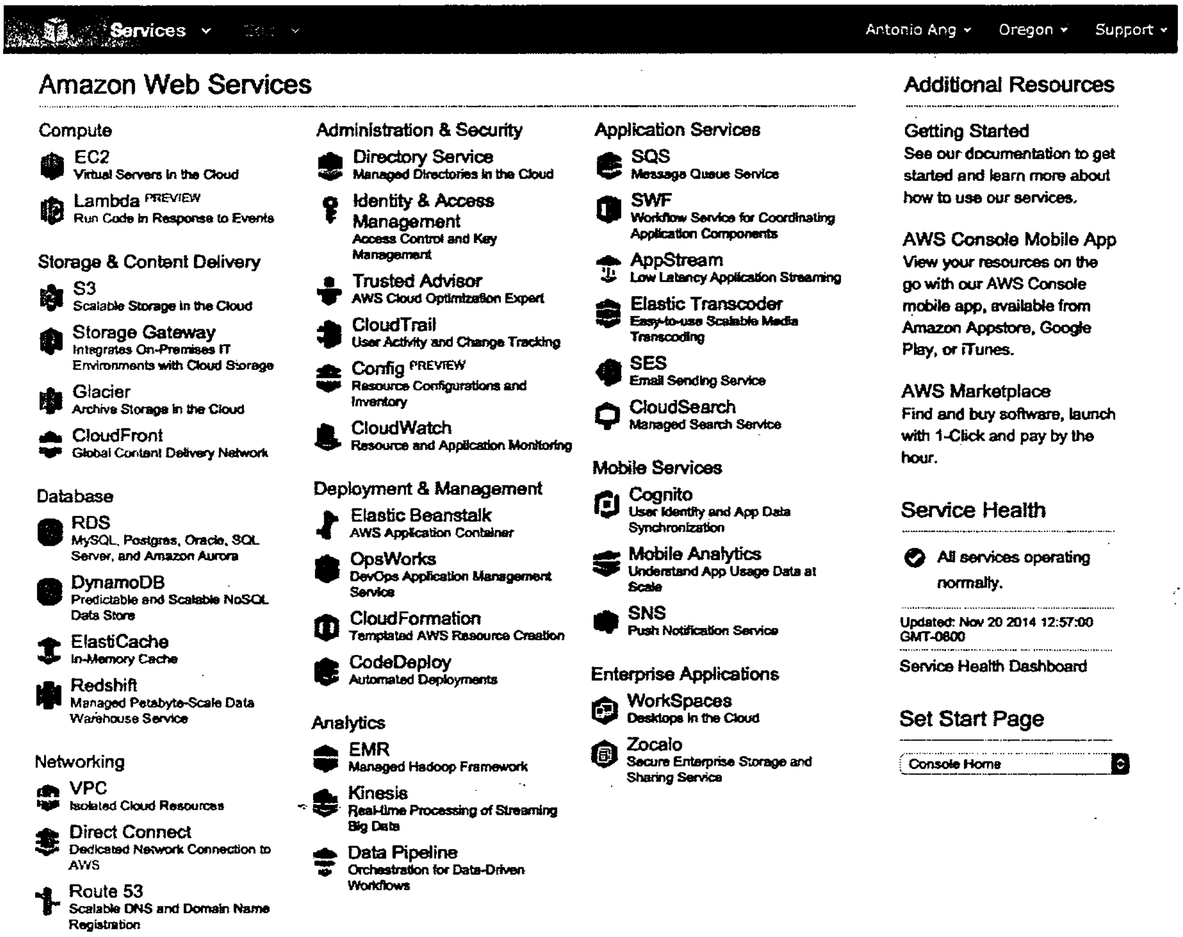
Lab 7

INTRODUCTION TO VIRTUAL PRIVATE CLOUD (VPC)

STEP 1: Log In to the Amazon Web Service Console

This laboratory experience is about Amazon Web Services and you will use the AWS Management Console in order to complete all the lab steps.



Amazon Web Services

Compute

- EC2 Virtual Servers in the Cloud
- Lambda PREVIEW Run Code in Response to Events

Storage & Content Delivery

- S3 Scalable Storage in the Cloud
- Storage Gateway Integrates On-Premises IT Environments with Cloud Storage
- Glacier Archive Storage in the Cloud
- CloudFront Global Content Delivery Network

Database

- RDS MySQL, Postgres, Oracle, SQL Server, and Amazon Aurora
- DynamoDB Predictable and Scalable NoSQL Data Store
- ElastiCache In-Memory Cache
- Redshift Managed Petabyte-Scale Data Warehouse Service

Networking

- VPC Isolated Cloud Resources
- Direct Connect Dedicated Network Connection to AWS
- Route 53 Scalable DNS and Domain Name Registration

Administration & Security

- Directory Service Managed Directories in the Cloud
- Identity & Access Management Access Control and Key Management
- Trusted Advisor AWS Cloud Optimization Expert
- CloudTrail User Activity and Change Tracking
- Config PREVIEW Resource Configurations and Inventory
- CloudWatch Resource and Application Monitoring

Deployment & Management

- Elastic Beanstalk AWS Application Container
- OpsWorks DevOps Application Management Service
- CloudFormation Templated AWS Resource Creation
- CodeDeploy Automated Deployments

Analytics

- EMR Managed Hadoop Framework
- Kinesis Real-time Processing of Streaming Big Data
- Data Pipeline Orchestration for Data-Driven Workflows

Application Services

- SQS Message Queue Service
- SWF Workflow Service for Coordinating Application Components
- AppStream Low Latency Application Streaming
- Elastic Transcoder Easy-to-use Scalable Media Transcoding
- SES Email Sending Service
- CloudSearch Managed Search Service

Mobile Services

- Cognito User Identity and App Data Synchronization
- Mobile Analytics Understand App Usage Data at Scale
- SNS Push Notification Service

Enterprise Applications

- WorkSpaces Desktops in the Cloud
- Zocalo Secure Enterprise Storage and Sharing Service

Additional Resources

Getting Started
See our documentation to get started and learn more about how to use our services.

AWS Console Mobile App
View your resources on the go with our AWS Console mobile app, available from Amazon Appstore, Google Play, or iTunes.

AWS Marketplace
Find and buy software, launch with 1-Click and pay by the hour.

Service Health

✓ All services operating normally.

Updated: Nov 20 2014 12:57:00 GMT-0800

Service Health Dashboard

Set Start Page

Console Home

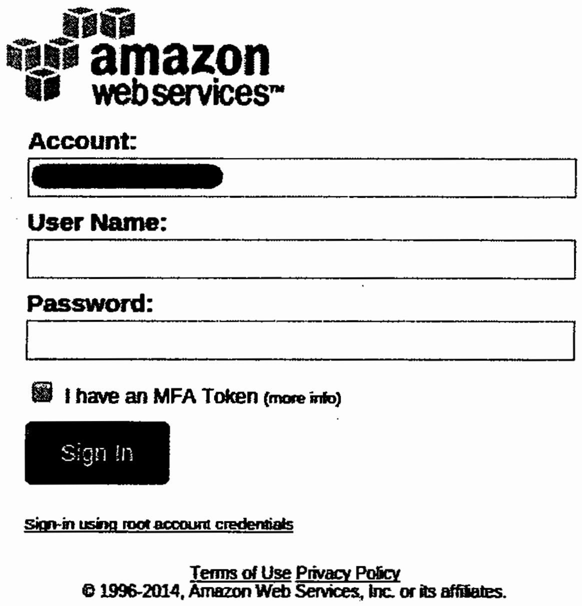
The AWS Management Console is a web control panel for managing all your AWS resources, from EC2 instances to SNS topics. The console enables cloud management for all aspects of the AWS account, including managing security credentials, or even setting up new IAM Users.

Log in to the AWS Management Console

In order to start the laboratory experience, open the Amazon Console by clicking this button:

Open AWS Console

Log in with the username **xxxx** and the password **xxxx** .



Account:

[redacted]

User Name:

Password:

☒ I have an MFA Token (more info)

Sign In

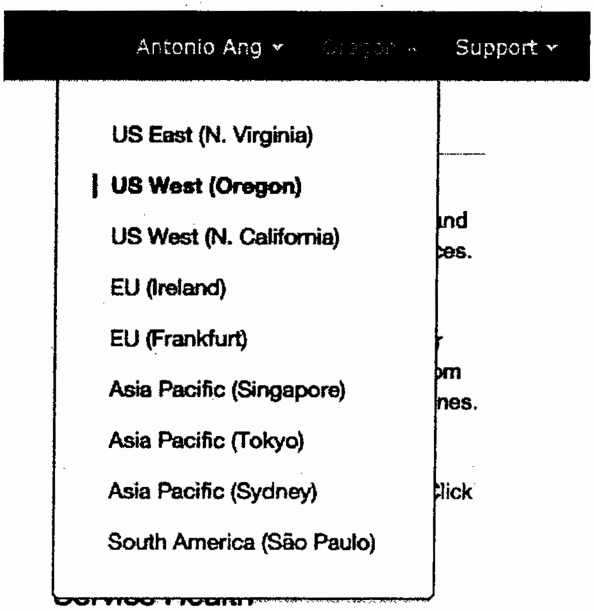
Sign-in using root account credentials

Terms of Use Privacy Policy
© 1996-2014, Amazon Web Services, Inc. or its affiliates.

Select the right AWS Region

Amazon Web Services is available in different regions all over the world, and the console lets you provision resources across multiple regions. You usually choose a region that best suits your business needs to optimize your customer's experience, but you must use the region **US West (Oregon)** for this laboratory.

You can select the **US West (Oregon)** region using the upper right dropdown menu on the AWS Console page.



STEP 2: Create a VPC

Amazon OpsWorks lets you easily orchestrate the different parts of your application using**Chef** to perform the actual automation. It presents the different AWS resources that make up your app as multiple layers, each composed of resources. A typical app might have two layers, an app server layer (where your Ruby/NodeJS/Python/PHP app actually runs) and a database layer (backed by RDS). Typically, you'd manage each instance and RDS installation separately, but with OpsWorks you can manage all instances in the "app server" layer together.

The advantage of using Chef is that you can use AWS' published OpsWorks cookbooks, open source community cookbooks, build your own, or mix and match. AWS publishes cookbooks for typical Rails applications, Nginx proxies, memcached servers, monitoring, haproxy, and more.

