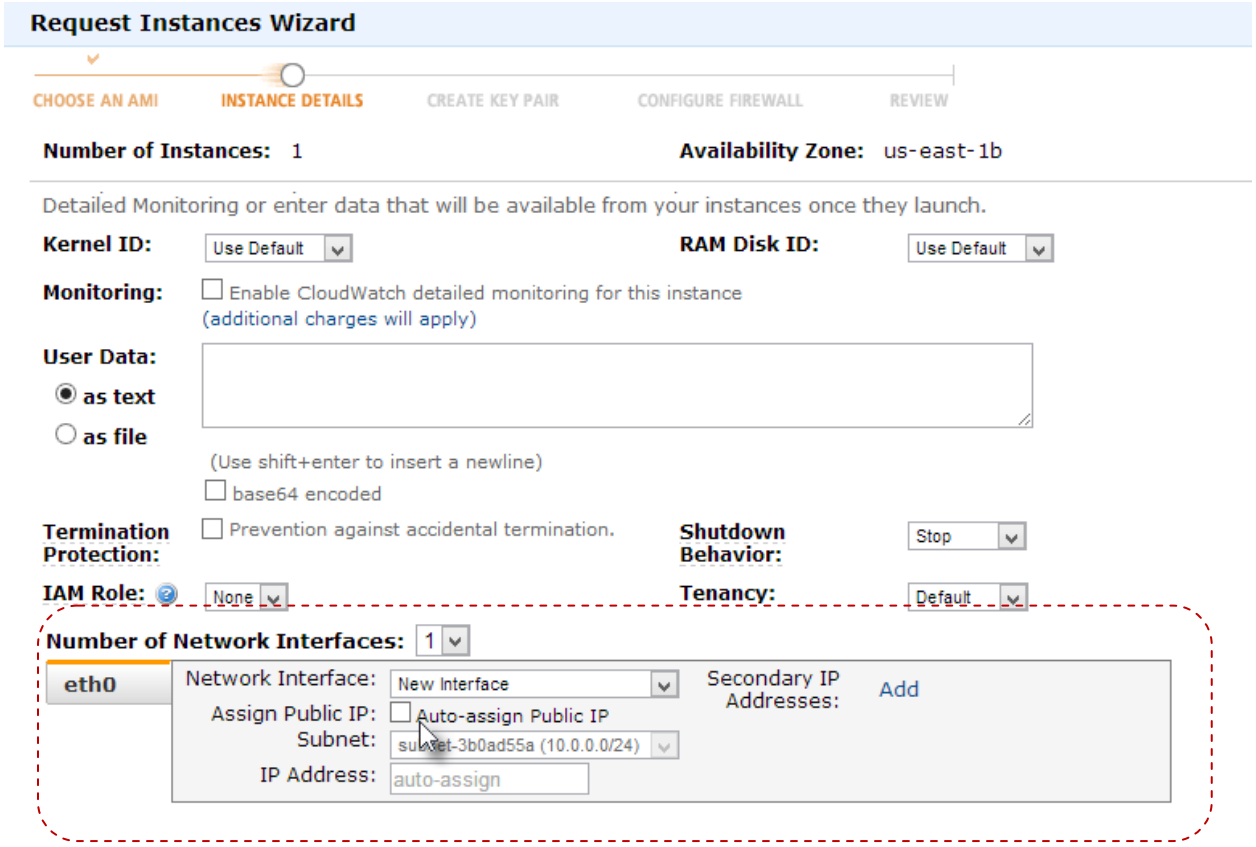
5. On the **CHOOSE AN AMI** page, the **Quick Start** tab displays a list of basic configurations called Amazon Machine Images (AMI). Choose the AMI that you want to use and click its **Select** button.

6. On the **INSTANCE DETAILS** page, in the **Instance Type** menu, leave the default value, **Micro (t1.micro)**, to launch a single micro instance.

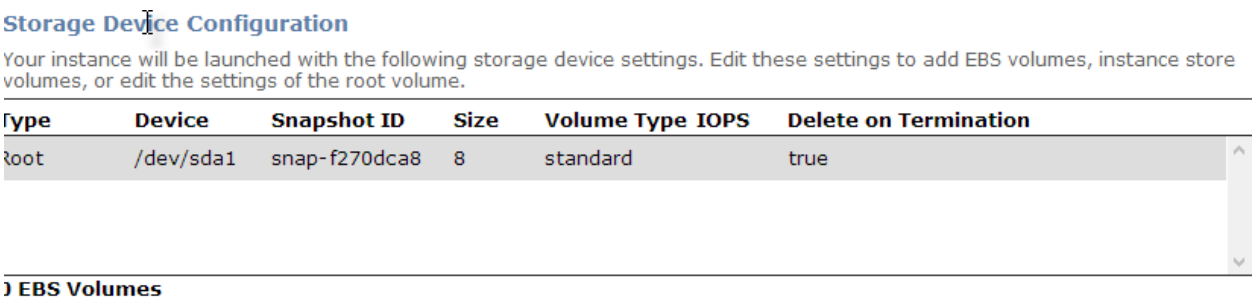
7. Under **Launch Instances**, confirm that your subnet is selected in the **Subnet** drop-down list box, and then click **Continue**.



