# Lab

## Introduction

In this lab, you will configure an Amazon Web Services environment. First, you will create an Amazon Simple Storage Service (Amazon S3) bucket to host a publicly accessible website. Next, you will build a custom Virtual Private Cloud (VPC) that will support public and private workloads. Finally, you will create a re-usable Amazon Elastic Compute Cloud (Amazon EC2) solution by using a custom AMI and a launch template.

## Configure an Amazon S3 bucket.

In this exercise, you will configure an S3 bucket. First, you will create an S3 bucket. Next, you will configure public access to the bucket. Finally, you will enable a static website for the bucket.

An [Amazon S3 bucket](https://docs.aws.amazon.com/AmazonS3/latest/dev/UsingBucket.html) is an object-based storage repository. The files are accessible from the internet by using a traditional HTML URL string. Objects have a maximum size of 5 TB each with no limit on total storage.

### **Create an S3 bucket.**

In this task, you will create an S3 bucket.

1. In the AWS Management Console, on the command bar, ensure that the selected region is **Ohio**.

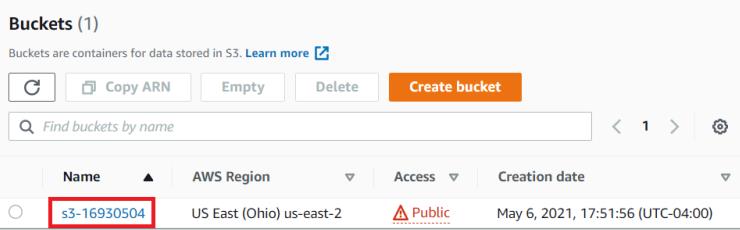
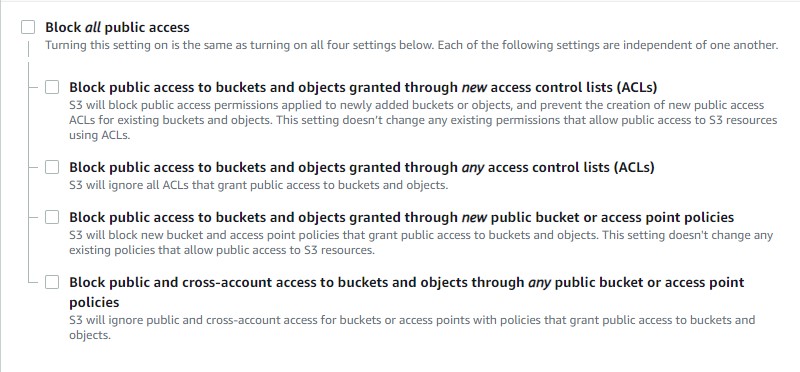
If US East (Ohio) is not selected as the region, you will not see the appropriate VPCs in the steps that follow.

1. In the AWS Management command bar, in Search, enter S3, and then in Services, select **S3**.
2. In the S3 Management Console, select **Create bucket**.
3. In Bucket name, enter s3-42446610, and then in AWS Region, ensure that **US East (Ohio) us-east-2** is displayed.
4. In Object Ownership, select **ACLs enabled**.
5. In Bucket Versioning, select **Enable**.
6. Select **Create bucket**.

Bucket versioning is not required if you are using an S3 bucket as a static website, but it is considered good practice for recovery, and auditing. You can also configure this after you create the bucket if requirements change.

### **Configure public access to an S3 bucket.**

In this task, you will configure public access to the contents of an S3 bucket.

1. On the Buckets page, select the **s3-42446610** bucket.
2. On the Permissions tab, in Block public access (bucket settings), select **Edit**.
3. Clear the **Block all public access** check box, and then select **Save changes**.
4. When prompted, in the Edit Block public access (bucket settings) window, enter confirm, and then select **Confirm**.

**Public access**—sometimes referred to as anonymous access—is disabled on S3 buckets by default. To configure targeted public access, you must completely unblock public access, configure security on the appropriate objects, and then block public access for any new content or policies.