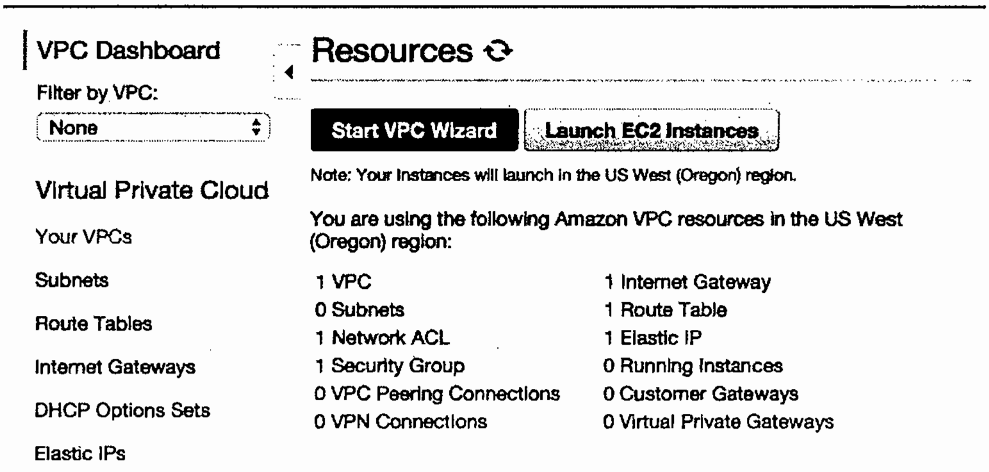
But before we get started building our first OpsWork stack, I'd like to remind you that it is just a collection of resources, and often doesn't create underlying resources like VPC networks automatically do. So we'll need to digress and make a VPC for all our instances to inhabit first.

Amazon Virtual Private Cloud (Amazon VPC) enables you to launch AWS resources into a virtual network that you've defined. This virtual network closely resembles a traditional network that you'd operate in your own data center -- with the benefits of using the scalable infrastructure of AWS. It is logically isolated from other virtual networks in the AWS cloud.

You can create a new VPC using the AWS Management Console.

Select the VPC service from the Management Console dashboard:

From the VPC dashboard, click on **Your VPCs** link in the sidebar menu.

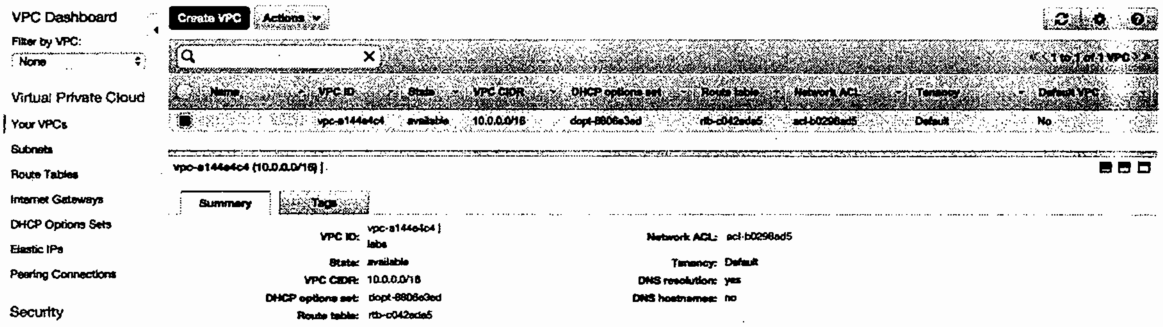


Note: Your instances will launch in the US West (Oregon) region.

You are using the following Amazon VPC resources in the US West (Oregon) region:

1 VPC	1 Internet Gateway
0 Subnets	1 Route Table
1 Network ACL	1 Elastic IP
1 Security Group	0 Running Instances
0 VPC Peering Connections	0 Customer Gateways
0 VPN Connections	0 Virtual Private Gateways

Your VPCs page lists all previously created VPCs (any new AWS account comes with a default fully-working VPC); click on the **Create VPC** blue button to begin creating a new VPC.



Name	VPC ID	State	VPC CIDR	DHCP options set	Route table	Network ACL	Tenancy	Default VPC
	vpc-a144e4c4	available	10.0.0.0/16	dopt-8806e3ed	rtb-c042ede5	acl-b0298ad5	Default	No

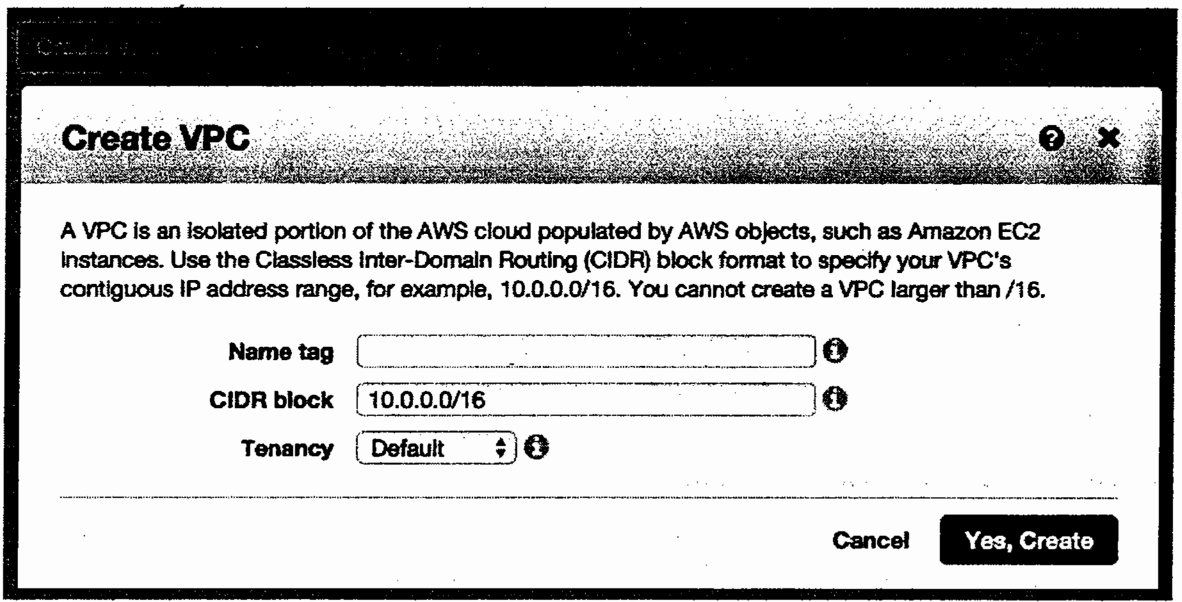
vpc-a144e4c4 (10.0.0.0/16) |

VPC ID: vpc-a144e4c4 | labs
State: available
VPC CIDR: 10.0.0.0/16
DHCP options set: dopt-8806e3ed
Route table: rtb-c042ede5

Network ACL: acl-b0298ad5
Tenancy: Default
DNS resolution: yes
DNS hostnames: no

In the Create VPC dialog box, specify the following VPC details as necessary, then click **Yes, Create**.

- ✓ **Name tag**: **vepsun-labs**. This is the name for your VPC; doing so creates a tag with a key of Name and the value that you specify.
- ✓ **CIDR block**: **10.0.0.0/16**. You should specify a CIDR block from the private (non-publicly routable) IP address ranges as specified in RFC 1918.
- ✓ **Tenancy**: **default**. Dedicated tenancy ensures your instances run on single-tenant hardware.



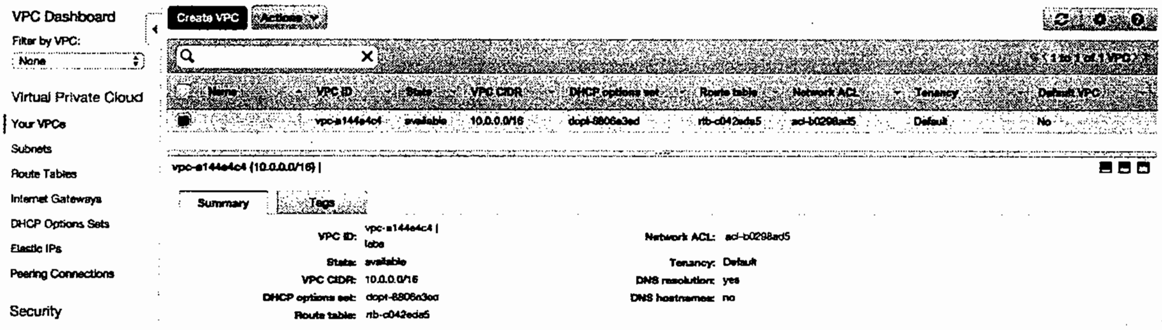
A VPC is an isolated portion of the AWS cloud populated by AWS objects, such as Amazon EC2 instances. Use the Classless Inter-Domain Routing (CIDR) block format to specify your VPC's contiguous IP address range, for example, 10.0.0.0/16. You cannot create a VPC larger than /16.

Name tag:
CIDR block: 10.0.0.0/16
Tenancy: Default

Amazon creates the requested VPC and the following linked services:

- ✓ a **DHCP options set** (this set enables DNS for instances that need to communicate over the VPC's Internet gateway)
- ✓ a **Route Table** (it contains a set of rules, called *routes*, that are used to determine where network traffic is directed)
- ✓ a **Network ACL** (it is a list of rules to determine whether traffic is allowed in or out of any subnet associated with the network ACL)

Note that no Subnets or Internet Gateways are automatically created -- you need to add them autonomously.



Name	VPC ID	State	VPC CIDR	DHCP options set	Route table	Network ACL	Tenancy	Default VPC
	vpc-a144e4c4	available	10.0.0.0/16	dopt-8806a3ed	rtb-c042ada5	acl-b0298ad5	Default	No

vpc-a144e4c4 (10.0.0.0/16) |

VPC ID: vpc-a144e4c4 | labs
State: available
VPC CIDR: 10.0.0.0/16
DHCP options set: dopt-8806a3ed
Route table: rtb-c042eda5

Network ACL: acl-b0298ad5
Tenancy: Default
DNS resolution: yes
DNS hostnames: no

Now you are ready to create your VPC subnets and customize the routing table.

STEP 3: Create a VPC subnet

A **VPC subnet** is a range of IP addresses in your VPC. You can add one or more subnets in each Availability Zone, but each subnet must reside entirely within one Availability Zone and cannot span zones. **Availability Zones** are distinct locations that are engineered to be isolated from failures in other Availability Zones. By launching instances in separate Availability Zones, you can protect your applications from the failure of a single location.

You can create a new subnet for your previously created VPC using the AWS Management Console.