Under **Advanced Instance Options**, you can specify the private IP address to use for the instance. You can also request that your instance receives a public IP address, as instances launched into a nondefault subnet are not assigned one by default. For this exercise, however, we'll leave **IP Address** empty and click **Continue** to accept the default settings.



Click **Continue** to use the default storage device



Specify any tags you'd like for your instance, and click **Continue**.

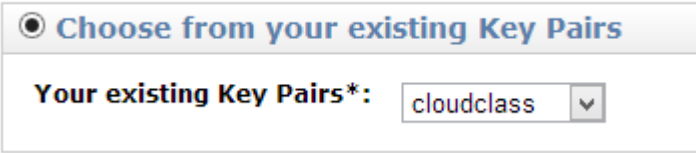
1. On the **Create Key Pair** page, you can select an existing key pair or create a new one. For this exercise, we'll create a key pair.

    a. Click **Create a new Key Pair** or **Choose an existing one if you have**

    To create a new one:

    b. Enter a name for your key pair (for example, `VPC_Keypair`), and then click **Create & Download your Key Pair**. You need the contents of the private key to connect to your instance after it's launched. Amazon Web Services doesn't keep the private portion of key pairs.

    c. When prompted, save the private key in a safe place on your system, and click **Continue**.



1. On the **CONFIGURE FIREWALL** page, select **Choose one or more of your existing Security Groups**. Select the `WebServerSG` group that you created previously, and then click **Continue**.

2. On the **REVIEW** page, review your settings. When you're satisfied with your selections, click **Launch** to launch your instance.

Step 5: Assign an Elastic IP Address to Your Instance

By default, an instance in a nondefault VPC is not assigned a public IP address, and is private. You can make an instance in a nondefault VPC public by attaching an Internet gateway to the VPC and providing the instance with a public IP address. In this exercise, you created an Internet gateway for your VPC using the VPC wizard. You accepted the default settings on the second **INSTANCE DETAILS** page in the launch wizard, so you did not receive a public IP address. Now, you'll create an Elastic IP address, which is a public IP address that belongs to your AWS account, and associate it with your instance to make it accessible from the Internet.

For more information about Elastic IP addresses, see Elastic IP Addresses in the *Amazon Virtual Private Cloud User Guide*.

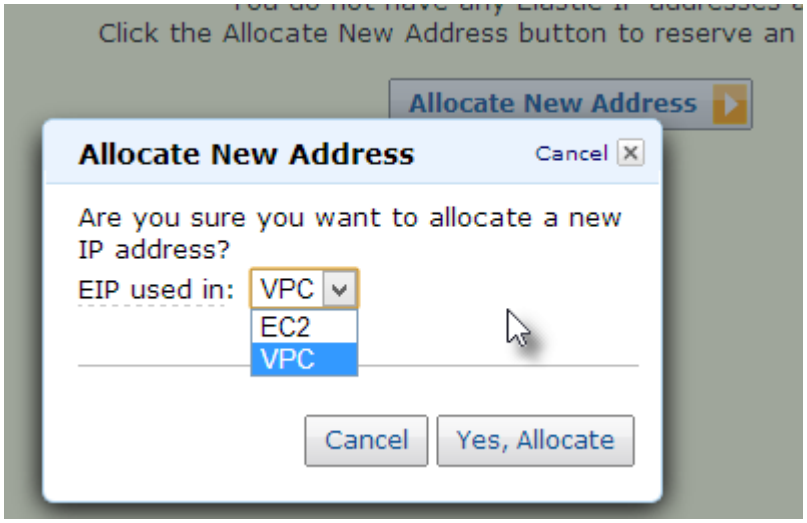
To allocate and assign an Elastic IP address

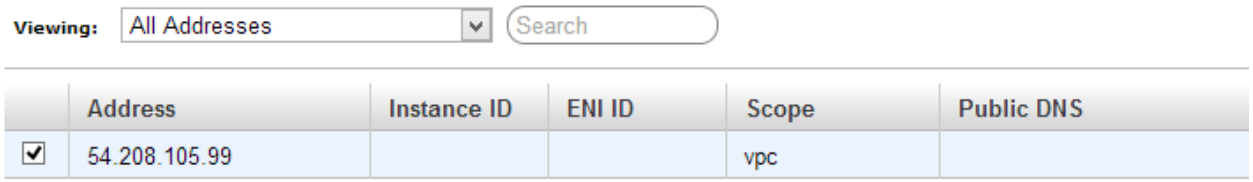
1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.