1. In Bucket policy, select **Edit**.
2. In the Policy editor, enter the following policy definition code, and then in the lower-right corner, select **Save changes**:

**JSON**

{

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Principal": "\*",

"Action": "s3:\*",

"Resource": "arn:aws:s3:::s3-42446610"

}

]

}

The **Network Failure - verify permissions to access-analyzer:ValidatePolicy in IAM to use policy validation** error can be safely ignored.

1. Open a new browser tab, go to https://github.com/LODSContent/ChallengeLabs\_Resources/raw/master/AWS-TE.1/AWS\_TE.pdf, and then save the file as AWS\_TE.pdf on your local computer.
2. Go to https://github.com/LODSContent/ChallengeLabs\_Resources/raw/master/AWS-TE.1/expert-logo.jpg, and then save the file as expert-logo.jpg on your local computer.
3. On your local computer, create a text file named index.html, and then add the following content:

**HTML**

<html lang="en">

<head>

<meta charset="utf-8" />

<title>Hello World!</title>

<link rel="stylesheet" href="hw.css">

</head>

<body>

<div class="mainBox">

<img src="https://s3-42446610.s3.us-east-2.amazonaws.com/expert-logo.jpg" alt="LODS Expert Challenge Logo">

<div class="textBox">

<h1>Hello World!</h1>

More information about AWS Certification and Training can be found here:

<a href="https://s3-42446610.s3.us-east-2.amazonaws.com/AWS\_TE.pdf">AWS Certification and Training Overview</a>

</div>

</div>

</body>

</html>

1. On your local computer, create a text file named hw.css, and then add the following content:

**CSS**

body {

background-color: #36a;

}

.mainBox {

width: 100%;

display: flex;

flex-direction: column;

align-items: center;

}

.textBox {

width: 30%;

text-align: center;

}

h1 {

color: #fff;

}

img {

width: 50%;

border: 1px solid #fff;

}

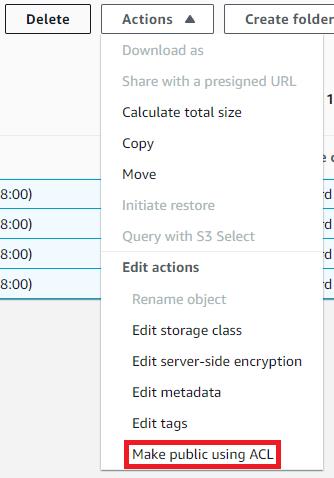
1. On the Objects tab, select **Upload**.
2. In Files and folders, select **Add files**, select the **index.html**, **hw.css**, **AWS\_TE.pdf**, and **expert-logo.jpg** files, and then select **Open**.

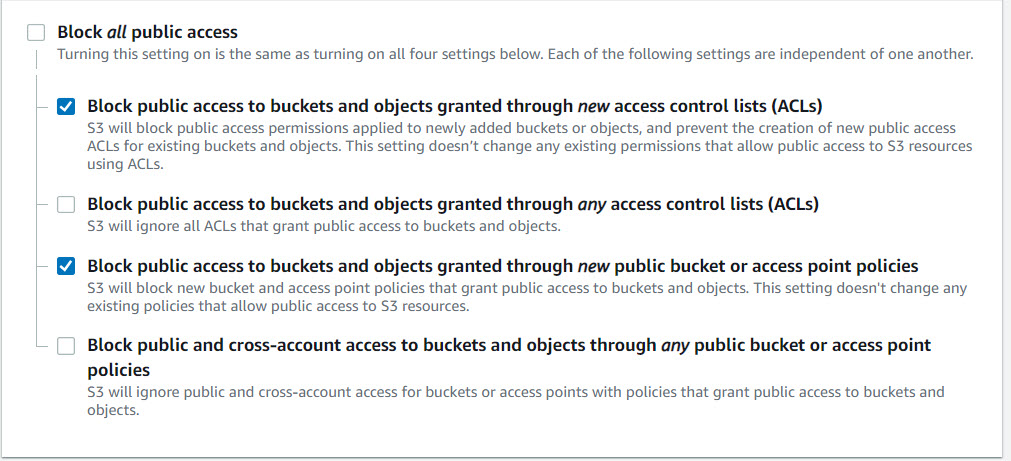
If you put the files in different locations, you may need to perform this step once for each location, selecting the appropriate files each time.