Select the VPC service from the Management Console dashboard:

Networking

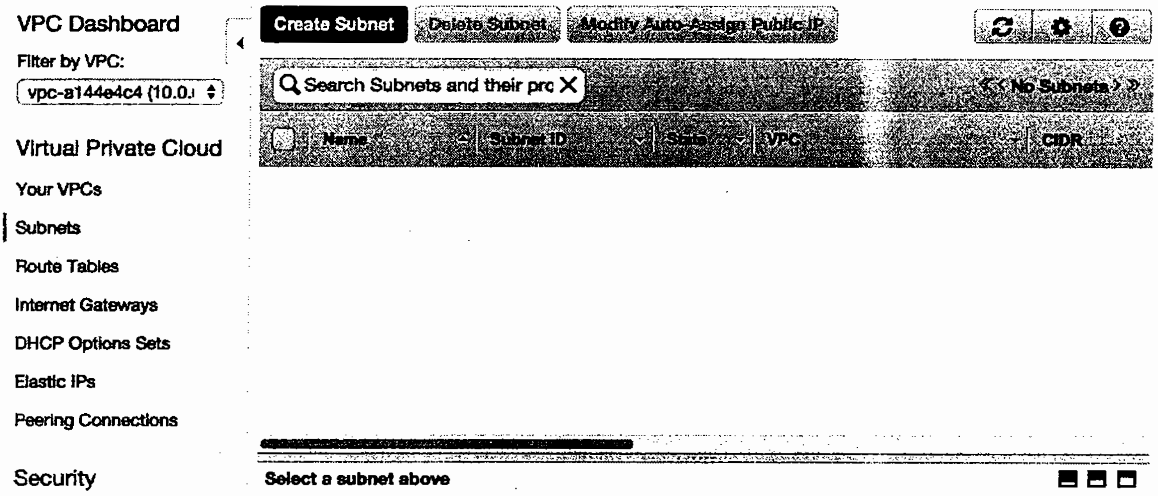
VPC

Isolated Cloud Resources

From the VPC dashboard, click the **Subnets** link in the sidebar menu.

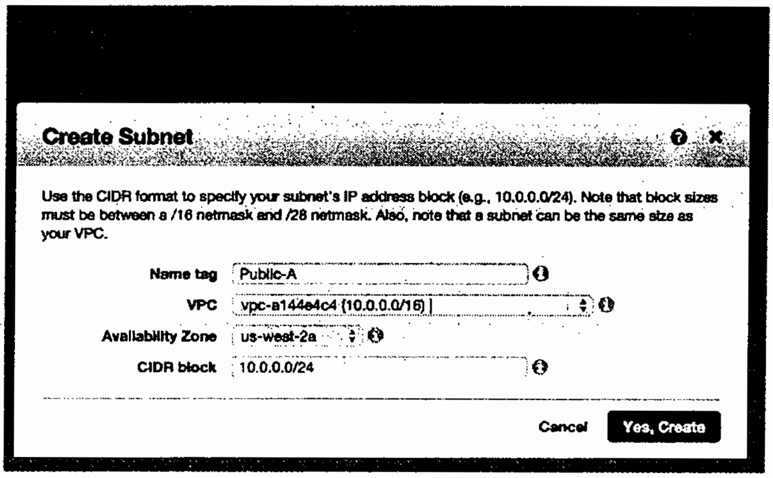
Your Subnets page lists all previously created subnets, you can use the **Filter by VPC** feature for listing only the services linked to a specific VPC.

Click on the **Create Subnet** blue button to begin creating a new subnet.



In the Create Subnet dialog box, specify the following Subnet details then click **Yes, Create**.

- ✓ **Name tag**: **Public-A**. This is the name for your subnet; doing so creates a tag with a key of Name and the value that you specify.
- ✓ **VPC**: **vepsun-labs**.
- ✓ **Availability Zone**: **us-west2a**.
- ✓ **CIDR block**: **10.0.0.0/24**. You should specify a CIDR block in the selected VPC.



Create Subnet

Use the CIDR format to specify your subnet's IP address block (e.g., 10.0.0.0/24). Note that block sizes must be between a /16 netmask and /28 netmask. Also, note that a subnet can be the same size as your VPC.

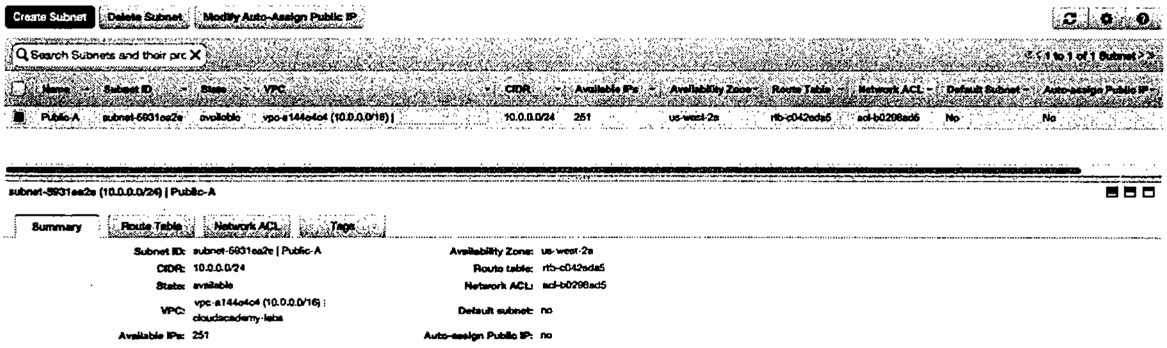
Name tag: Public-A

VPC: vpc-a144e4c4 (10.0.0.0/16) |

Availability Zone: us-west-2a

CIDR block: 10.0.0.0/24

As you can see, the created subnet is automatically attached to the default VPC Route table and the default Network ACL.



Name	Subnet ID	State	VPC	CIDR	Available IPs	Availability Zone	Route Table	Network ACL	Default Subnet	Auto-assign Public IP
Public-A	subnet-5931ea2e	available	vpc-a144e4o4 (10.0.0.0/16) |	10.0.0.0/24	251	us-west-2a	rtb-c042eda5	acl-b0298ad5	No	No

subnet-5931ea2e (10.0.0.0/24) | Public-A

Subnet ID:	subnet-5931ea2e | Public-A	Availability Zone:	us-west-2a
CIDR:	10.0.0.0/24	Route table:	rtb-c042eda5
State:	available	Network ACL:	acl-b0298ad5
VPC:	vpc-a144e4o4 (10.0.0.0/16) | cloudacademy-labs	Default subnet:	no
Available IPs:	251	Auto-assign Public IP:	no

STEP 4: Create a VPC Internet Gateway

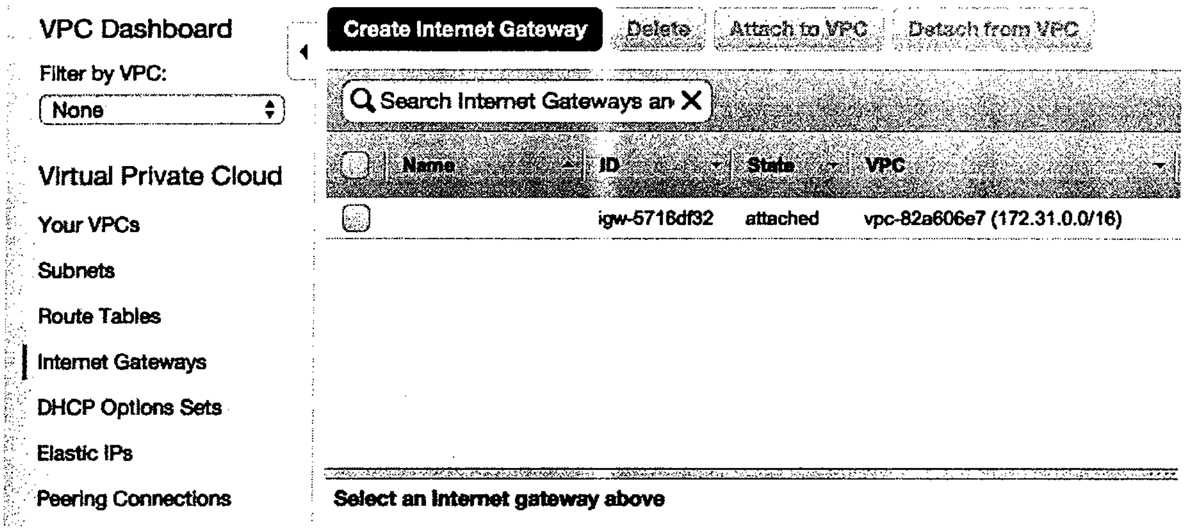
An **Internet Gateway** is a horizontally scaled, redundant, and highly available VPC component that allows communication between instances in your VPC and the Internet. It imposes no availability risks or bandwidth constraints on your network traffic. An Internet gateway serves two purposes: to provide a target in your VPC route tables for Internet-routable traffic, and to perform network address translation (NAT) for instances that have been assigned public IP addresses.

You can create a new **Internet Gateway** for your previously created VPC using the AWS Management Console.

Select the VPC service from the AWS Management Console dashboard:

From the VPC dashboard, click the **Internet Gateways** link in the sidebar menu.

The **Internet Gateways** page lists all previously created gateways. Click on the **Create Internet Gateway** blue button to begin creating a new gateway.

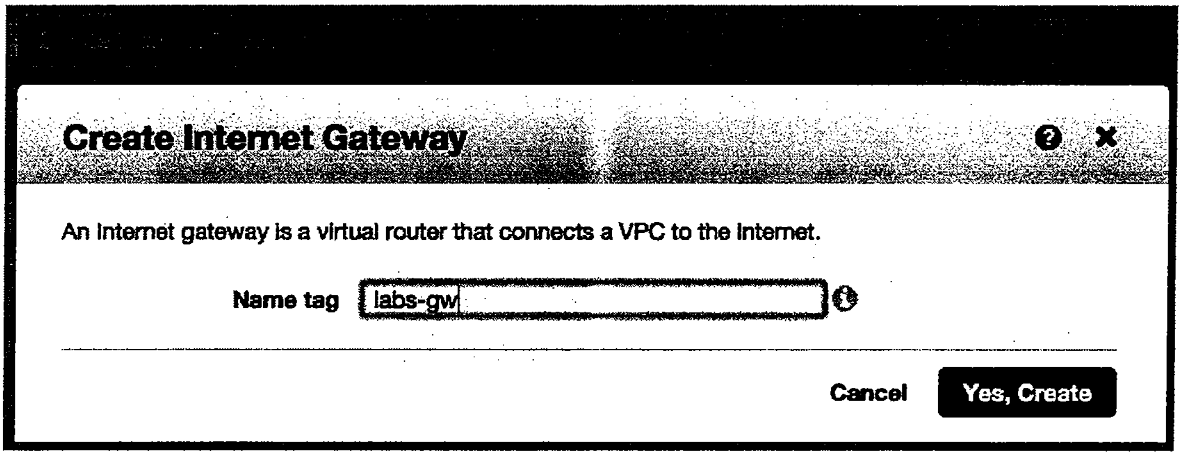


□	Name	ID	State	VPC
□		igw-5716df32	attached	vpc-82a606e7 (172.31.0.0/16)

Select an Internet gateway above

Creating a gateway is a onestep operation, you only need to choose a meaningful name.

Use labs-gw as **Name tag** and then click **Yes, Create**.