2. Click **Elastic IPs** in the navigation pane.

3. Click the **Allocate New Address** button.

4. In the **EIP used in** list, select `VPC`, and then click **Yes, Allocate**.



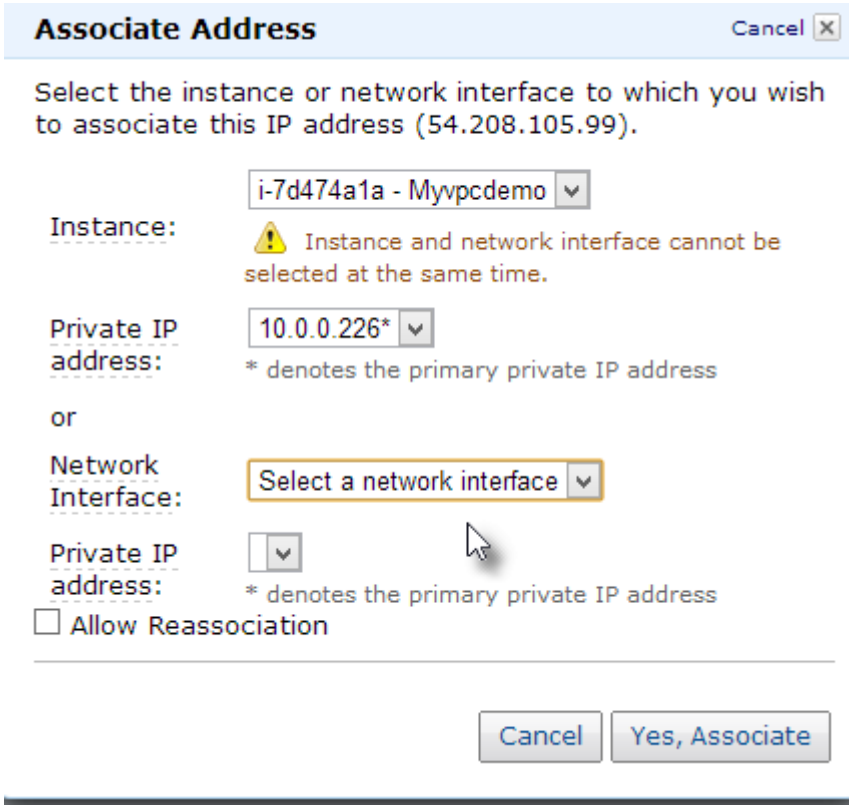


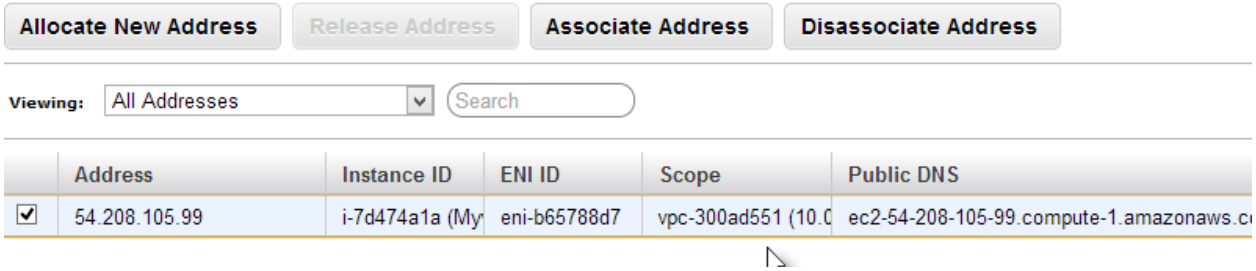
1. Select the Elastic IP address from the list and click the **Associate Address** button.