1. In the lower-right corner, select **Upload**.
2. When the upload is completed, review the information on the page, and then select **Close**.

If there are any failures listed, make note of which files failed, and then attempt to upload those files again.

1. Select the check boxes for all four objects, and then on the **Actions** menu, select **Make public using ACL**.

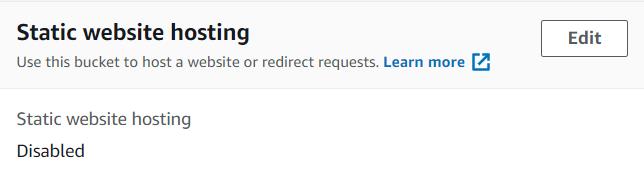


1. On the Make public page, verify that the four files are listed, and then select **Make public**.
2. On the Make public: status page, verify that there are no failures listed, and then select **Close**.
3. On the Permissions tab, in Block public access (bucket settings), select **Edit**.
4. Select **Block public access to buckets and objects granted through new access control lists (ACLs)** and **Block public access to buckets and objects granted through new public bucket or access point policies**, and then select **Save changes**.
5. In the Edit Block public access (bucket settings) window, enter confirm, and then select **Confirm**.

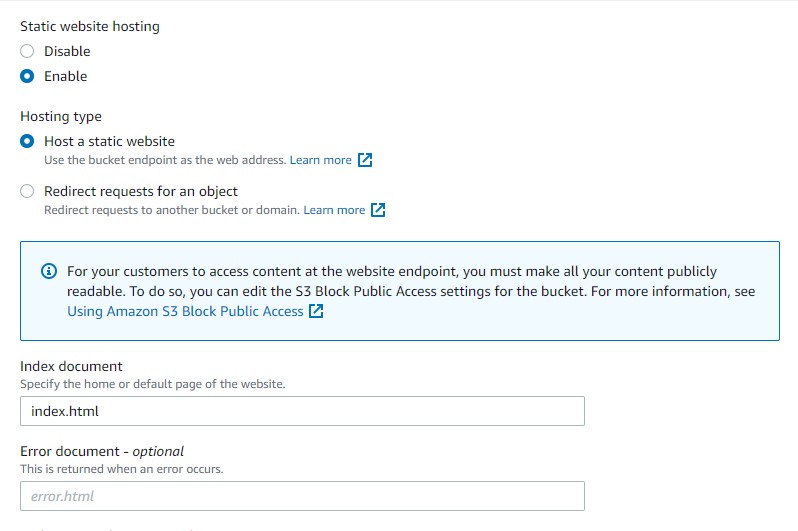
### **Create a static website by using an S3 bucket.**

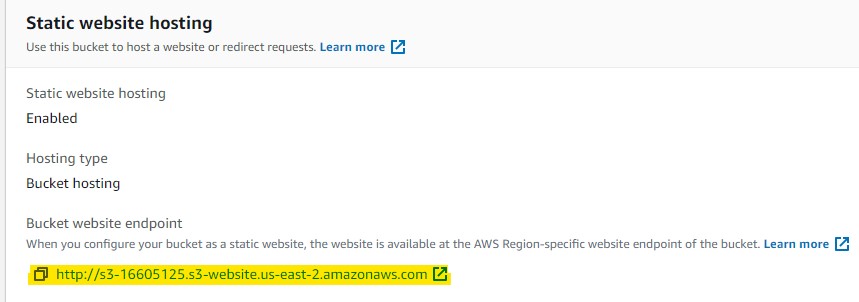
In this task, you will create a static website by using an S3 bucket.