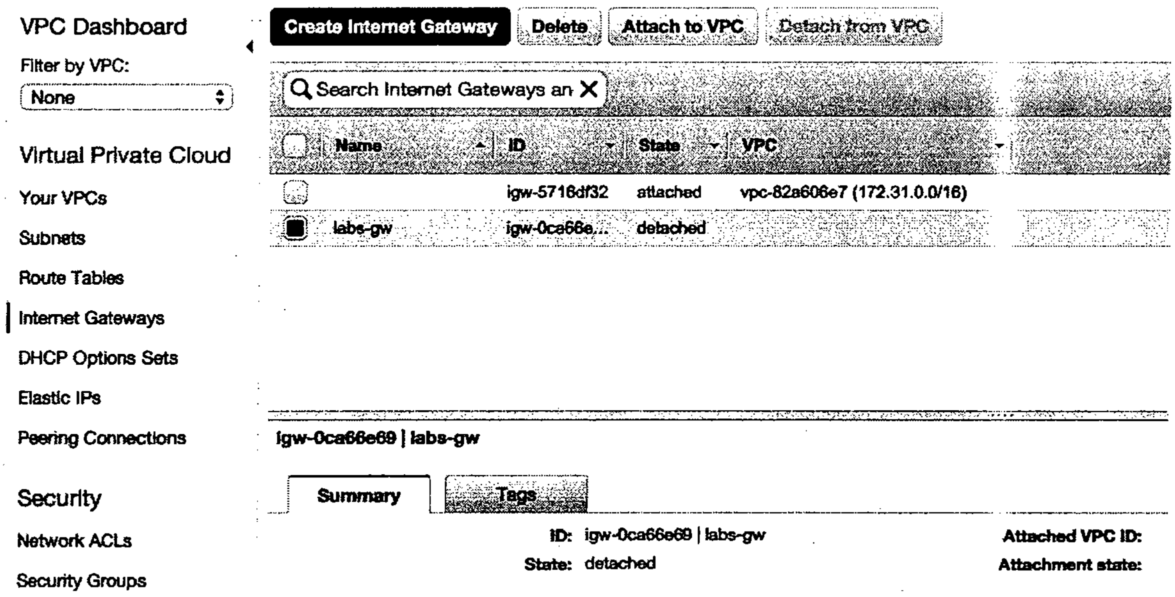


An Internet gateway is a virtual router that connects a VPC to the Internet.

Name tag: labs-gw

How to attach the Internet Gateway to a VPC

Select the Internet gateway that you just created, and then click **Attach to VPC**.



	Name	ID	State	VPC
□		igw-5716df32	attached	vpc-82a606e7 (172.31.0.0/16)
■	labs-gw	igw-0ca66e...	detached	

igw-0ca66e69 | labs-gw

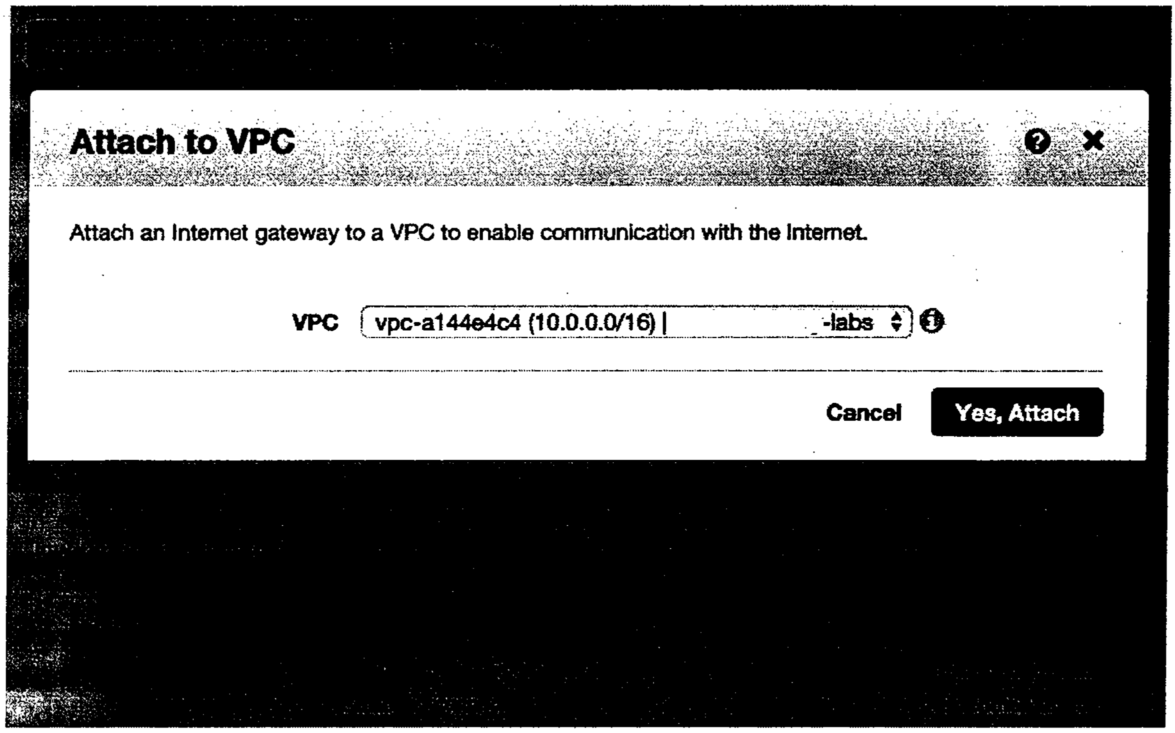
ID: igw-0ca66e69 | labs-gw

State: detached

Attached VPC ID:

Attachment state:

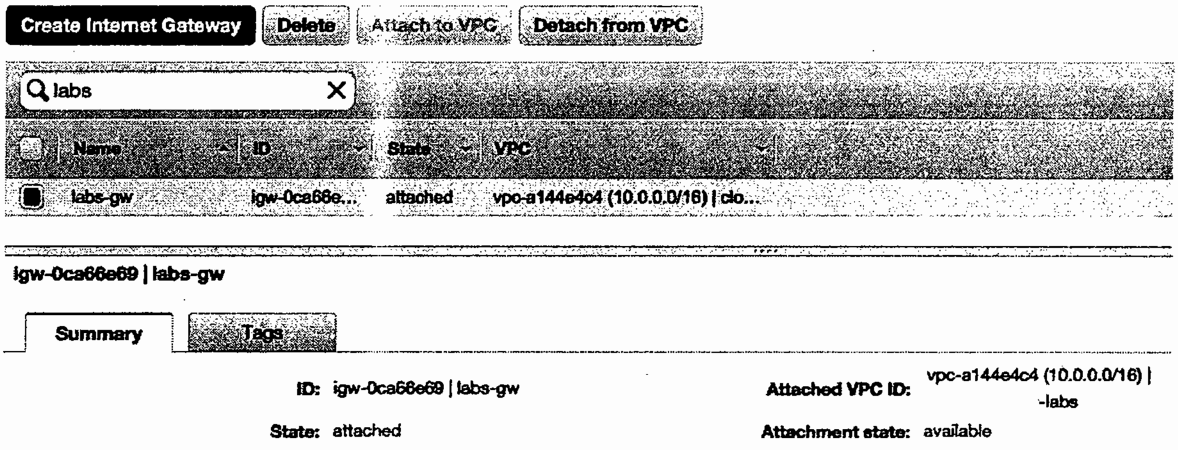
In the Attach to VPC dialog box, select the VPC **vepsun-labs** from the list, and then click **Yes, Attach**.



Attach an Internet gateway to a VPC to enable communication with the Internet.

VPC vpc-a144e4c4 (10.0.0.0/16) | -labs

Your new Internet Gateway is ready to be used by the EC2 instances of the selected VPC.



Name	ID	State	VPC
labs-gw	igw-0ca66e...	attached	vpc-a144e4c4 (10.0.0.0/16) | clo...

igw-0ca66e69 | labs-gw

ID: igw-0ca66e69 | labs-gw

State: attached

Attached VPC ID: vpc-a144e4c4 (10.0.0.0/16) | -labs

Attachment state: available

STEP 5: Connect the Internet Gateway to the VPC Route Table

To use an **Internet gateway** your subnet's **route table** must contain a route that directs Internet-bound traffic to the Internet gateway. You can scope the route to all destinations not explicitly known to the route table (0.0.0.0/0), or you can scope the route to a narrower range of IP addresses; for example, the public IP addresses of your company's public endpoints outside of AWS, or the Elastic IP addresses of other Amazon EC2 instances outside your VPC. If your subnet is associated with a route table that has a route to an Internet gateway, it's known as a**public subnet**.

You can add routes to your previously created VPC **Route Table** using the AWS Management Console.

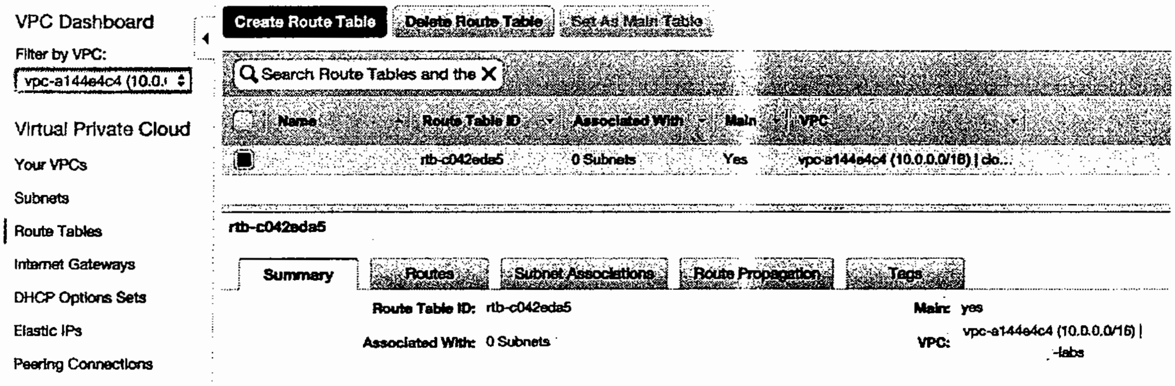
Select the VPC service from the AWS Management Console dashboard:

Networking

VPC
Isolated Cloud Resources

From the VPC dashboard, click the **Route tables** link in the sidebar menu.

The **Route tables** page lists all previously created route tables. In order to select the Route Table of your **vepsun-labs** VPC, you can check the VPC column or use the **Filter by VPC**feature in the left sidebar for listing the Route Tables attached to **vepsun-labs**.



