

## Module 7 Lab 1: Virtual Private Cloud Subnets

### VPC Dashboard

The screenshot shows the AWS VPC Dashboard. On the left, there's a navigation sidebar with sections for VPCs, Security, and Virtual Private Network (VPN). The main area displays a table of existing VPCs. The columns include Name, VPC ID, State, IPv4 CIDR, IPv6 CIDR, DHCP options set, Main Route table, Main Network ACL, and Tenancy. Two VPCs are listed:

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP options set	Main Route table	Main Network ACL	Tenancy
vpc-09b6581c2ed32f1417	vpc-09b6581c2ed32f1417	available	172.30.0.0/16	-	dept-f1b6b396	rtb-018fdf8e317474d0	acl-06273233e30b1dce4	default
vpc-56524731	vpc-56524731	available	172.31.0.0/16	-	dept-f1b6b396	rtb-385adb5e	acl-a357f0c5	default

Starting at the VPC Dashboard, select Create VPC.

### VPC - Create VPC

The screenshot shows the 'Create VPC' configuration page. It includes fields for Name tag (vpcdemo), IPv4 CIDR block (10.0.0.0/16), IPv6 CIDR block (selected as 'Amazon provided IPv6 CIDR block'), and Tenancy (Default). At the bottom, there are 'Cancel' and 'Create' buttons.

AVPC is an isolated portion of the AWS cloud populated by AWS objects, such as Amazon EC2 instances. You must specify an IPv4 address range for your VPC. Specify the IPv4 address range as a Classless Inter-Domain Routing (CIDR) block; for example, 10.0.0.0/16. You cannot specify an IPv4 CIDR block larger than /16. You can optionally associate an IPv6 CIDR block with the VPC.

Name tag: vpcdemo

IPv4 CIDR block\*: 10.0.0.0/16

IPv6 CIDR block:  Amazon provided IPv6 CIDR block

Tenancy: Default

\* Required

Create

The IPv4 CIDR block is inputted manually to reflect the internal VPC network.

## VPC Dashboard

The screenshot shows the AWS VPC Dashboard. On the left, there's a sidebar with navigation links for VPC Dashboard, Filter by VPC, Select a VPC, VIRTUAL PRIVATE CLOUD (Your VPCs: Subnets, Route Tables, Internet Gateways, Egress Only Internet Gateways, DHCP Options Sets New, Elastic IPs New, Endpoints, Endpoint Services, NAT Gateways, Peering Connections), SECURITY (Network ACLs, Security Groups New), and VIRTUAL PRIVATE NETWORK (VPN) (Customer Gateways, Virtual Private Gateways, Site-to-Site VPN). The main area has tabs for Create VPC (selected), Actions, and a search bar. A table lists existing VPCs: VPC\_1 (available, IPv4 CIDR 172.30.0.0/16, IPv6 CIDR 2600:11c:9d6:c600::/64, DHCP options set default, Main Route table rtb-018fdfc8e317474d0, Main Network ACL acl-06273233e36b1dce4, Tenancy default), vpcdemo (available, IPv4 CIDR 10.0.0.0/16, IPv6 CIDR 2600:1f1c:9d6:c600::/56, DHCP options set default, Main Route table rtb-0546f42092b2af8fa, Main Network ACL acl-09e23e26a421217ec, Tenancy default), and default (available, IPv4 CIDR 172.31.0.0/16, IPv6 CIDR 2600:1f1b:9d6:c600::/56, DHCP options set default, Main Route table rtb-385adb5e, Main Network ACL acl-a357fd0c5, Tenancy default). Below the table, a detailed view of the vpcdemo VPC is shown with its configuration: VPC ID vpc-095246e87b82cc2c9, State available, IPv4 CIDR 10.0.0.0/16, IPv6 CIDR 2600:1f1c:9d6:c600::/56, Network ACL acl-09e23e26a421217ec, DHCP options set default, Route table rtb-0546f42092b2af8fa, Tenancy default, Default VPC No, ClassicLink Disabled, DNS resolution Enabled, DNS hostnames Disabled, ClassicLink DNS Support Disabled, and Owner 75827676861.

The new VPC has been created (vpc-095246e87b82cc2c9).

## VPC Dashboard - Subnets

The screenshot shows the AWS VPC Dashboard - Subnets page. The sidebar is the same as the previous screen. The main area has tabs for Create subnet (selected), Actions, and a search bar. A table lists subnets: subnet-02c109fc3a196d183 (available, VPC VPC\_1, IPv4 CIDR 172.30.0.0/24, Available IPv4 251, IPv6 CIDR -, Availability Zone us-west-1a, Availability Zone ID usw1-az3, Route table rtb-018fdfc8e317474d0, Tenancy a), subnet-077cd1a3367eb6c3c (available, VPC VPC\_1, IPv4 CIDR 172.30.1.0/24, Available IPv4 251, IPv6 CIDR -, Availability Zone us-west-1b, Availability Zone ID usw1-az1, Route table rtb-018fdfc8e317474d0, Tenancy a), subnet-6dafbe36 (available, VPC default, IPv4 CIDR 172.31.0.0/20, Available IPv4 4091, IPv6 CIDR -, Availability Zone us-west-1a, Availability Zone ID usw1-az3, Route table rtb-385adb5e, Tenancy a), and subnet-7991551f (available, VPC default, IPv4 CIDR 172.31.16.0/20, Available IPv4 4091, IPv6 CIDR -, Availability Zone us-west-1b, Availability Zone ID usw1-az1, Route table rtb-385adb5e, Tenancy a).

The current subnets refer to other AWS projects. New subnets are required for this vpcdemo just created.

## Subnets - Create subnet

[Subnets](#) > Create subnet

### Create subnet

Specify your subnet's IP address block in CIDR format; for example, 10.0.0.0/24. IPv4 block sizes must be between a /16 netmask and /28 netmask, and can be the same size as your VPC. An IPv6 CIDR block must be a /64 CIDR block.

Name tag	10.0.1.0 - us-west-1a	<small>i</small>
VPC*	vpc-095246e87b82cc2c9	<small>i</small>
Availability Zone	No preference	<small>i</small>
VPC CIDRs		
CIDR	Status	Status Reason
10.0.0.0/16	associated	
2600:11fc:9d6:c600:/56	associated	
IPv4 CIDR block*	10.0.1.0/24	<small>i</small>
IPv6 CIDR block	Don't Assign IPv6	<small>i</small>

\* Required

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

IPv4 CIDR block manually set to 10.0.1.0/24, to AZ us-west-1a

## VPC Dashboard

[Create subnet](#) [Actions](#)

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Availability Zone	Availability Zone ID	Route table
subnet-7991551f	subnet-012a74242ca9f538d	available	vpc-56524731   default	172.31.16.0/20	4091	-	us-west-1b	usw1-az1	rtb-3d5adb5e
subnet-6fdabfe36		available	vpc-56524731   default	172.31.0.0/20	4091	-	us-west-1a	usw1-az3	rtb-385adfb6e
subnet-077cd1a3397eb6c3c		available	vpc-08b6581c2ed32f417   VPC_1	172.30.1.0/24	251	-	us-west-1b	usw1-az1	rtb-018fcfc8e317474d0
subnet.02c:109fc3a196d183		available	vpc-08b6581c2ed32f417   VPC_1	172.30.0.0/24	251	-	us-west-1a	usw1-az3	rtb-018fcfc8e317474d0
<b>10.0.1.0 - us-west-1a</b>	<b>subnet-012a74242ca9f538d</b>	<b>available</b>	<b>vpc-095246e87b82cc2c9   vpcdemo</b>	<b>10.0.1.0/24</b>	<b>251</b>	<b>-</b>	<b>us-west-1a</b>	<b>usw1-az3</b>	<b>rtb-0546f42092b2af8fa</b>