2. In the **Associate Address** dialog box, do the following, and then click **Yes, Associate**:

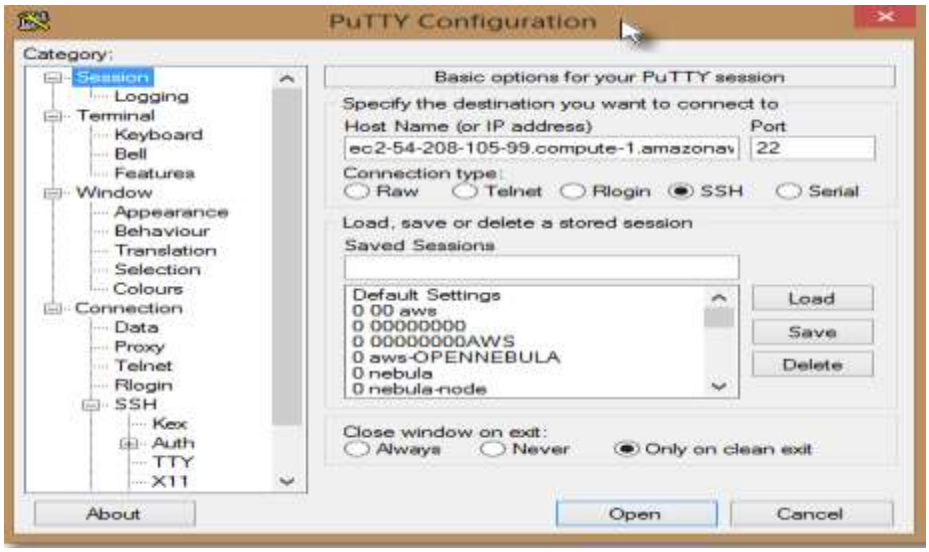
    a. Select the network interface from the **Network Interface** list, or select the instance from the **Instance** list. Note that the advantage of making the Elastic IP address as an attribute of the network interface instead of associating it directly with the instance is that you can move all the attributes of the network interface from one instance to another in a single step.

    b. Select the IP address to associate the EIP with from the corresponding **Private IP address** list.





Our instance is now accessible from the Internet. You can also access the instance using SSH or Remote Desktop from your home network, specifying the Elastic IP address of the instance as the address to connect to



That’s it you have VPC and an instance running in it.

Additional Information: [For extra reading]

Overview of DHCP Options Sets

The Dynamic Host Configuration Protocol (DHCP) provides a standard for passing configuration information to hosts on a TCP/IP network. The `options` field of a DHCP message contains the configuration parameters. Some of those parameters are the domain name, domain name server, and the netbios-node-type.

DHCP options sets are associated with your AWS account so that you can use them across all of your virtual private clouds (VPC).

The Amazon EC2 instances you launch into a nondefault VPC are private by default; they're not assigned a public IP address unless you specifically assign one during launch. By default, all instances in a nondefault VPC receive an unresolvable host name that AWS assigns (for example, ip-10-0-0-202). You can assign your own domain name to your instances, and use up to four of your own DNS servers. To do that, you must specify a special set of DHCP options to use with the VPC. This set can contain other commonly used DHCP options (see the following table for the full list of supported options).

DHCP Option Name	Description
domain-name-servers	The IP addresses of up to four domain name servers, or AmazonProvidedDNS. The default DHCP option set specifies AmazonProvidedDNS.
domain-name	If you're using AmazonProvidedDNS in US East (Northern Virginia) Region, specify compute-1.amazonaws.com. If you're using AmazonProvidedDNS in another region, specify *region*.compute.amazonaws.com. Otherwise, specify a domain name (for example,`MyCompany.com`).
ntp-servers	The IP addresses of up to four Network Time Protocol (NTP) servers.
netbios-name-servers	The IP addresses of up to four NetBIOS name servers.
netbios-node-type	The NetBIOS node type (1, 2, 4, or 8). We recommend that you specify 2 (broadcast and multicast are not currently supported).

Amazon DNS Server

When you create a VPC, we automatically create a set of DHCP options and associate them with the VPC. This set includes only a single option: `domain-name-servers=AmazonProvidedDNS`. This is an Amazon DNS server, and this option enables DNS for instances that need to communicate over the VPC's Internet gateway. The string `AmazonProvidedDNS` maps to a DNS server running on a reserved IP address at the base of the VPC network range "plus two". For example, the DNS Server on a 10.0.0.0/16 network is located at 10.0.0.2.

Note

You can also use the Amazon DNS server IP address 169.254.169.253, though some servers don't allow its use. Windows Server 2008, for example, disallows the use of a DNS server located in the 169.254.x.x network range.

Changing DHCP Options

After you create a set of DHCP options, you can't modify them. If you want your VPC to use a different set of DHCP options, you must create a new set and associate them with your VPC. You can also set up your VPC to use no DHCP options at all.