Name	Route Table ID	Associated With	Main	VPC
	rtb-c042eda5	0 Subnets	Yes	vpc-a144e4c4 (10.0.0.0/16) | clo...

rtb-c042eda5

Route Table ID: rtb-c042eda5

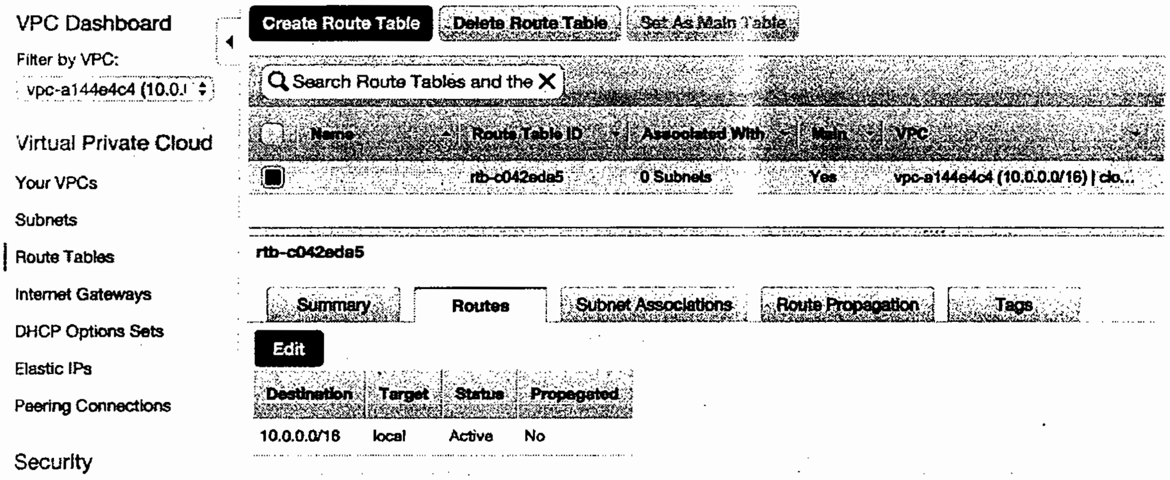
Associated With: 0 Subnets

Main: yes

VPC: vpc-a144e4c4 (10.0.0.0/16) | -labs

Select the **Main** route table to show its detailed information and then select the **Routes** tab pane.

Routes is set of rules which are used to determine where network traffic is directed. For adding a new route, click the blue **Edit** button.

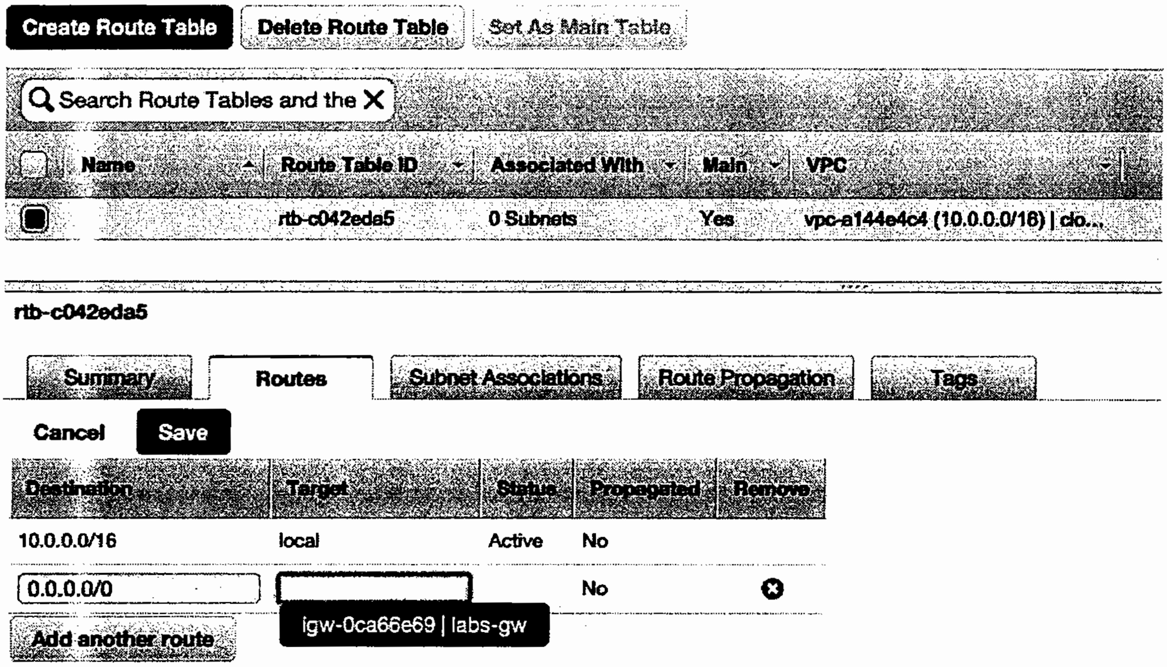


Name	Route Table ID	Associated With	Main	VPC
	rtb-c042eda5	0 Subnets	Yes	vpc-a144e4c4 (10.0.0.0/16) | clo...

rtb-c042eda5

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

Enter **0.0.0.0/0** a destination CIDR block and then select the previously created Internet Gateway from the Target list. Click **Save** when you're done.

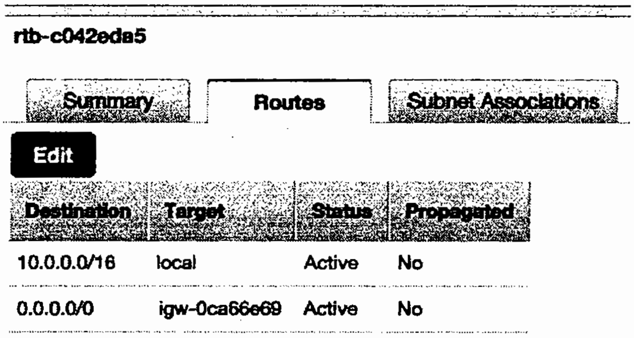


☐	Name	Route Table ID	Associated With	Main	VPC
■		rtb-c042eda5	0 Subnets	Yes	vpc-a144e4c4 (10.0.0.0/16) | clo...

rtb-c042eda5

Destination	Target	Status	Propagated	Remove
10.0.0.0/16	local	Active	No	
0.0.0.0/0			No	⊗

Thanks to the new route rule, all VPC external traffic will be routed to the Internet Gateway and then to the Internet.


rtb-c042eda5

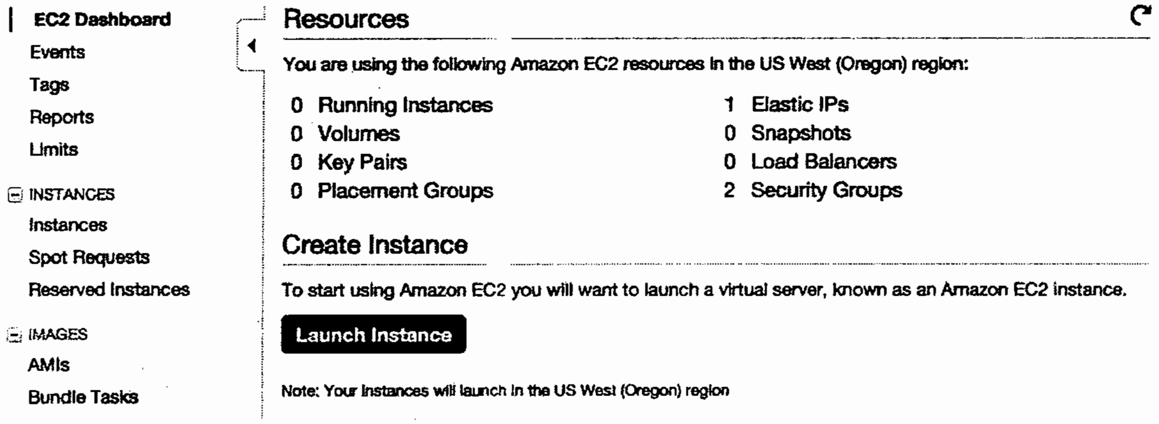
Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	igw-0ca66e69	Active	No

STEP 6: Create an EC2 instance

You can launch an EC2 instance using the EC2 launch wizard.

Select the EC2 service from the Management Console dashboard:

From the EC2 dashboard, click **Launch Instance**.



Resources

You are using the following Amazon EC2 resources in the US West (Oregon) region:

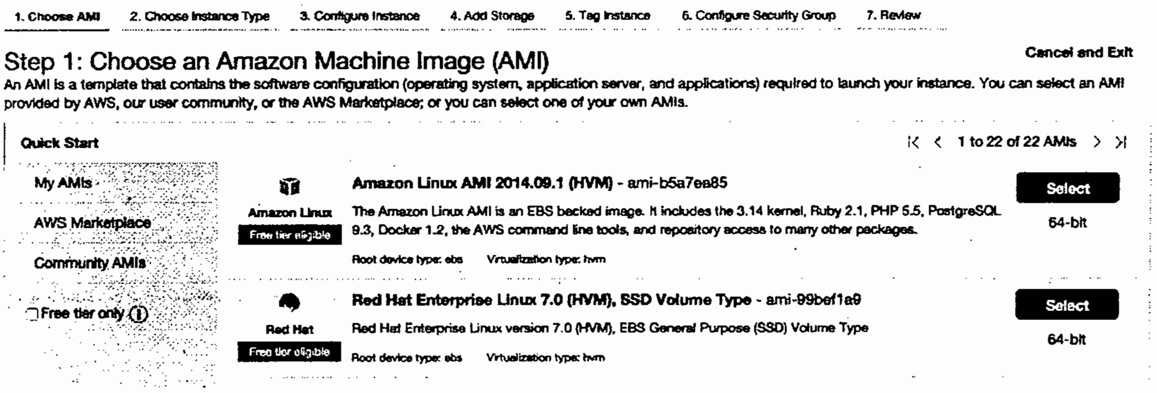
0 Running Instances	1 Elastic IPs
0 Volumes	0 Snapshots
0 Key Pairs	0 Load Balancers
0 Placement Groups	2 Security Groups

Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

Note: Your instances will launch in the US West (Oregon) region

The **Choose an Amazon Machine Image (AMI)** page displays a list of basic configurations called **Amazon Machine Images (AMIs)** that serve as templates for your instance. Select the first listed 64-bit **Amazon Linux AMI**.



Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Amazon Linux AMI 2014.09.1 (HVM) - ami-b5a7ea85

The Amazon Linux AMI is an EBS backed image. It includes the 3.14 kernel, Ruby 2.1, PHP 5.5, PostgreSQL 9.3, Docker 1.2, the AWS command line tools, and repository access to many other packages.