1. On the s3-42446610 page, select the **Properties** tab, and then in Static website hosting, select **Edit**.



1. On the Edit static website hosting page, in Static website hosting, select **Enable**.
2. In Hosting type, ensure that **Host a static website** is selected, and then in Index document, enter index.html.



1. Select **Save changes**.
2. In Static website hosting, in Bucket website endpoint, select the link to open the website in a new browser tab.

The website should display the uploaded graphic and the link to the PDF file.

## Configure a VPC

In this exercise, you will configure a VPC. First, you will create a VPC. and then you will add subnets to the VPC.

A VPC is a virtual representation of a physical network environment. The VPC establishes the network in which you will host AWS resources. The VPC provides the TCP/IP address range of the overall network. Within the VPC, you create subnets as a subset of the overall TCP/IP address range.

### **Create a VPC**

1. On the AWS command bar, in Search, enter VPC, and then in Services, select **VPC**.
2. On the VPC dashboard navigation pane, in Virtual private cloud, select **Your VPCs**
3. On the Your VPCs page, select **Create VPC**.
4. On the Create VPC page, in VPC settings, In Name tag, enter vpc-42446610.
5. In IPv4 CIDR block, in IPv4 CIDR, enter 10.23.0.0/16, and then select **Create VPC**.
6. Review the properties of the VPC.

The Route 53 Resolver DNS Firewall error is to be expected.

1. In the VPC dashboard navigation pane, in Virtual private cloud, select **Subnets**, and then review the **IPv4 CIDR** ranges for the current subnets.
2. You should not see any subnets that begin with 10.23.
3. When you create a VPC manually, no subnets are created in the CIDR block you defined for the VPC. Since the CIDR block you created was 10.23.0.0/16, subnets for that VPC would need to start with 10.23. This means that there are no subnets defined for your VPC.

You will create subnets in the next task.

### **Create a public and a private subnet.**

1. On the Subnets page, select **Create subnet**.
2. On the Create subnet page, in the VPC pane, in In VPC ID, select **vpc-42446610**.
3. In Subnet settings, Subnet 1 of 1, in Subnet name, enter Public Subnet 42446610.
   * + 1. In Availability Zone, ensure that **No preference** is selected.
       2. In IPv4 subnet CIDR block, enter 10.23.0.0/24, and then select **Add new subnet**.
4. In Subnet 2 of 2, in Subnet name, enter Private Subnet 42446610.
   * + 1. In Availability Zone, ensure that **No preference** is selected.
       2. In IPv4 subnet CIDR block, enter 10.23.1.0/24, and then select **Create subnet**.

Subnets are assigned to one of the availability zones for the VPC's region. You can allow AWS to select the zone, or you can specify the zone assignment. When EC2 instances are built, they are connected to a subnet, which will then dictate which availability zone will be the home for the EC2 instance.

1. On the Subnets page, select the **Public Subnet 42446610** check box, select **Actions**, and then select **Edit subnet settings**.
2. On the Edit subnet settings page, in Auto-assign IP settings, select **Enable auto-assign public IPv4 address**, and then select **Save**.
3. On the Subnets page, review the properties of the new subnets.

The IPv4 CIDR ranges should begin with 10.23.

## Configure routing.

In this exercise, you will configure routing for a VPC. First, you will create an internet gateway, and then you will create a network access translation (NAT) gateway. Next, you will create route tables for the public and private subnets. Finally, you will associate the subnets to the route tables.

As in a physical network, you configure route tables and gateways that allow traffic to flow into and out of the subnets in the VPC.

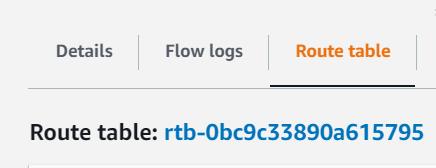
### **Review the default route.**

* 1. On the Subnets page, select the **Public Subnet 42446610** check box, and then in the Details pane, review the properties of the subnet.