You can have multiple sets of DHCP options, but you can associate only one set of DHCP options with a VPC at a time. If you delete a VPC, the DHCP options set associated with the VPC are also deleted.

After you associate a new set of DHCP options with a VPC, any existing instances and all new instances that you launch in the VPC use these options. You don't need to restart or relaunch the instances. They automatically pick up the changes within a few hours, depending on how frequently the instance renews its DHCP lease. If you want, you can explicitly renew the lease using the operating system on the instance.

Working with DHCP Options Sets

Creating a DHCP Options Set

You can create as many additional DHCP options sets as you want. However, you can only associate a VPC with one set of DHCP options at a time. After you create a set of DHCP options, you must configure your VPC to use it. For more information, see Changing the Set of DHCP Options a VPC Uses.

To create a DHCP options set

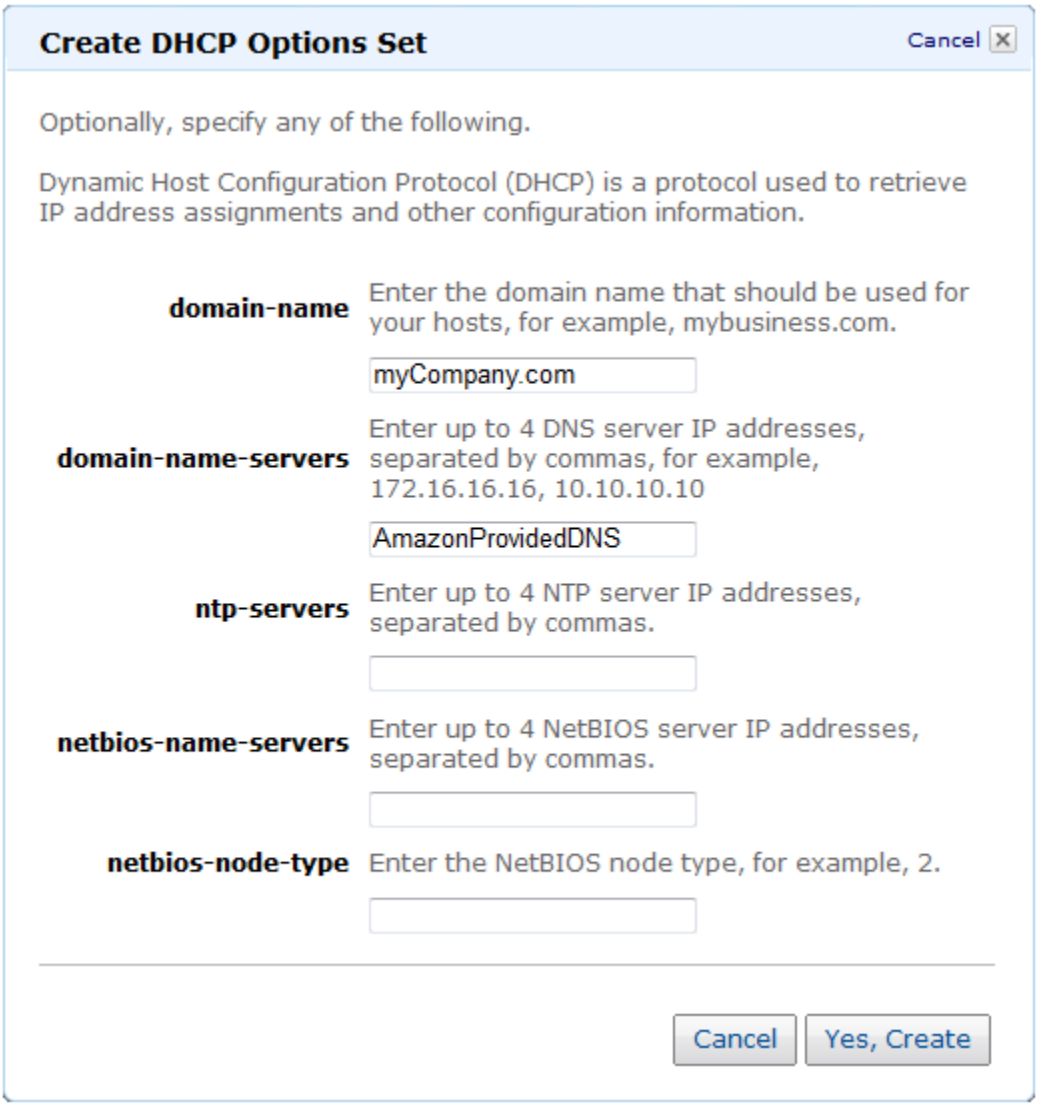
1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.