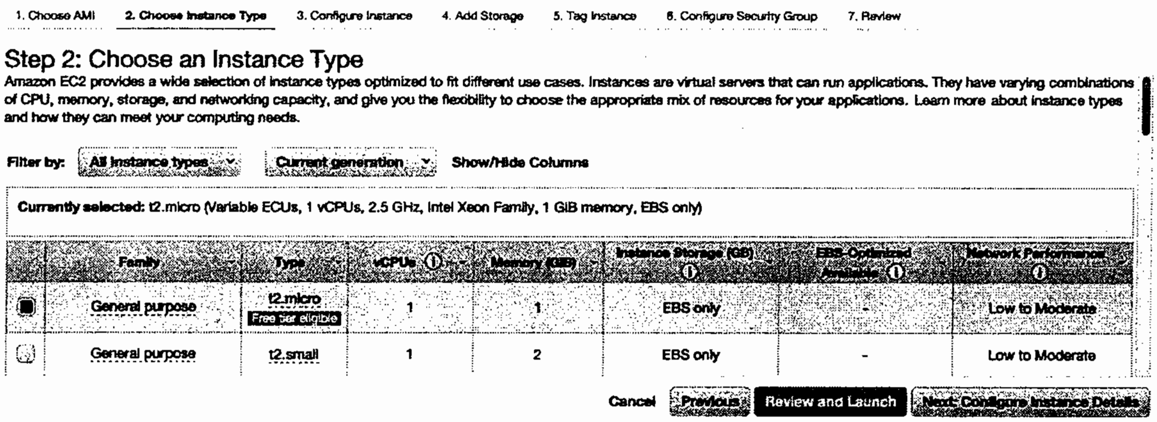
Root device type: ebs Virtualization type: hvm

Red Hat Enterprise Linux 7.0 (HVM), SSD Volume Type - ami-99bef1a9

Red Hat Enterprise Linux version 7.0 (HVM), EBS General Purpose (SSD) Volume Type

Root device type: ebs Virtualization type: hvm

On the **Choose an Instance Type** page, do **not** change any options and click **Next: Configure Instance Details.**



1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Tag Instance 6. Configure Security Group 7. Review

Step 2: Choose an Instance Type

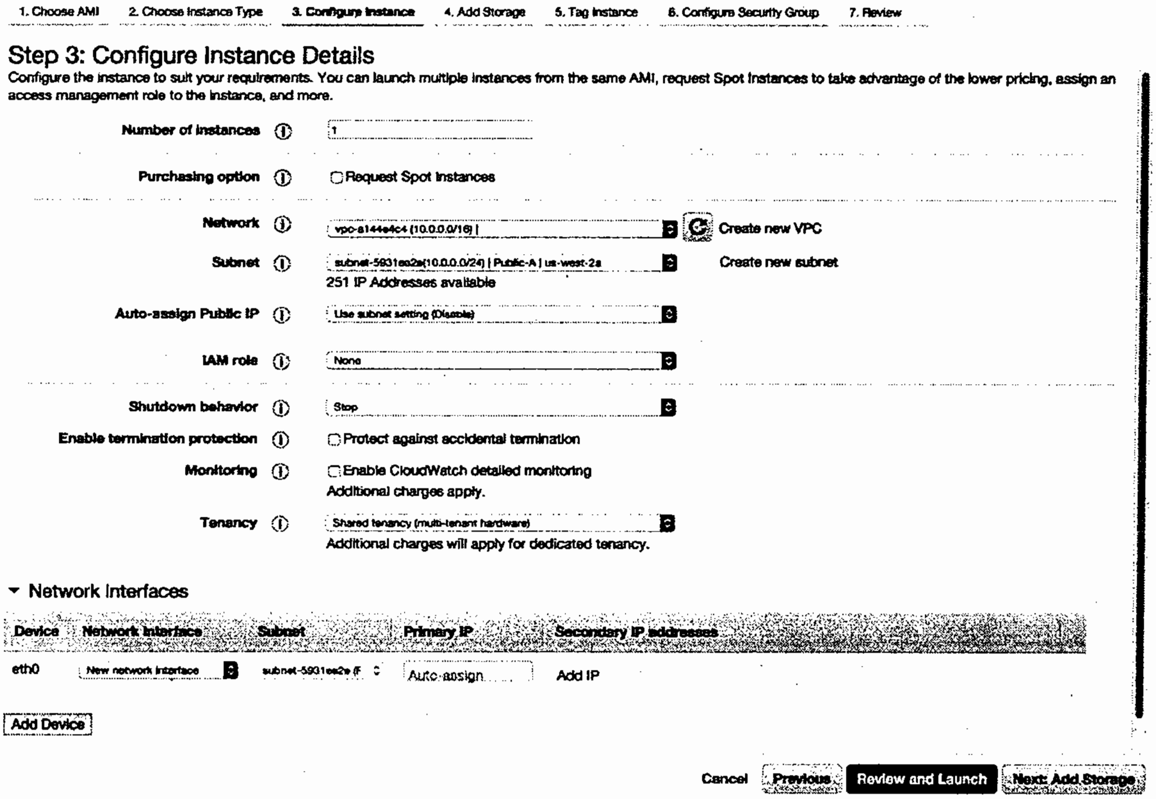
Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

Filter by: All instance types ˅ Current generation ˅ Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
●	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate
○	General purpose	t2.small	1	2	EBS only	-	Low to Moderate

On the **Configure Instance Details** tab, check the selected **Network (VPC)** and **Subnet**. Change them, if needed, and then click **Next: Add Storage**.



1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Tag Instance 6. Configure Security Group 7. Review

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot Instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances: 1

Purchasing option: □ Request Spot Instances

Network: vpc-a144e4c4 (10.0.0.0/16) | — Create new VPC

Subnet: subnet-5931ea2a(10.0.0.0/24) | Public-A | us-west-2a — Create new subnet
251 IP Addresses available

Auto-assign Public IP: Use subnet setting (Disable)

IAM role: None

Shutdown behavior: Stop

Enable termination protection: □ Protect against accidental termination

Monitoring: □ Enable CloudWatch detailed monitoring
Additional charges apply.

Tenancy: Shared tenancy (multi-tenant hardware)
Additional charges will apply for dedicated tenancy.

▾ Network Interfaces

Device	Network Interface	Subnet	Primary IP	Secondary IP addresses
eth0	New network interface	subnet-5931ea2a (F	Auto-assign	Add IP

On the **Add Storage** tab, do **not** change any options, and click the **Review and Launch** button.

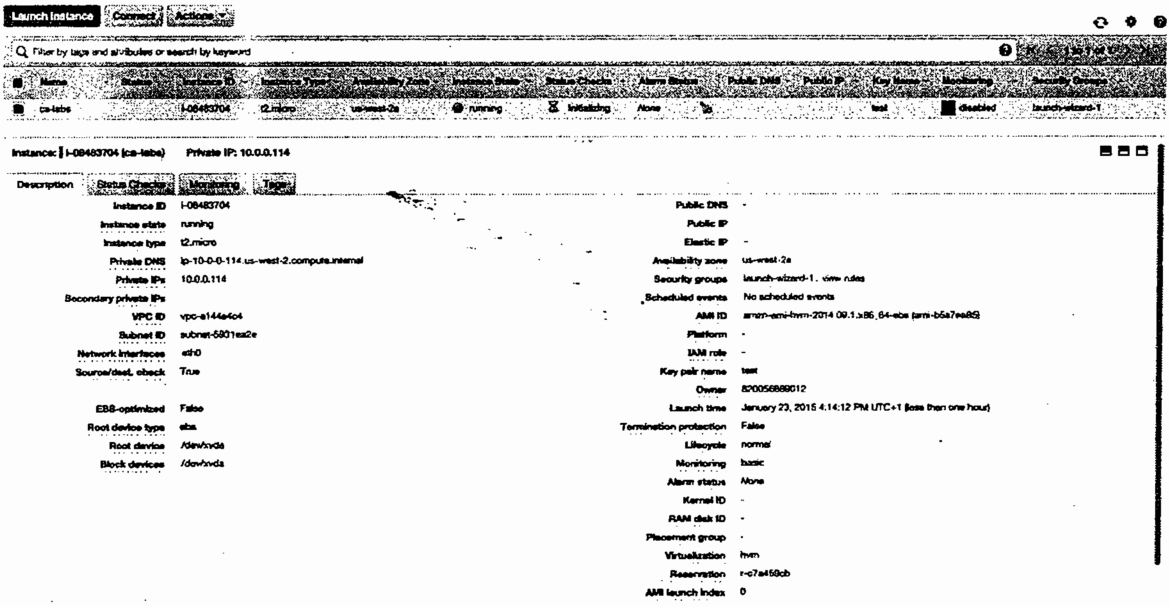
On the Review Instance Launch page, click **Launch**.

In the **Select an existing key pair or create a new key pair** dialog box, select **Create a new key pair**, then type a KeyPair name (e.g., "TestKeys") and download it.

Select the acknowledgment checkbox, and then click **Launch Instances**.

A confirmation page will let you know that your instance is launching. Click **View Instances** to close the confirmation page and return to the console.

On the Instances Screen, you can view the status of your instance. It will take a short time for your instance to be launched. When you launch an instance, its initial state defaults to *pending*. After the instance starts, its Instance State changes to *running*, and it receives a public DNS name.



Name	Status	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS	Public IP	Key Name	Monitoring	Security Groups
ca-labs		i-08483704	t2.micro	us-west-2a	● running	Initializing	None			test	disabled	launch-wizard-1

Instance: i-08483704 (ca-labs) Private IP: 10.0.0.114

Instance ID	i-08483704
Instance state	running
Instance type	t2.micro
Private DNS	ip-10-0-0-114.us-west-2.compute.internal
Private IPs	10.0.0.114
Secondary private IPs	
VPC ID	vpc-a144e4c4
Subnet ID	subnet-5801ea2e
Network interfaces	eth0
Source/dest. check	True
EBS-optimized	False
Root device type	ebs
Root device	/dev/xvda
Block devices	/dev/xvda

Public DNS	-
Public IP	
Elastic IP	-
Availability zone	us-west-2a
Security groups	launch-wizard-1. view rules
Scheduled events	No scheduled events
AMI ID	amzn-ami-hvm-2014.09.1.x86_64-ebs (ami-b5a7ea85)
Platform	-
IAM role	-
Key pair name	test
Owner	820056888012
Launch time	January 23, 2015 4:14:12 PM UTC+1 (less than one hour)
Termination protection	False
Lifecycle	normal
Monitoring	basic
Alarm status	None
Kernel ID	-
RAM disk ID	-
Placement group	-
Virtualization	hvm
Reservation	r-c7a459cb
AMI launch index	0

STEP 7: Allocate and Associate an Elastic IP

An **Elastic IP address (EIP)** is a static and public IP address that you can associate with an EC2 instance. You can allocate an Elastic IP address using the AWS Management Console.