You may need to scroll down to see the subnet properties.

* 1. In the Public Subnet 42446610 tile, select the **Route table** tab, locate the route table id.

The route table id will begin with the letters rtb-, followed by a random string of letters and numbers.



There is only one route listed in the route table. This route defines the address space for the VPC—it only has routes to the other subnets in the VPC.

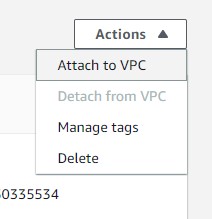
* 1. Clear the **Public Subnet 42446610** check box, select the **Private Subnet 42446610** check box, and then review the route table for the subnet.

The private subnet should use the same route table as the public subnet.

VPCs have a default route table that defines all internal subnets. The default route table does not have any entries for routes to the internet. All subnets in the VPC will use this route table unless they are modified to use a different route table.

### **Create an internet gateway.**

1. In the VPC dashboard navigation pane, in Virtual private cloud, select **Internet gateways**.
2. On the Internet gateways page, select **Create internet gateway**.
3. On the Create internet gateway page, in Internet gateway settings, in Name tag, enter igw-42446610, and then select **Create internet gateway**.
4. On the Actions menu, select **Attach to VPC**.



1. In the list of available VPCs, select **vpc-42446610**, and then select **Attach internet gateway**.

An [internet gateway](https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Internet_Gateway.html) provides direct access to and from the internet from the subnets to which it is aligned. Before you can associate an internet gateway with a subnet, you need to attach the gateway to the VPC that contains the subnet. A subnet that is linked to an internet gateway is called a public subnet. You use a route table to link a subnet to an internet gateway.

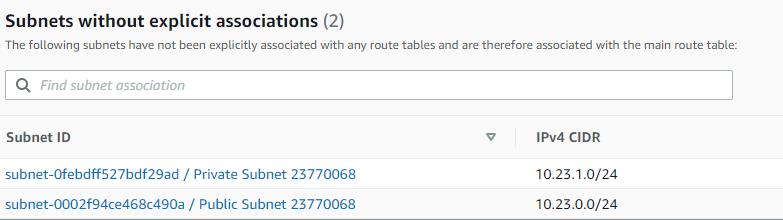
### **Create a NAT gateway.**

1. In the VPC dashboard navigation pane, in Virtual private cloud, select **NAT gateways**.
2. On the NAT gateways page, select **Create NAT gateway**.
3. On the Create NAT gateway page, in NAT gateway settings, in Name, enter NAT-gw-42446610.
4. In Subnet, select **Public Subnet 42446610**, and then select **Allocate Elastic IP**.
5. Select **Create NAT gateway**.

A [NAT gateway](https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat-gateway.html) provides indirect access to the internet from subnets that do not have internet gateways. These subnets are called private subnets. Because the NAT gateway itself needs direct access to the internet, it must be located on a subnet that will have an internet gateway. You use a route table to direct the traffic from the private subnets to the NAT gateway.

### **Update the route tables.**

1. In the VPC dashboard navigation pane, in Virtual private cloud, select **Route tables**.
2. On the Route tables page, select the **<RouteTableId>** check box.
3. In the Name column, select the edit icon.
4. In Edit Name, enter Private-rt-42446610, and then select **Save**.
5. In the Private-rt-42446610 pane, select the **Subnet associations** tab, review the subnet associations.

You should see that both of the subnets are currently inheriting this route table from the VPC.