Subnet: subnet-012a74242ca9f538d

[Description](#) [Flow Logs](#) [Route Table](#) [Network ACL](#) [Tags](#) [Sharing](#)

Subnet ID	subnet-012a74242ca9f538d	State		available
VPC	vpc-095246e87b82cc2c9   vpcdemo	IPv4 CIDR		10.0.1.0/24
Available IPv4 Addresses	251	IPv6 CIDR		-
Availability Zone	us-west-1a (usw1-az3)	Route Table		rtb-0546f42092b2af8fa
Network ACL	ad-09e236268421217ec	Default subnet		No
Auto-assign public IPv4 address	No	Auto-assign IPv6 address		No
Outpost ID	-	Owner		750287676961

The internal subnet (or internal datacenter) was created inside the vpcdemo VPC.

## Subnets - Create subnet

Subnets > Create subnet

### Create subnet

Specify your subnet's IP address block in CIDR format; for example, 10.0.0.0/24. IPv4 block sizes must be between a /16 netmask and /28 netmask, and can be the same size as your VPC. An IPv6 CIDR block must be a /64 CIDR block.

Name tag  ⓘ

VPC\*  ⓘ

Availability Zone  ⓘ

VPC CIDRs	CIDR	Status	Status Reason
	10.0.0.0/16	associated	
	2600:11fc:9d6:c600:/56	associated	

IPv4 CIDR block\*  ⓘ

IPv6 CIDR block  ⓘ

\* Required Cancel

A second private subnet is created for internal use only. Note for high availability, us-west-1b was selected.

## VPC Dashboard - Subnets

Filter by tags and attributes or search by keyword

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Availability Zone	Availability Zone ID	Route table
subnet-077cd1a3367eb6c3c	available	vpc-08b6581c2ed32f417   VPC_1	172.30.1.0/24	251	-	-	us-west-1b	usw1-az1	rtb-018fcfcfe317474d0
subnet-02c109fc3a196d183	available	vpc-08b6581c2ed32f417   VPC_1	172.30.0.0/24	251	-	-	us-west-1a	usw1-az3	rtb-018fcfcfe317474d0
subnet-7915151f	available	vpc-56524731   default	172.31.16.0/20	4091	-	-	us-west-1b	usw1-az1	rtb-385adb5e
subnet-6dadbe36	available	vpc-56524731   default	172.31.0.0/20	4091	-	-	us-west-1a	usw1-az3	rtb-385adb5e
10.0.1.0 - us-west-1a	subnet-012a74242ca9f538d	available	vpc-095246e87b82cc2c9   vpcdemo	10.0.1.0/24	251	-	us-west-1a	usw1-az3	rtb-0540f42092b2af8fa
10.0.2.0 - us-west-1b	subnet-0f51a3c8f627a7772	available	vpc-095246e87b82cc2c9   vpcdemo	10.0.2.0/24	251	-	us-west-1b	usw1-az1	rtb-0540f42092b2af8fa

Subnets: subnet-012a74242ca9f538d, subnet-0f51a3c8f627a7772

By default, at this moment both subnets are private because there is no Internet Gateway (IGW) attached.

## VPC Dashboard - Create internet gateway

The screenshot shows the VPC Dashboard with the title "VPC Dashboard - Create internet gateway". At the top, there is a "Create internet gateway" button and an "Actions" dropdown menu. Below this is a search bar with the placeholder "Filter by tags and attributes or search by keyword". A table lists two existing Internet Gateways:

Name	ID	State	VPC	Owner
igw-0d062499a67...	igw-0d062499a67...	attached	vpc-08b6581c2ed32f417   VPC_1	758287676861
igw-2e29224a	igw-2e29224a	attached	vpc-66524731   default	758287676861

At the bottom of the table, a message says "Select an internet gateway above". There are also standard browser navigation buttons at the bottom right.

For VPC\_1 and default, an internet gateway is already set up.

## Internet gateways - Create internet gateway

The screenshot shows the "Create internet gateway" wizard. The first step, "Create internet gateway", is selected. It displays the following information:

- An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.
- A text input field labeled "Name tag" contains the value "igwdemo".
- A note indicates that the "Name tag" is required.
- Buttons for "Cancel" and "Create" are visible at the bottom right.

A new igw is created to support the vpcdemo VPC

## VPC Dashboard - Internet Gateways

Name	ID	State	VPC	Owner
igwdemo	igw-0cab531e67bd9b950	detached	-	758287676861
igw-0d062499a67...	igw-0d062499a67...	attached	vpc-08b6581c2ed32f417   VPC_1	758287676861
igw-2e9224a	igw-2e9224a	attached	vpc-66524731   default	758287676861

Internet gateway: igw-0cab531e67bd9b950

Description Tags

ID: igw-0cab531e67bd9b950  
State: detached  
Attached VPC ID: -  
Owner: 758287676861

Note that igwdemo is detached.

## Internet Gateways - Attach to VPC

Internet gateways > Attach to VPC

Attach to VPC

Attach an internet gateway to a VPC to enable communication with the internet. Specify the VPC you would like to attach below.

VPC\* vpc-095246e87b82cc2c9

AWS Command Line

Required

Filter by attributes

VPC ID	Name
vpc-095246e87b82cc2c9	vpcdemo

Cancel Attach

igwdemo will be attached to vpcdemo. Note that only one internet gateway can be attached to any VPC's.

## VPC Dashboard - Create internet gateway

igwdemo is now attached to vpcdemo.

## VPC Dashboard - Create route table

The screenshot shows the AWS VPC Dashboard interface for creating a route table. At the top, there is a search bar labeled "Filter by tags and attributes or search by keyword" and a table header with columns: Name, Route Table ID, Explicit subnet association, Edge associations, Main, VPC ID, and Owner. Below the table, a message states: "Route Table: rtb-0546f42092b2af8fa". The "Subnet Associations" tab is selected, showing a table with columns: Subnet ID, IPv4 CIDR, and IPv6 CIDR. A message below the table says: "You do not have any subnet associations". At the bottom, a note states: "The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:" followed by a table showing subnets: subnet-012a74242ca9f53... (10.0.1.0/24) and subnet-0f51a3c8f627a777... (10.0.2.0/24).

No associated subnets are established to the vpcdemo VPC.

## Route Tables - Create route table

Route Tables > Create route table

### Create route table

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Name tag: rtDemo

VPC: vpc-095246e87b82cc2c9

\* Required

Cancel Create

A new route table (rtDemo) is created in the vpcdemo VPC.

## VPC Dashboard - Create route table

Route Table: rtb-00376893ebd9e1c74

Summary Routes Subnet Associations Edge Associations Route Propagation Tags

Edit routes

View: All routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	active	No
2600:1f1c:9d6:c600::/56	local	active	No

There are no routes established for internet traffic at 0.0.0.0/0

## Route Tables - Edit routes

Route Tables > Edit routes

### Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	active	No
2600:111c:9d6:c600::/56	local	active	No
0.0.0.0/0	igw-0cab531e67bd9b950	active	No 
/0	igw-0cab531e67bd9b950	active	No 

[Add route](#)

\* Required

[Cancel](#) [Save routes](#)

All internet traffic (IPv4 and IPv6) will route through the igwdemo internet gateway.