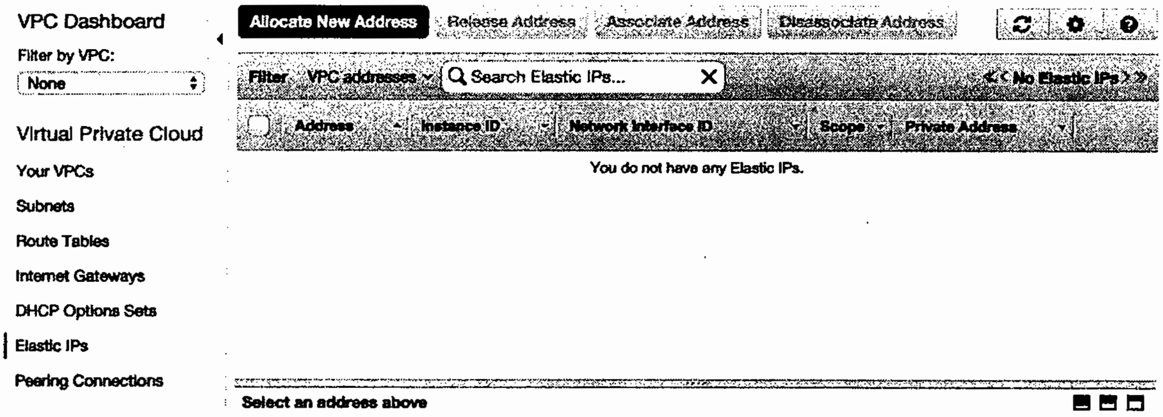
Select the VPC service from the Management Console dashboard:

Networking

VPC

Isolated Cloud Resources

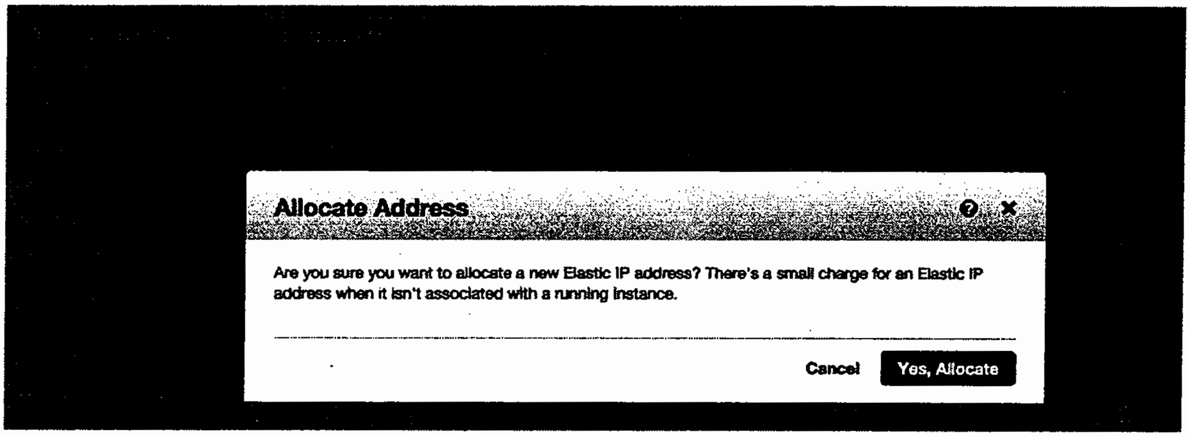
From the VPC dashboard, click on **Elastic IPs** link in the sidebar menu.



You do not have any Elastic IPs.

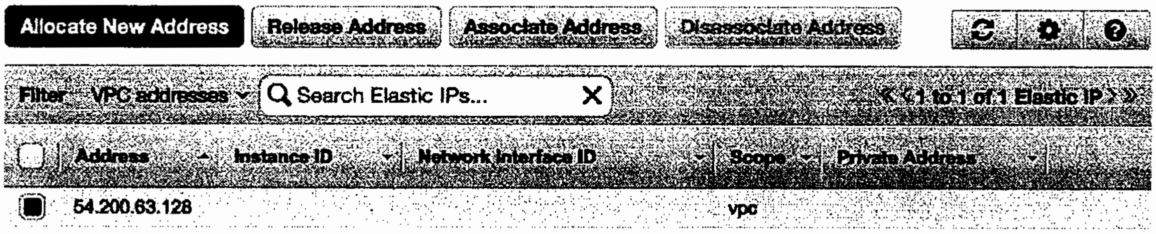
Select an address above

Click **Allocate New Address** and then click **Yes, Allocate**.



Are you sure you want to allocate a new Elastic IP address? There's a small charge for an Elastic IP address when it isn't associated with a running instance.

A new Elastic IPs is now ready to be associated with an EC2 instance.



	Address	Instance ID	Network Interface ID	Scope	Private Address
■	54.200.63.128			vpc	

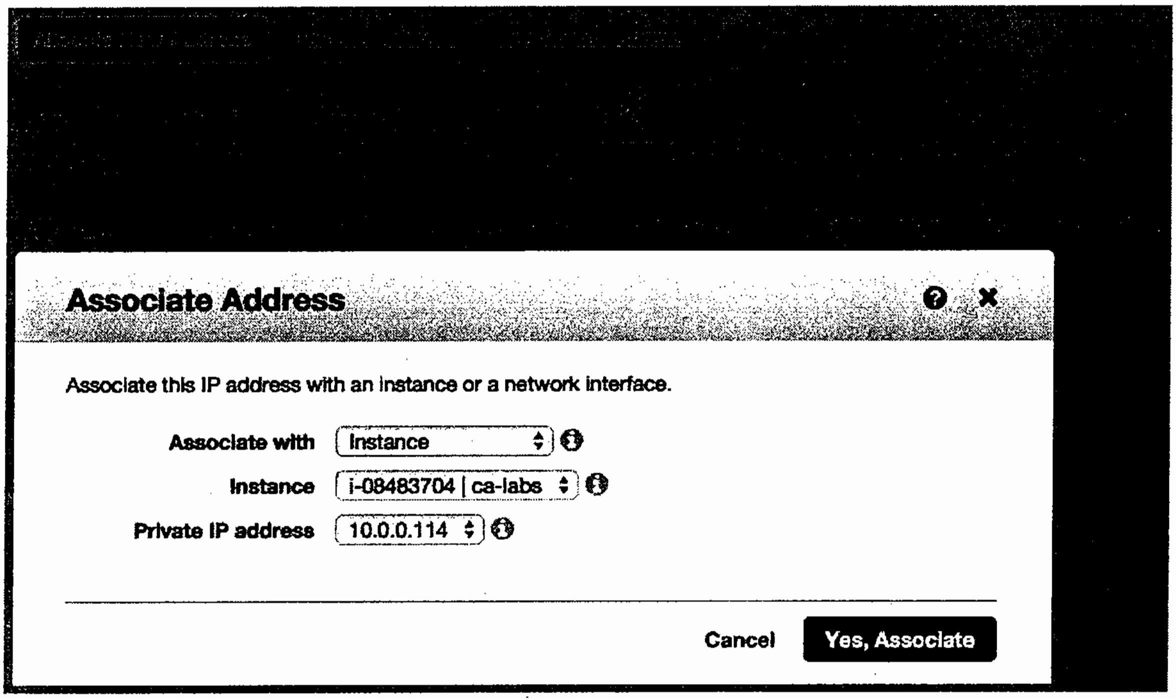
Associate the Elastic IP to an existing EC2 instance

In order to use the allocated EIP, you need to associate it with a previously launched instance or network interface.

Click **Associate Address** and then select the following values:

- ✓ Associate with: instance
- ✓ Instance: ca-labs

Click **Yes, Associate** to associate the EIP to the selected EC2 instance.



Associate Address

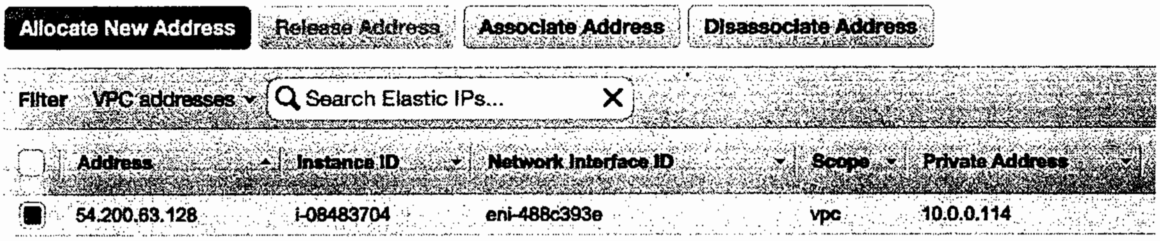
Associate this IP address with an instance or a network interface.

Associate with: Instance

Instance: i-08483704 | ca-labs

Private IP address: 10.0.0.114

The Elastic IPs page now lists the EIP associated to the ca-labs.

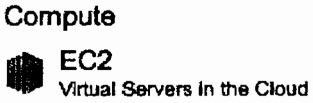


□	Address	Instance ID	Network Interface ID	Scope	Private Address
■	54.200.63.128	i-08483704	eni-488c393e	vpc	10.0.0.114

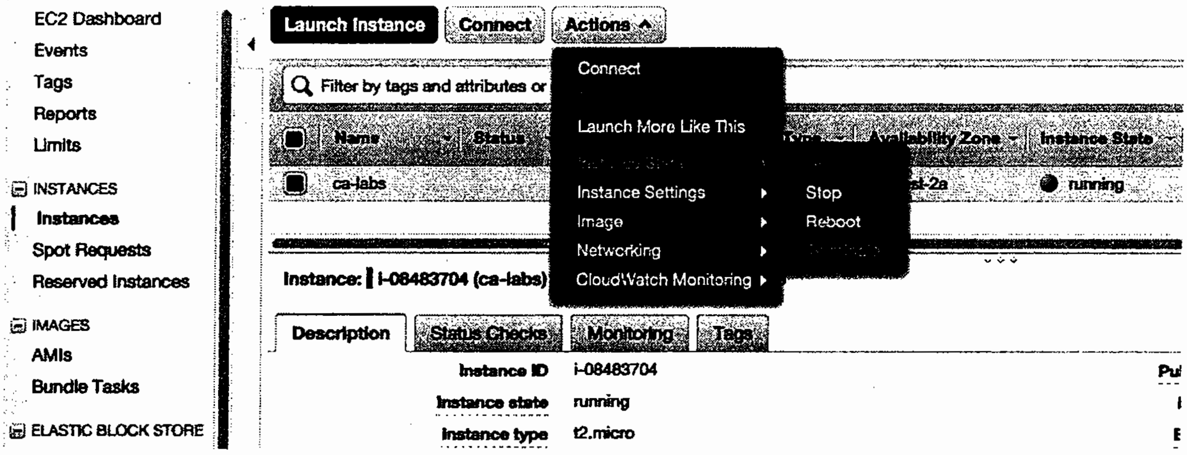
STEP 8: Terminate an EC2 instance

When you've decided that you no longer need an instance, you can terminate it.