2. Click **DHCP Options Sets** in the navigation pane, and then click the **Create DHCP Options Set** button.

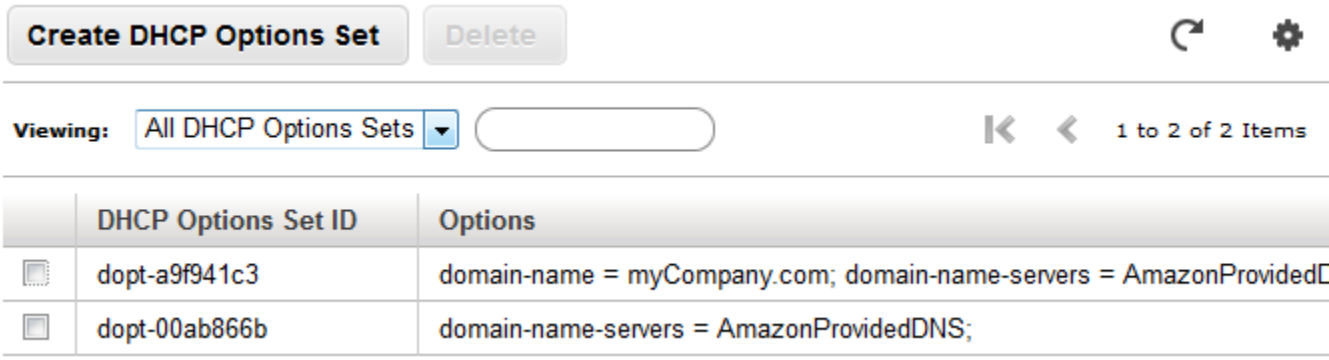
3. In the **Create DHCP Options Set** dialog box, enter values for the options that you want to use, and then click **Yes, Create**.

   Important

   If your VPC has an Internet gateway, make sure to specify your own DNS server or Amazon's DNS server (AmazonProvidedDNS) for the **domain-name-servers** value. Otherwise, the instances that need to communicate with the Internet won't have access to DNS.



   The new set of DHCP options appears in your list of DHCP options. The following image shows an example of the list, with both the set of DHCP options you just created and the set that automatically came with your VPC (where the only option is domain-name-servers=AmazonProvidedDNS).



4. Make a note of the ID of the new set of DHCP options (dopt-*XXXXXXXX*). You will need it to associate the new set of options with your VPC.

Although you've created a set of DHCP options, you must associate it with your VPC for the options to take effect. You can create multiple sets of DHCP options, but you can associate only one set of DHCP options with your VPC at a time.

Changing the Set of DHCP Options a VPC Uses

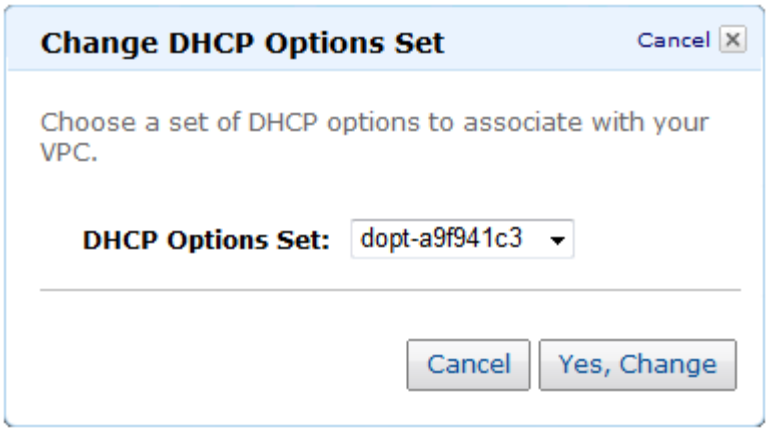
You can change which set of DHCP options your VPC uses. If you want the VPC to use no DHCP options, see Changing a VPC to use No DHCP Options.

Note

The following procedure assumes that you've already created the DHCP options set you want to change to. If you haven't, create the options set now. For more information, see Creating a DHCP Options Set.

To change the DHCP options set associated with a VPC

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. Click **Your VPCs** in the navigation pane.
3. Select the VPC and click the **Change DHCP Options Set** button.
4. In the **Change DHCP Options Set** dialog box, select a set of options from the drop-down list, and then click **Yes, Change**.



After you associate a new set of DHCP options with the VPC, any existing instances and all new instances that you launch in that VPC use the options. You don't need to restart or relaunch the instances. They automatically pick up the changes within a few hours, depending on how frequently the instance renews its DHCP lease. If you want, you can explicitly renew the lease using the operating system on the instance.