## Route Tables - Edit subnet associations

Route Tables > Edit subnet associations

### Edit subnet associations

Route table rtb-00378893ebd9e1c74 (rtdemo)			
Associated subnets <a href="#">subnet-012a74242ca9f538d</a>			
<input type="text"/> Filter by attributes or search by keyword <span style="float: right;">K &lt; f to 2 of 2 &gt;  </span>			
Subnet ID	IPv4 CIDR	IPv6 CIDR	Current Route Table
<input checked="" type="checkbox"/> subnet-012a74242ca9f538d   10.0.1.0 - us-west-1a	10.0.1.0/24	-	Main
<input type="checkbox"/> subnet-0f5fa3c8fb27a7772   10.0.2.0 - us-west-1b	10.0.2.0/24	-	Main

\* Required

[Cancel](#) [Save](#)

Now the public subnet can be associated with the private subnet.

## VPC Dashboard - Create subnet

The screenshot shows the AWS VPC Dashboard with a list of subnets. A context menu is open over a specific subnet entry, with the option 'Modify auto-assign IP settings' highlighted.

Name	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Availability Zone	Availability Zone ID	Route table	
367e76c3c	vpc-08b6581c2ed32417   VPC_1	172.30.1.0/24	251	-	us-west-1b	usw1-az1	rtb-018fcfcf6e317474d0	
aa196d183	vpc-08b6581c2ed32417   VPC_1	172.30.0.0/24	251	-	us-west-1a	usw1-az3	rtb-018fcfcf0e317474d0	
	vpc-56524731   default	172.31.16.0/20	4091	-	us-west-1b	usw1-az1	rtb-385adfb6e	
	vpc-56524731   default	172.31.0.0/20	4091	-	us-west-1a	usw1-az3	rtb-385adfb6e	
<input checked="" type="checkbox"/> 10.0.1.0 - us-west-1a	subnet-012a74242ca9f538d	vpc-095246e87b82cc2:9   vpcdemo	10.0.1.0/24	251	-	us-west-1a	usw1-az3	rtb-00376893ebd9e1c74   rtb-00376893ebd9e1c74   r
<input type="checkbox"/> 10.0.2.0 - us-west-1b	subnet-0f51a3c8f627a7772	vpc-095246e87b82cc2:9   vpcdemo	10.0.2.0/24	251	-	us-west-1b	usw1-az1	rtb-054642092b2a8ffa

Subnet: subnet-012a74242ca9f538d

Description Flow Logs Route Table Network ACL Tags Sharing

Subnet ID: subnet-012a74242ca9f538d  
VPC: vpc-095246e87b82cc2:9 | vpcdemo  
Available IPv4 Addresses: 251  
Availability Zone: us-west-1a (usw1-az3)  
Network ACL: ad-09e23e26a421217ec  
Auto-assign public IPv4 address: No  
Output Port ID: -  
Owner: 758287676861

The public facing subnet needs to be allowed a public IP address.

## Subnets - Modify auto-assign IP settings

Subnets > Modify auto-assign IP settings

### Modify auto-assign IP settings

Enable the auto-assign IP address setting to automatically request a public IPv4 or IPv6 address for an instance launched in this subnet. You can override the auto-assign IP settings for an instance at launch time.

Subnet ID: subnet-012a74242ca9f538d

Auto-assign IPv4  Enable auto-assign public IPv4 address

\* Required

Cancel Save

Now when EC2 instances are launched, auto assigned public IPv4 addresses can be established.

## VPC Dashboard - Create subnet

Subnet: subnet-012a74242ca9f538d

Available IPv4	IPv6 CIDR	Availability Zone	Availability Zone ID	Route table	Network ACL	Default subnet	Auto-assign public IPv4 address	Auto-assign IPv6	Owner
251	-	us-west-1b	usw1-az1	rtb-018dfc8e317474d0	aci-06273233e36b1dce4	No	Yes	No	758287i
251	-	us-west-1a	usw1-az3	rtb-018dfc8e317474d0	aci-06273233e36b1dce4	No	Yes	No	758287i
4091	-	us-west-1b	usw1-az1	rtb-385adb5e	aci-a3570c5	Yes	Yes	No	758287i
4091	-	us-west-1a	usw1-az3	rtb-385adb5e	aci-a3570c5	Yes	Yes	No	758287i
251	-	us-west-1a	usw1-az3	rtb-00376893ebd9e1c74   rtde...	aci-09e23e26a421217ec	No	Yes	No	758287i
251	-	us-west-1b	usw1-az1	rtb-054042092b2af8fa	aci-09e23e26a421217ec	No	No	No	758287i

IPv4 addresses will now be assigned automatically for EC2 instances.

## EC2 - AMI

1. Choose AMI   2. Choose Instance Type   3. Configure Instance   4. Add Storage   5. Add Tags   6. Configure Security Group   7. Review   Cancel and Exit

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.

Search for an AMI by entering a search term e.g. "Windows"

Quick Start

My AMIs	Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-061cc1f0bc2c28943f Amazon Linux Free tier eligible	State: available IPv4 CIDR: 10.0.1.0/24 IPv6 CIDR: - Route Table: rtb-00376893ebd9e1c74   rdemo Default subnet: No Auto-assign IPv6 address: No Owner: 7582876765861
AWS Marketplace	Amazon Linux 2018.03.0 (HVM), SSD Volume Type - ami-0d3caf10672b8e870 Amazon Linux Free tier eligible	Select 64-bit (x86)
Community AMIs	Amazon Linux 2018.03.0 (HVM), SSD Volume Type - ami-0d3caf10672b8e870 Amazon Linux Free tier eligible	Select 64-bit (x86)
<input checked="" type="checkbox"/> Free tier only	Red Hat Enterprise Linux 8 (HVM), SSD Volume Type - ami-066df92ac6f03efca Red Hat Free tier eligible	Select 64-bit (x86)
	SUSE Linux Enterprise Server 15 SP1 (HVM), SSD Volume Type - ami-02d732ce729636ebb SUSE Linux Free tier eligible	Select 64-bit (x86)
	Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-0f56279347d2fa43e Ubuntu	Select

To utilize the subnets and VPC, two instances will be started. Linux AMI 2018.03.0 is selected.

## EC2 - Instance type

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 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.