Select the EC2 service from the Management Console dashboard:



In the navigation pane, click **Instances**.



ca-labs ... st-2a ● running

Instance: i-08483704 (ca-labs)

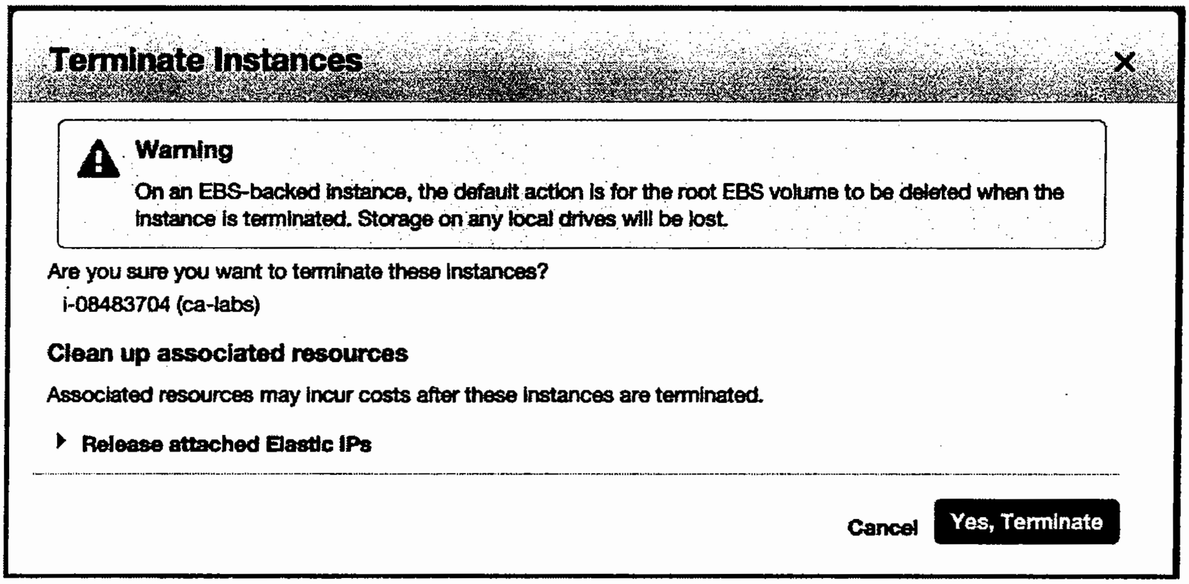
Instance ID i-08483704

Instance state running

Instance type t2.micro

Select the instance ca-labs, click **Actions**, select **Instance State**, and then click **Terminate**.

Click **Yes, Terminate** when prompted for confirmation.



Terminate Instances ✕

Warning

On an EBS-backed instance, the default action is for the root EBS volume to be deleted when the instance is terminated. Storage on any local drives will be lost.

Are you sure you want to terminate these instances?
i-08483704 (ca-labs)

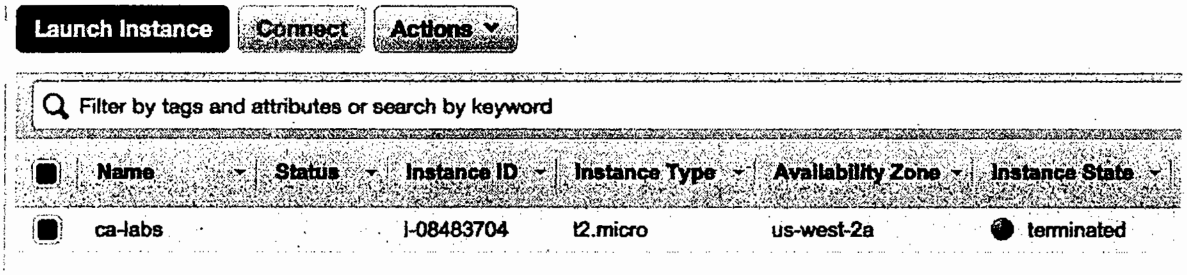
Clean up associated resources

Associated resources may incur costs after these instances are terminated.

▸ Release attached Elastic IPs

Cancel | Yes, Terminate

Now your instance is completely destroyed.



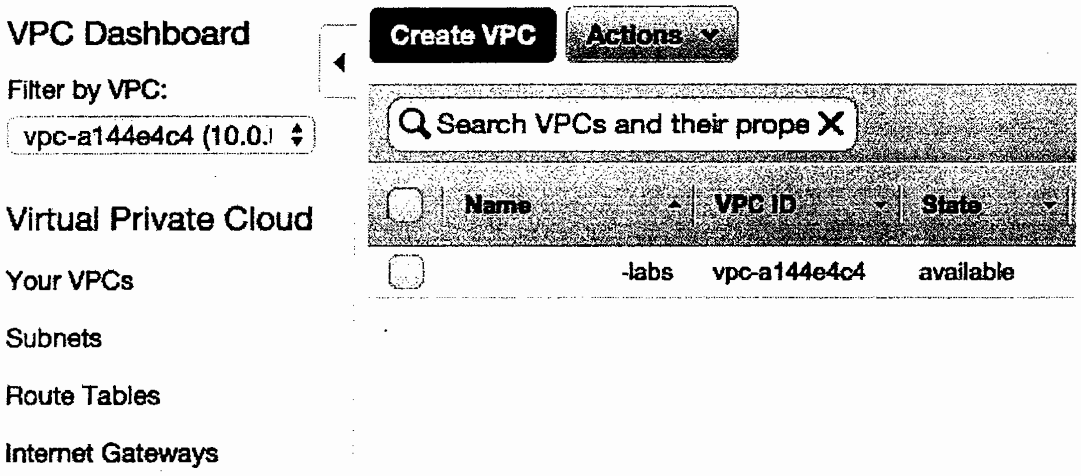
Name	Status	Instance ID	Instance Type	Availability Zone	Instance State
ca-labs		i-08483704	t2.micro	us-west-2a	● terminated

STEP 9: Destroy a VPC

You can delete your VPC at any time, but you must terminate all instances in the VPC first. You don't need to delete all VPC components (subnets, security groups, network ACLs, route tables, Internet gateways, VPC peering connections, and DHCP options) one-by-one if you use the AWS Management Console.

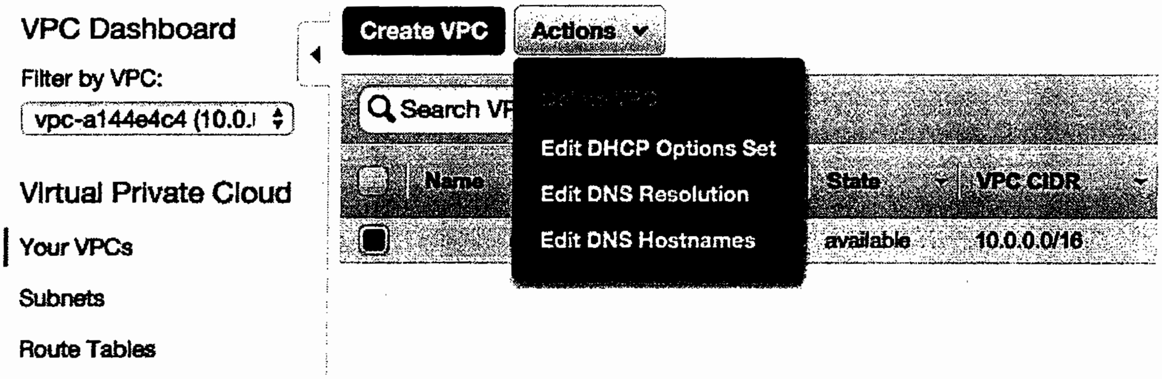
Select the VPC service from the Management Console dashboard: Select the VPC service from the Management Console dashboard:

From the VPC dashboard, click on **Your VPCs** link in the sidebar menu.

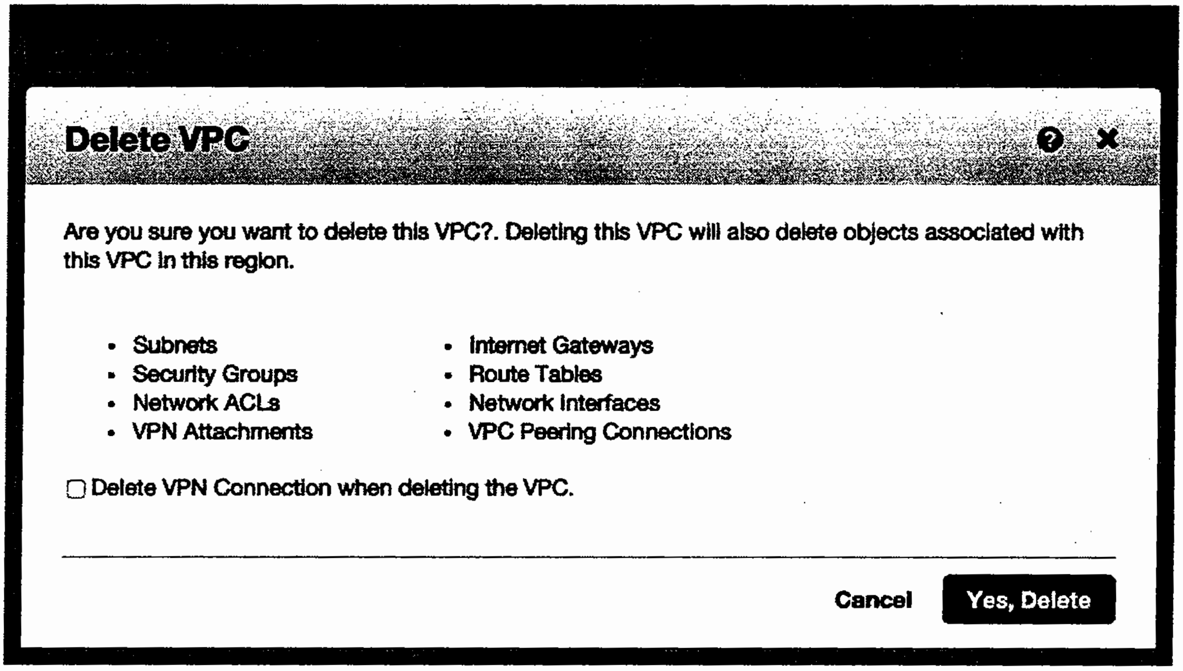


□	Name	VPC ID	State
□	-labs	vpc-a144e4c4	available

Select the VPC vepsun-labs, and then click **Actions**, and select **Delete VPC**.



If you need to delete the **VPN connection**, select the option to do so, otherwise, leave it unselected. Click **Yes, Delete.**



Delete VPC

Are you sure you want to delete this VPC?. Deleting this VPC will also delete objects associated with this VPC in this region.

- Subnets
- Security Groups
- Network ACLs
- VPN Attachments
- Internet Gateways
- Route Tables
- Network Interfaces
- VPC Peering Connections

□ Delete VPN Connection when deleting the VPC.