Changing a VPC to use No DHCP Options

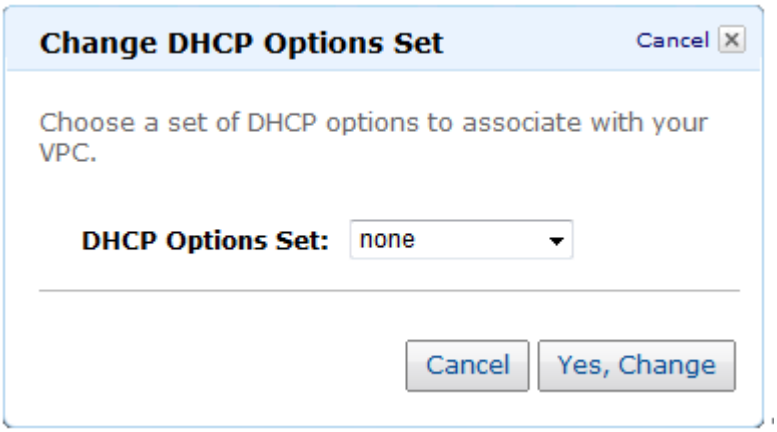
You can set up your VPC to use no set of DHCP options.

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.

2. Click **Your VPCs** in the navigation pane.

3. Select the VPC and click the **Change DHCP Options Set** button.

4. In the **Change DHCP Options Set** dialog box, select **none** from the drop-down list, and then click **Yes, Change**.



You don't need to restart or relaunch the instances. They automatically pick up the changes within a few hours, depending on how frequently the instance renews its DHCP lease. If you want, you can explicitly renew the lease using the operating system on the instance.

Deleting a DHCP Options Set

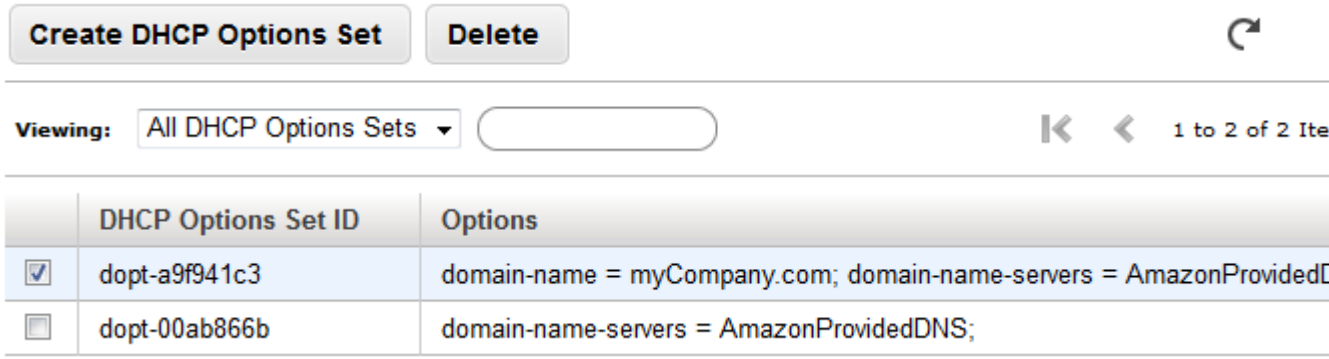
When you no longer need a DHCP options set, use the following procedure to delete it. The VPC must not be using the set of options.

To delete a DHCP options set

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.

2. Click **DHCP Options Set** in the navigation pane.

3. Select the set of DHCP options to delete, and then click **Delete**.



4. In the **Delete DHCP Options Set** dialog box, click **Yes, Delete**.

Other VPC options

Select a VPC configuration below:

○ **VPC with a Single Public Subnet Only**

Your instances run in a private, isolated section of the AWS cloud with direct access to the Internet. Network access control lists and security groups can be used to provide strict control over inbound and outbound network traffic to your instances.

◉ **VPC with Public and Private Subnets**

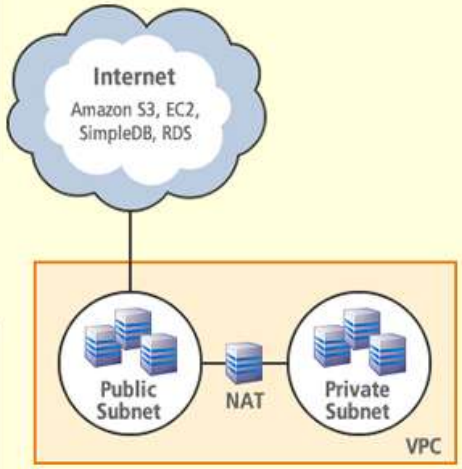
In addition to containing a public subnet, this configuration adds a private subnet whose instances are not addressable from the Internet. Instances in the private subnet can establish outbound connections to the Internet via the public subnet using Network Address Translation.