1. On the Route tables page, ensure that **Private-rt-42446610** check box is selected, select the **Routes** tab, and then in the Routes tile, select **Edit routes**.
2. On the Edit routes page, select **Add route**, and then in Destination, enter 0.0.0.0/0.
3. In Target, select **NAT Gateway**, select **NAT-gw-42446610**, and then select **Save changes**.
4. On the breadcrumb menu, select **Route tables** to return to the route tables page.
5. On the Route tables page, select **Create route table**.
6. On the Create route table page, in Route table settings, in Name, enter Public-rt-42446610, in VPC, select **vpc-42446610**, and then select **Create route table**.
7. On the Public-rt-42446610 page, on the Routes tab, in the Routes tile, select **Edit routes**.
8. On the Edit routes page, select **Add route**, and then in Destination, enter 0.0.0.0/0.
9. In Target, select **Internet Gateway**, select **igw-42446610**, select **Save changes**.
10. On the Public-rt-42446610 page, select the **Subnet associations** tab, and then in the Explicit subnet associations tile, select **Edit subnet associations**.\
11. On the Edite subnet associations page, select the **Public Subnet 42446610** check box, and then select **Save associations**.
12. On the breadcrumb menu, select **Route tables** to return to the Route tables page.
13. On the Route tables page, select **Private-rt-42446610** check box, select the **Subnet associations** tab, and then select **Edit subnet associations**.
14. On the Edit subnet associations page, select the **Private Subnet 42446610** check box, and then select **Save associations**.

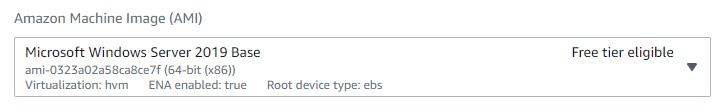
## Create an EC2 solution.

In this exercise, you will create an EC2 instance by using a custom Amazon Machine Image (AMI) and a launch template. First, you will create a Windows Server 2019 EC2 instance that will automatically install an Internet Information Service (IIS) web server, and then you will create a custom AMI from the instance. Next, you will create a security group that will allow HTTP access from the internet, and then you will create an EC2 launch template that will use the custom AMI and security group. Finally, you will create a new EC2 instance by using the launch template.

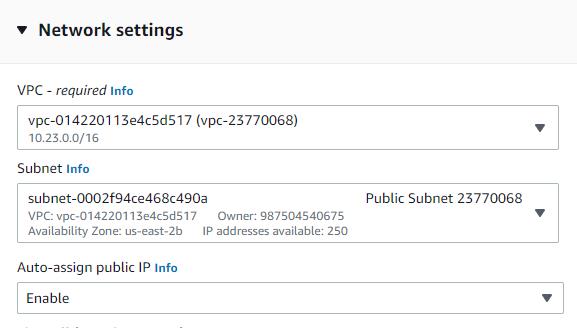
When you create an EC2 instance, you need to specify the AMI that your instance will use. AWS provides many standard images, but if you have a custom configuration that you need to use in your environment—for example, a web server—you can create an AMI that contains your configuration. You can create a launch template if you need a repeatable configuration to automate your deployments. The configuration that you define in a launch template can include a custom AMI, a specific instance type, a key pair, and a security group.

### **Create an EC2 instance.**

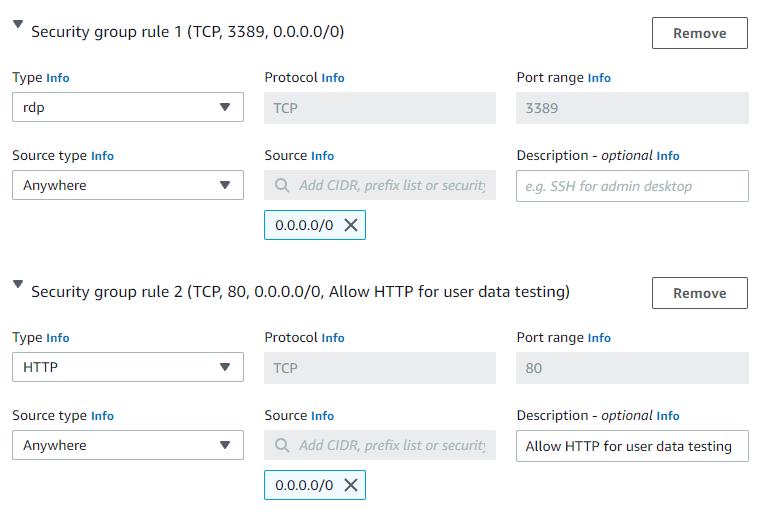
* + 1. On the Services menu, in Compute, select **EC2**.
    2. In the EC2 Dashboard navigation pane, in Instances, select **Instances**.
    3. On the Instances page, select **Launch instances** to start the Amazon EC2 Launch Instance wizard.
    4. On the Launch an instance page, in Name and tags, In Name, enter Win-starter-23770068.
    5. In Application and OS Images (Amazon Machine Image), in Quick Start, select **Windows**, and then in Amazon Machine Image (AMI), select the **Microsoft Windows Server 2019 Base** image.