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	IPv6 Support	
General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes	
General purpose	<b>t2.micro</b> <span style="background-color: #00AEEF; color: white;">Free tier eligible</span>	1	1	EBS only	-	Low to Moderate	Yes	
General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes	
General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes	
General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes	
General purpose	t2.xlarge	4	16	EBS only	-	Moderate	Yes	
General purpose	t2.2xlarge	8	32	EBS only	-	Moderate	Yes	
General purpose	t3a.nano	2	0.5	EBS only	Yes	Up to 5 Gigabit	Yes	
General purpose	t3a.micro	2	1	EBS only	Yes	Up to 5 Gigabit	Yes	
General purpose	t3a.small	2	2	EBS only	Yes	Up to 5 Gigabit	Yes	
General purpose	t3a.medium	2	4	EBS only	Yes	Up to 5 Gigabit	Yes	

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Instance Details](#)

**t2.micro is selected.**

## EC2 - Configure Instance Details

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 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	<input type="text" value="1"/>	Launch into Auto Scaling Group
Purchasing option	<input type="checkbox"/> Request Spot instances	
Network	Network: <input type="text" value="vpc-095246e87b82cc2c9   vpcdemo"/> <a href="#">Create new VPC</a> Subnet: <input type="text" value="subnet-012a74242ca9f538d   10.0.1.0 - us-west-1a"/> <a href="#">Create new subnet</a> 251 IP Addresses available	
Auto-assign Public IP	<input type="checkbox"/> Use subnet setting (Enable)	
Placement group	<input type="checkbox"/> Add instance to placement group	
Capacity Reservation	<input type="checkbox"/> Open <a href="#">Create new Capacity Reservation</a>	
IAM role	<input type="text" value="None"/> <a href="#">Create new IAM role</a>	
Shutdown behavior	<input type="radio"/> Stop <input type="radio"/> Stop - Hibernate	
Stop - Hibernate behavior	<input type="checkbox"/> Enable hibernation as an additional stop behavior <input type="checkbox"/> Protect against accidental termination	
Enable termination protection	<input type="checkbox"/>	
Monitoring	<input type="checkbox"/> Enable CloudWatch detailed monitoring Additional charges apply	
Tenancy	<input type="radio"/> Shared - Run a shared hardware instance Additional charges will apply for dedicated tenancy	
T2/T3 Unlimited	<input type="checkbox"/> Enable <small>Additional charges may apply</small>	

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

**Note that vpcdemo and the public facing subnet 10.0.1.0 is selected.**

## EC2 - Add Storage

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

### Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more about storage options in Amazon EC2](#)

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/xvda	snap-00a5302a9e1c67d18	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more about free usage tier eligibility and usage restrictions](#)

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Tags](#)

Default storage is selected.

## EC2 - Add Tags

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

### Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.  
A copy of a tag can be applied to volumes, instances or both.  
Tags will be applied to all instances and volumes. [Learn more about tagging your Amazon EC2 resources](#).

Key	(128 characters maximum)	Value	(256 characters maximum)	Instances	Volumes
depart		dev project1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

[Add another tag](#) (Up to 50 tags maximum)

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Security Group](#)

Tags defined.

## EC2 - Configure Security Group

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

**Step 6: Configure Security Group**

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. Learn more about Amazon EC2 security groups.

Assign a security group:  Create a new security group  Select an existing security group

Security group name: vpc-sgdemo  
Description: vpc-sgdemo

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop
HTTP	TCP	80	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop
HTTPS	TCP	443	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop

Add Rule

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

Cancel Previous Review and Launch

Security group created with SSH for administration and HTTP/HTTPS for public accessible web services. However, for this lab only SSH is required.

## EC2 - Review Instance Launch

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

**Step 7: Review Instance Launch**

Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-0d3caf10672b8e870

**Free tier eligible** The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.  
Root Device Type: ebs Virtualization type: hvm

**Instance Type** [Edit instance type](#)

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

**Security Groups** [Edit security groups](#)

Security group name: vpc-sgdemo  
Description: vpc-sgdemo

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	0.0.0.0/0	
HTTP	TCP	80	0.0.0.0/0	
HTTP	TCP	80	/0	
HTTPS	TCP	443	0.0.0.0/0	
HTTPS	TCP	443	/0	

**Instance Details** [Edit instance details](#)

**Storage** [Edit storage](#)

**Tags** [Edit tags](#)

Cancel Previous Launch

Note HTTP/HTTPS show up twice for IPv4 and IPv6 addressing.

## EC2 - Keypair

The screenshot shows the 'Step 7: Review Instance Launch' page. A modal window titled 'Select an existing key pair or create a new key pair' is displayed. In the modal, a dropdown menu shows 'Create a new key pair' selected. The key pair name 'vpc-kpdemo' is entered in the input field, and a 'Download Key Pair' button is visible. The main review page shows the following details:

- Instance Type:** t2.micro
- Security Groups:** vpc-sgdemo
- Network:** SSH (TCP), HTTP (TCP), HTTPS (TCP)

Buttons at the bottom include 'Cancel', 'Launch Instances', and 'Launch'.

A new keypair has been created and downloaded locally.

## EC2 Dashboard

The screenshot shows the EC2 Dashboard with a single instance listed:

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP	IPv6 IPs	Key Name	Monitoring	Last
EC2 Public Subnet	i-0e98bcf502e0b5ea	t2.micro	us-west-1a	running	Initializing	None	50.18.234.133	-	-	vpc-kpdemo	disabled	May

Details for the instance 'i-0e98bcf502e0b5ea' (EC2 Public Subnet) are shown:

Description	Value
Instance ID	i-0e98bcf502e0b5ea
Instance state	running
Instance type	t2.micro
Finding	Opt-in to AWS Compute Optimizer for recommendations. Learn more
Private DNS	ip-10-0-1-166.us-west-1.compute.internal
Private IPs	10.0.1.166
Public DNS (IPv4)	50.18.234.133
IPv4 Public IP	50.18.234.133
IPv6 IPs	-
Elastic IPs	-
Availability zone	us-west-1a
Security groups	vpc-sgdemo, view inbound rules, view outbound rules
CloudWatch Metrics	Metrics for CloudWatch Metrics

Instance has been successfully created. Another private instance is now required.

## EC2 - Configure Instance

1. Choose AMI   2. Choose Instance Type   3. Configure Instance   4. Add Storage   5. Add Tags   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	<input type="text" value="1"/>	Launch into Auto Scaling Group
Purchasing option	<input type="checkbox"/> Request Spot instances	
Network	vpc-095246e87b82cc2c9   vpcdemo <input type="button" value="Create new VPC"/>	
Subnet	subnet-0f51a3c8f627a7772   10.0.2.0 - us-west-1b <input type="button" value="Create new subnet"/> 251 IP Addresses available	
Auto-assign Public IP	<input type="button" value="Use subnet setting (Disable)"/>	
Placement group	<input type="checkbox"/> Add instance to placement group	
Capacity Reservation	<input type="button" value="Open"/>	<input type="button" value="Create new Capacity Reservation"/>
IAM role	<input type="button" value="None"/> <input type="button" value="Create new IAM role"/>	
Shutdown behavior	<input type="button" value="Stop"/>	
Stop - Hibernate behavior	<input type="checkbox"/> Enable hibernation as an additional stop behavior	
Enable termination protection	<input type="checkbox"/> Protect against accidental termination	
Monitoring	<input type="checkbox"/> Enable CloudWatch detailed monitoring Additional charges apply	
Tenancy	<input type="button" value="Shared - Run a shared hardware instance"/> Additional charges will apply for dedicated tenancy.	
T2/T3 Unlimited	<input type="checkbox"/> Enable Additional charges may apply	

**Cancel** **Previous** **Review and Launch** **Next: Add Storage**

However, this instance will be set to 10.0.2.0, the private subnet in the vpcdemo VPC.

## EC2 - Configure Security Group

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

**Step 6: Configure Security Group**

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. Learn more about Amazon EC2 security groups.