○ **VPC with Public and Private Subnets and Hardware VPN Access**

This configuration adds an IPsec Virtual Private Network (VPN) connection between your Amazon VPC and your datacenter - effectively extending your datacenter to the cloud while also providing direct access to the Internet for public subnet instances in your Amazon VPC.

○ **VPC with a Private Subnet Only and Hardware VPN Access**

Your instances run in a private, isolated section of the AWS cloud with a private subnet whose instances are not addressable from the Internet. You can connect this private subnet to your corporate datacenter via an IPsec Virtual Private Network (VPN) tunnel.



Creates: a /16 network with two /24 subnets. Public subnet instances use Elastic IPs to access the Internet. Private subnet instances access the Internet via a Network Address Translation (NAT) instance in the public subnet. (Hourly charges for NAT instances apply)

Create an Amazon Virtual Private Cloud Cancel ☒

Select a VPC configuration below:

○ **VPC with a Single Public Subnet Only**

Your instances run in a private, isolated section of the AWS cloud with direct access to the Internet. Network access control lists and security groups can be used to provide strict control over inbound and outbound network traffic to your instances.

○ **VPC with Public and Private Subnets**

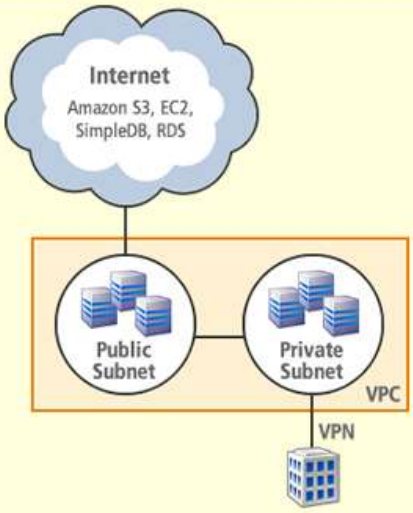
In addition to containing a public subnet, this configuration adds a private subnet whose instances are not addressable from the Internet. Instances in the private subnet can establish outbound connections to the Internet via the public subnet using Network Address Translation.

◉ **VPC with Public and Private Subnets and Hardware VPN Access**

This configuration adds an IPsec Virtual Private Network (VPN) connection between your Amazon VPC and your datacenter - effectively extending your datacenter to the cloud while also providing direct access to the Internet for public subnet instances in your Amazon VPC.

○ **VPC with a Private Subnet Only and Hardware VPN Access**

Your instances run in a private, isolated section of the AWS cloud with a private subnet whose instances are not addressable from the Internet. You can connect this private subnet to your corporate datacenter via an IPsec Virtual Private Network (VPN) tunnel.



Creates: a /16 network with two /24 subnets. One subnet is directly connected to the Internet while the other subnet is connected to your corporate network via IPsec VPN tunnel. (VPN charges apply)

Select a VPC configuration below:

○ **VPC with a Single Public Subnet Only**

Your instances run in a private, isolated section of the AWS cloud with direct access to the Internet. Network access control lists and security groups can be used to provide strict control over inbound and outbound network traffic to your instances.

○ **VPC with Public and Private Subnets**

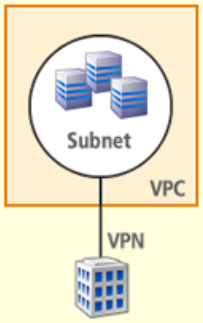
In addition to containing a public subnet, this configuration adds a private subnet whose instances are not addressable from the Internet. Instances in the private subnet can establish outbound connections to the Internet via the public subnet using Network Address Translation.

○ **VPC with Public and Private Subnets and Hardware VPN Access**

This configuration adds an IPsec Virtual Private Network (VPN) connection between your Amazon VPC and your datacenter - effectively extending your datacenter to the cloud while also providing direct access to the Internet for public subnet instances in your Amazon VPC.

◉ **VPC with a Private Subnet Only and Hardware VPN Access**

Your instances run in a private, isolated section of the AWS cloud with a private subnet whose instances are not addressable from the Internet. You can connect this private subnet to your corporate datacenter via an IPsec Virtual Private Network (VPN) tunnel.



Creates: a /16 network with a /24 subnet and provisions an IPsec VPN tunnel between your Amazon VPC and your corporate network. (VPN charges apply)