Ensure that you use **Quick Start** and select the AMI that is eligible for the free tier and does NOT include the containers service.

* + 1. In Instance type, ensure that **t2.micro** is selected.
    2. In Key pair (login), select **Create new key pair**.
    3. In the Create key pair window, in Key pair name, enter Win-42446610-keypair, and then select **Create key pair**.
    4. In Network settings, select **Edit**, in VPC, select **vpc-42446610**, in Subnet, select **Public Subnet 42446610**, and then in Auto-assign Public IP, select **Enable**.



* + 1. In Security group name, enter Win-sg-42446610, and then select **Add security group rule**.
    2. In Security group rule 2, in Type, select **HTTP**, in Source type, select **Anywhere**, and then in Description, enter Allow HTTP for user data testing.



* + 1. Expand Advanced details, and then in Metadata version, select **V1 and V2 (token optional)**.
    2. In User data, enter the following script:

**UserData**

<powershell>

Install-WindowsFeature -name Web-Server -IncludeManagementTools

</powershell>

* + 1. Review the instance configuration settings, and then in the Summary, select **Launch instance**.
    2. Review the Launch Status page, and then select **View all instances**.
    3. Periodically refresh the page until the instance state value changes to **Running** and the Status Checks value changes to **2/2 checks passed**.

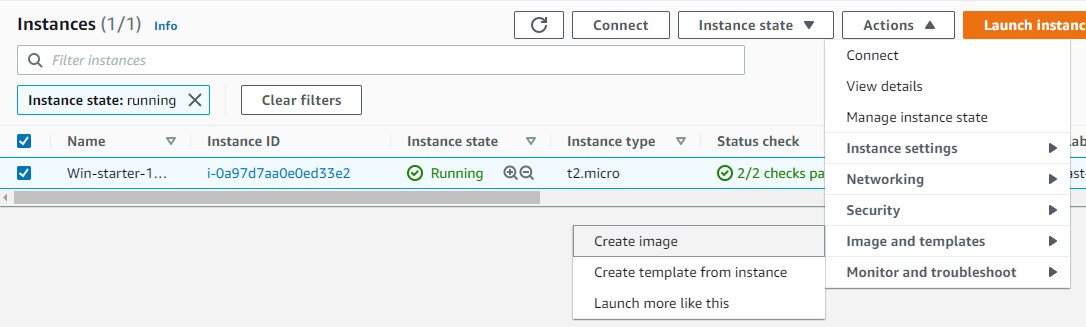
The completed instance stateIt may take approximately five minutes IIS to finish installing on the instance and for the status to change.

* + 1. On the Instances page, select **Win-Starter-42446610** check box, and then review the properties of the instance.
    2. On the Details tab, copy the **Public IPv4 address**.
    3. Open a new browser window, and then go to the IP address you copied.

The Internet Information Service starter page should load. If it does, you have successfully created an EC2 instance.

If the page fails to load, make sure that your browser is using HTTP not HTTPS. You created a security group rule that allows HTTP traffic from anywhere.