**Assign a security group:**  Create a new security group  Select an existing security group

**Security group name:**

**Description:**

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom <input type="button" value="0.0.0.0/0"/>	e.g. SSH for Admin Desktop
HTTP	TCP	80	Custom <input type="button" value="0.0.0.0/0, ::/0"/>	e.g. SSH for Admin Desktop
HTTPS	TCP	443	Custom <input type="button" value="0.0.0.0/0, ::/0"/>	e.g. SSH for Admin Desktop

**Add Rule**

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

**Cancel** **Previous** **Review and Launch**

The instance is given HTTP/HTTPS and SSH port access.

## EC2 - Review

Step 7: Review Instance Launch

AMI Details

**Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-0d3caf10672b8e870**

Free tier eligible Root Device Type: ebs Virtualization type: hvm

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

Security Groups

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	0.0.0.0/0	
HTTP	TCP	80	0.0.0.0/0	
HTTP	TCP	80	::/0	
HTTPS	TCP	443	0.0.0.0/0	
HTTPS	TCP	443	::/0	

Instance Details

Storage

Cancel Previous Launch

All settings match the public subnet except for the subnet itself.

## EC2 - Keypair

Step 7: Review Instance Launch

**Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-0d3caf10672b8e870**

Free tier eligible Root Device Type: ebs Virtualization type: hvm

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)
t2.micro	Variable	1	1

Security Groups

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	0.0.0.0/0	
HTTP	TCP	80	0.0.0.0/0	
HTTP	TCP	80	::/0	
HTTPS	TCP	443	0.0.0.0/0	
HTTPS	TCP	443	::/0	

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.

Choose an existing key pair Select a key pair vpc-kpdemo

I acknowledge that I have access to the selected private key file (vpc-kpdemo.pem), and that without this file, I won't be able to log into my instance.

Cancel Launch Instances

Instance Details

Storage

Tags

Cancel Previous Launch

For brevity, the same keypair used for the public subnet is selected.

## EC2 Dashboard

EC2 Dashboard showing two instances:

- EC2 Private Subnet**: Instance ID i-068b3ae3df0115132, t2.micro, us-west-1b, running, Status Checks Initializing, Public DNS (IPv4) -, IPv4 Public IP -, IPv6 IPs -, Key Name vpc-kpdemo, Monitoring disabled, Last Activity May.
- EC2 Public Subnet**: Instance ID i-0e98bcf502e0b5ea, t2.micro, us-west-1a, running, Status Checks 2/2 checks ..., Public DNS (IPv4) 50.18.234.133, IPv4 Public IP 50.18.234.133, IPv6 IPs -, Key Name vpc-kpdemo, Monitoring disabled, Last Activity May.

A private subnet is now active. Note that a public IPv4 IP address has not been assigned.

## Security Groups - Edit inbound rules - sgdemo2

Edit inbound rules [Info](#)

Inbound rules control the incoming traffic that's allowed to reach the instance.

Type	Protocol	Port range	Source	Description - optional
SSH	TCP	22	Custom 0.0.0.0/0	

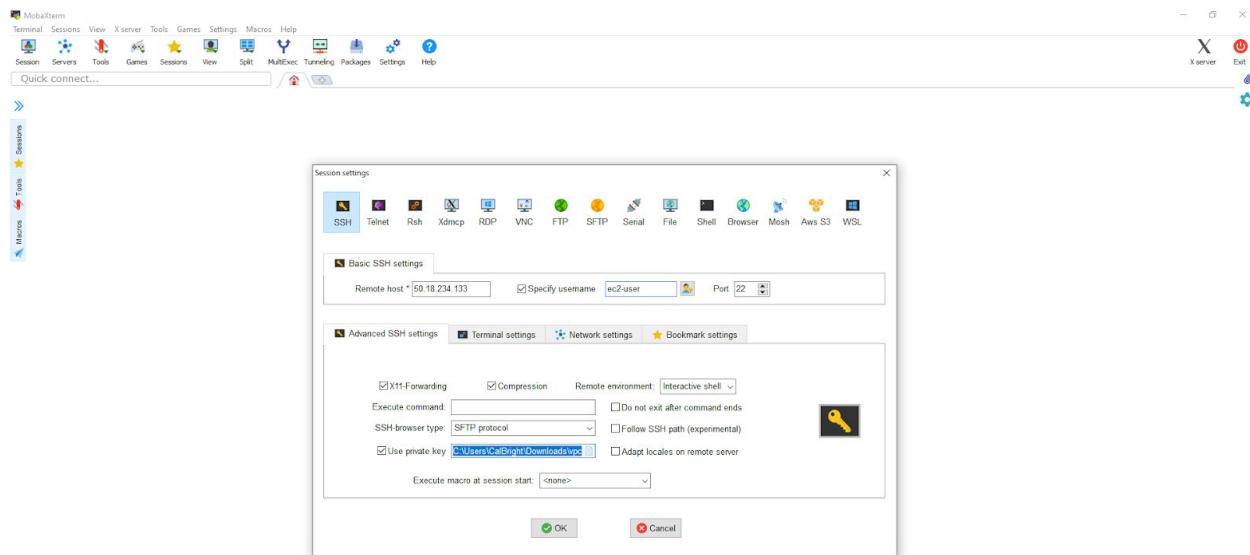
[Add rule](#)

**NOTE:** Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

[Cancel](#) [Preview changes](#) [Save rules](#)

For best practices, unused ports are closed.

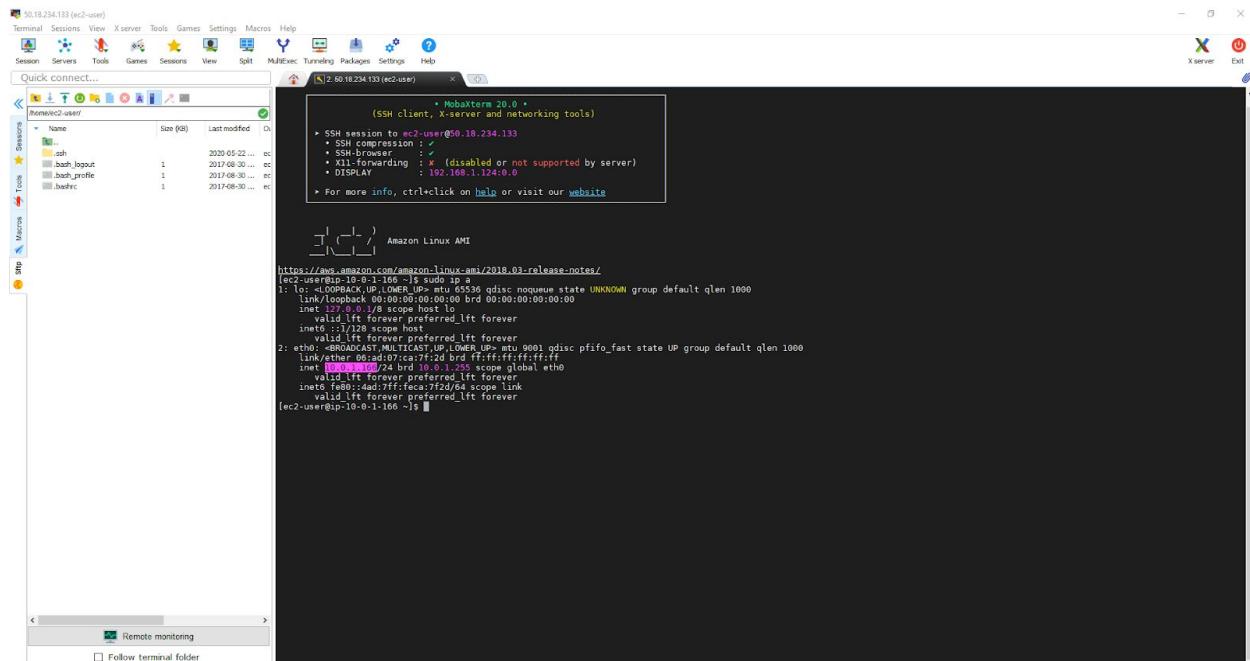
## MobaXterm



UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

MobaXterm is used to SSH into the public facing subnet attached to the vpcdemo VPC.

## MobaXterm



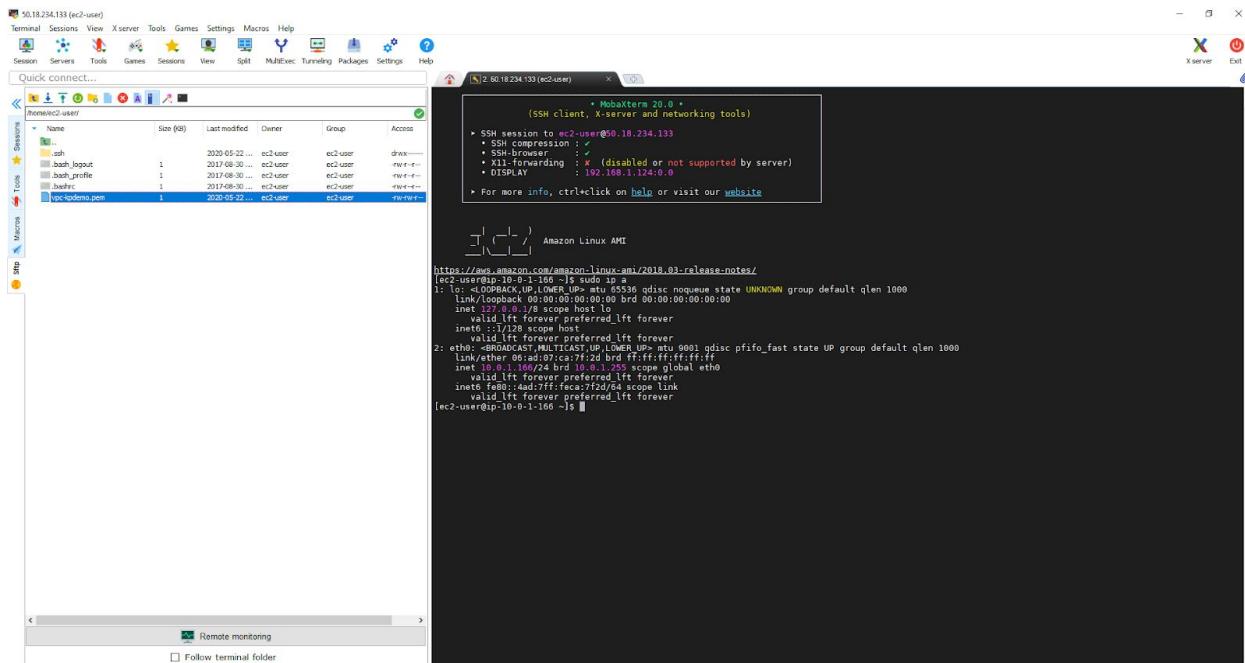
Note the correct public subnet IPv4 address has been assigned, 10.0.1

## Windows 10 - Local Folder

Name	Date modified	Type	Size
▼ Today (1)			
vpc-kpdemo.pem	5/22/2020 7:41 PM	PEM File	2 KB
↓ Drag and drop files here ↓			

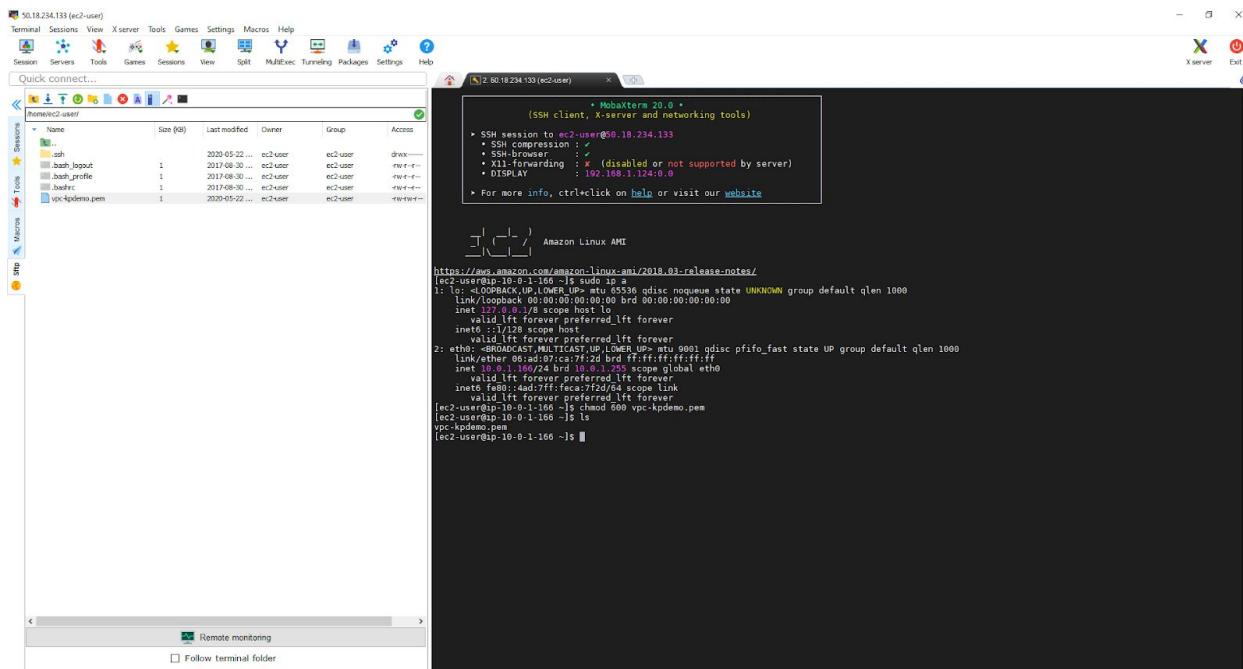
The pem keypair must be copied over to the EC2 instance home directory. This is accomplished by dragging and dropping the pem keypair into the /home/ec2-user/ sidebar in MobaXterm.

## MobaXterm



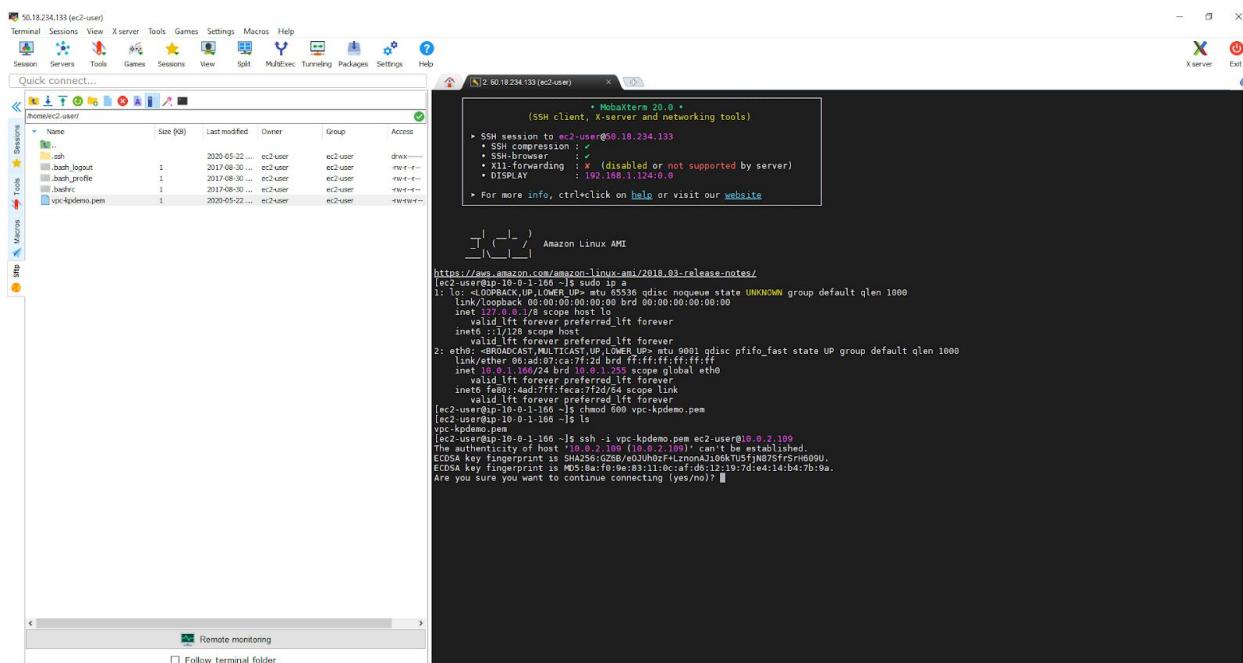
Note the vpc-kpdemo.pem file is now in the home directory.

## MobaXterm



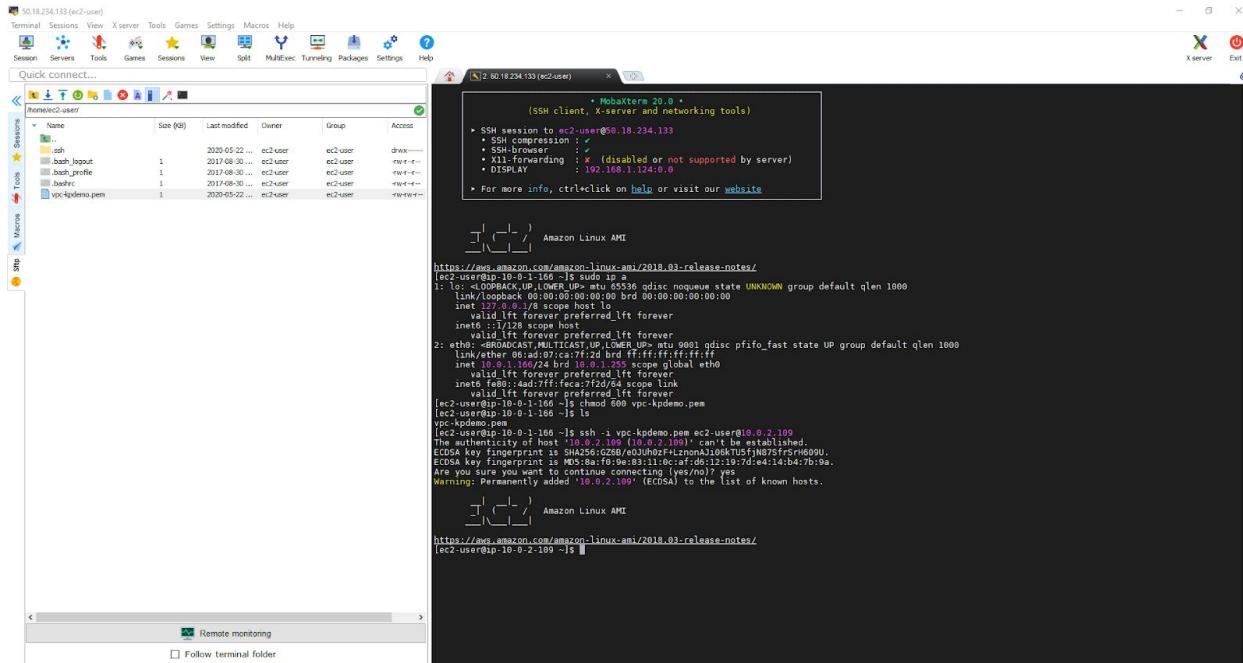
Permissions of the pem file must be changed to 600 before an SSH connection can be established to the private subnet.

## MobaXterm



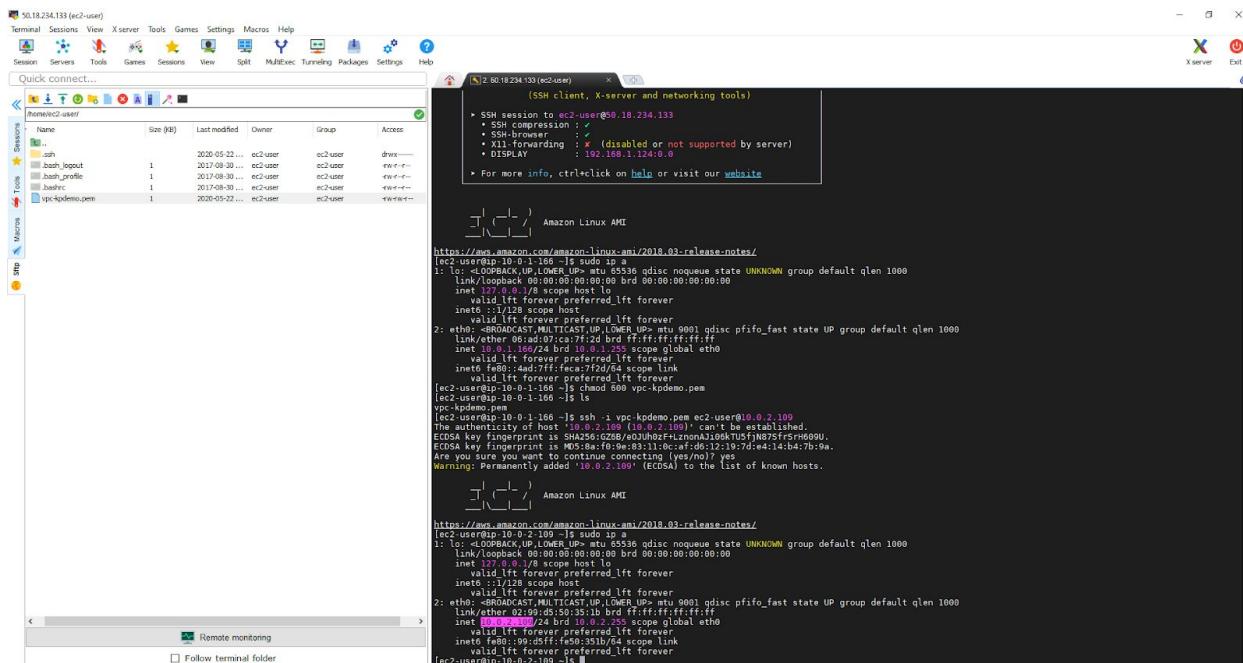
A warning that the private IPv4 address cannot be authenticated.

## MobaXterm



Successful connection to the private subnet through the public subnet.

## MobaXterm



The IP is now consistent with the private IPv4 created in the vpcdemo VPC.

## VPC Dashboard - Network ACLs

The screenshot shows the AWS VPC Network ACLs dashboard. At the top, there's a search bar and a table listing network ACLs. One ACL, 'acl-09e23e26a421217ec', is selected and shown in more detail below.

**Selected Network ACL Details:**

- Name: acl-09e23e26a421217ec
- Network ACL ID: aclo9e23e26a421217ec
- Associated with: 2 Subnets
- Default: Yes
- VPC: vpc-095246e87b82cc2c9 | vpcdemo
- Owner: 758287676861

**Outbound Rules Tab:**

Under the Outbound Rules tab, there are four rules listed:

Rule #	Type	Protocol	Port Range	Destination	Allow / Deny
100	ALL Traffic	ALL	ALL	0.0.0.0/0	ALLOW
101	ALL Traffic	ALL	ALL	::/0	ALLOW
*	ALL Traffic	ALL	ALL	0.0.0.0/0	DENY
*	ALL Traffic	ALL	ALL	::/0	DENY

A new Access Control List (NACL-WebDemo) will close the default open ports.

## Network ACLs - Create network ACL

The screenshot shows the 'Create network ACL' wizard. Step 1: Set basic information.

Fields filled in:

- Name tag: NACL-WebDemo
- VPC: vpc-095246e87b82cc2c9

Buttons at the bottom: 'Cancel' and 'Create'.

As default all ports will be in the DENY state.

## VPC Dashboard - Network ACLs

The screenshot shows the AWS VPC Network ACLs dashboard. At the top, there is a search bar and a table listing Network ACLs. One row is selected, showing details for 'NACL-WebDemo'. Below the table, tabs for 'Details', 'Inbound Rules', 'Outbound Rules' (which is selected), and 'Tags' are visible. Under 'Outbound Rules', a table shows two rules, both of which are set to 'DENY'. The first rule covers all traffic (Type: ALL Traffic, Protocol: ALL, Port Range: ALL) and the second covers port 3306 (Type: ALL Traffic, Protocol: ALL, Port Range: 3306).

Name	Network ACL ID	Associated with	Default	VPC	Owner
NACL-WebDemo	acl-003bdec67aeaf0549	-	No	vpc-095246e87b82c2c9   vpcdemo	758287676861
	acl-09273233e36b...	2 Subnets	Yes	vpc-08b6581c2ed32f417   VPC_1	758287676861
	acl-09e23e26a421...	2 Subnets	Yes	vpc-095246e87b82c2c9   vpcdemo	758287676861
	acl-a357ff0c5	2 Subnets	Yes	vpc-56524731   default	758287676861

Rule #	Type	Protocol	Port Range	Destination	Allow / Deny
*	ALL Traffic	ALL	ALL	0.0.0.0/0	DENY
*	ALL Traffic	ALL	ALL	:3306	DENY

All Traffic is now in the DENY state for Outbound Rules.

## Network ACLs - Edit inbound rules

The screenshot shows the 'Edit inbound rules' interface for the Network ACL 'acl-003bdec67aeaf0549'. It displays three rules allowing traffic on ports 80, 443, and 22 respectively. Each rule is set to 'ALLOW' and has a delete icon next to it. At the bottom, there are buttons for 'Add Rule', 'Required', 'Cancel', and 'Save'.

Rule #	Type	Protocol	Port Range	Source	Allow / Deny
100	HTTP (80)	TCP (6)	80	0.0.0.0/0	ALLOW
200	HTTPS (443)	TCP (6)	443	0.0.0.0/0	ALLOW
300	SSH (22)	TCP (6)	22	0.0.0.0/0	ALLOW

Inbound rules are edited to reflect the required ports.

## Network ACLs - Edit outbound rules

Network ACLs > Edit outbound rules

Edit outbound rules

Network ACL acl-003bdec67aea0f0549

Rule #	Type	Protocol	Port Range	Destination	Allow / Deny
100	HTTP (80)	TCP (6)	80	0.0.0.0/0	ALLOW
200	HTTPS (443)	TCP (6)	443	0.0.0.0/0	ALLOW
300	SSH (22)	TCP (6)	22	0.0.0.0/0	ALLOW

[Add Rule](#)

\* Required

[Cancel](#) [Save](#)

Outbound rules are now set.

## VPC Dashboard - Network ACL

Create network ACL Actions ▾

Filter by tags and attributes or search by keyword

Name	Network ACL ID	Associated with	Default	VPC	Owner
NACL-WebDemo	acl-003bdec67aea0f0549	-	No	vpc-095246e67b82cc2c9   vpcdemo	758287676981
acl-06273233e36b...	2 Subnets	Yes	vpc-08b6581c2ed32f417   VPC_1	758287676981	
acl-09e23e26a421...	2 Subnets	Yes	vpc-095246e67b82cc2c9   vpcdemo	758287676981	
acl-a357f0c5	2 Subnets	Yes	vpc-56524731   default	758287676981	

Network ACL: acl-003bdec67aea0f0549

Details Inbound Rules Outbound Rules Subnet associations Tags

Edit inbound rules

View All rules

Rule #	Type	Protocol	Port Range	Source	Allow / Deny
100	HTTP (80)	TCP (6)	80	0.0.0.0/0	ALLOW
200	HTTPS (443)	TCP (6)	443	0.0.0.0/0	ALLOW
300	SSH (22)	TCP (6)	22	0.0.0.0/0	ALLOW
*	All Traffic	All	All	0.0.0.0/0	DENY
*	All Traffic	All	All	::/0	DENY

Inbound rules set for NACL-WebDemo

## VPC Dashboard - Network ACL

The screenshot shows the AWS VPC Network ACL dashboard. At the top, there are tabs for 'Create network ACL' and 'Actions'. Below is a search bar and a table listing Network ACLs. The table has columns: Name, Network ACL ID, Associated with, Default, VPC, and Owner. One row is selected, showing 'NACL-WebDemo' with 'acl-003bdec67aeaf0549' as the ID, associated with 'vpcdemo' VPC, and owned by '758287676961'. Below the table, it says 'Network ACL: acl-003bdec67aeaf0549'. Underneath are tabs for 'Details', 'Inbound Rules', 'Outbound Rules' (which is selected), 'Subnet associations', and 'Tags'. The 'Edit outbound rules' button is visible. The 'Outbound Rules' table has columns: Rule #, Type, Protocol, Port Range, Destination, and Allow / Deny. It lists five rules: rule 100 (HTTP 80), rule 200 (HTTPS 443), rule 300 (SSH 22), and two wildcard rules (\* ALL Traffic). All rules have 'Allow' status.

Rule #	Type	Protocol	Port Range	Destination	Allow / Deny
100	HTTP (80)	TCP (6)	80	0.0.0.0/0	ALLOW
200	HTTPS (443)	TCP (6)	443	0.0.0.0/0	ALLOW
300	SSH (22)	TCP (6)	22	0.0.0.0/0	ALLOW
*	All Traffic	ALL	ALL	0.0.0.0/0	DENY
*	All Traffic	ALL	ALL	/0	DENY

Outbound rules set for NACL-WebDemo