### **Create a custom AMI from an EC2 instance.**

1. On the Instances page, ensure that the **Win-starter-42446610** check box is selected.
2. Select the **Actions** menu, select **Image and templates**, and then select **Create image**.
3. On the Create image page, in Image name, enter Windows Web Server 2019, in Image description, enter Windows Server 2019 with IIS installed, and then select **Create image**.
4. In the EC2 Dashboard navigation pane, in Images, select **AMIs** to display the status of the new image.
5. Periodically refresh the page until the value in the Status column changes to **available**.

You can launch an EC2 instance by using just the AMI, but you would need to set all of the other configurations—instance type, networking, security groups, etc.—every time the instance is called. You can use a launch template to pre-define the configuration you need and simplify the build process.

Do not proceed until the status changes to available. The AMI cannot be used to launch an instance until it is available.

### **Create a security group.**

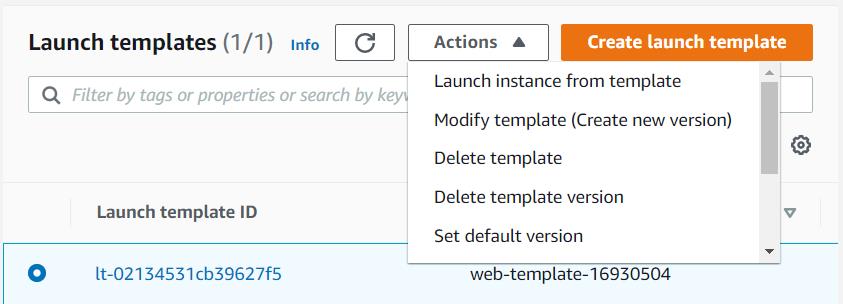
1. In the EC2 Dashboard navigation pane, in Network & Security, select **Security Groups**.
2. On the Security Groups page, select **Create security group**.
3. On the Create security group page, in Basic details, in Security group name, enter Web-sg-42446610, in Description, enter Allow public access to web servers in vpc-42446610, and then in VPC, select **vpc-42446610**.
4. In Inbound rules, select **Add rule** to add an inbound rule for the group.
5. In Type, select **HTTP**, in Source, select **Anywhere-IPv4**, in Description, enter Allow HTTP from anywhere, and then select **Create security group**.

You will use this security group in the launch template that you will create in the next task.

### **Create a custom launch template.**

1. In the EC2 Dashboard navigation pane, in Instances, select **Launch Templates**.
2. Select **Create launch template**.
3. On the Create launch template page, in Launch template name and description, in Launch template name, enter web-template-42446610, and then in Template version description, enter Windows 2019 web server running on a t2.micro instance.
4. In Launch template contents, in Application and OS Images (Amazon Machine Images), select **My AMIs**, and then in Amazon Machine Image (AMI), select Windows Web Server 2019.
5. In Instance type, select **t2.micro**.
6. In Key pair (login), in Key pair name, select **Win-42446610-keypair**.
7. In Network settings, in Security groups, select **Web-sg-42446610**.
8. In Launch template contents, expand **Advanced details**, and then in Metadata version, select **V1 and V2 (token optional)**.
9. Select **Create launch template**.
10. Review the results page, verify that the launch template was successfully created, and then select **View launch templates**.

### **Deploy an EC2 instance by using a launch template.**

1. On the Launch Templates page, select **web-template-42446610**, select the **Actions** menu, and then select **Launch instance from template**.
2. Review the settings that will be used to launch the instance. You should see the configurations identified in the launch template.
3. On the Launch instance from template page, in Resource tags, select **Add new tag**.
4. In Key, enter Name, in Value, enter Win-Web-42446610, and then in the Summary tile, select **Launch instance**.
5. Verify that the launch was successfully initiated, and then select **View launch templates**.
6. In the EC2 Dashboard navigation pane, in Instances, select **Instances**.
7. On the Instances page, select the **Win-Web-42446610** check box, and then review the properties of the instance.

If you do not see the Win-Web-42446610 image, refresh the page.

1. On the Details tab, copy the **Public IPv4 address**.
2. Open a new browser window, and then go to the IP address you just copied.

The Internet Information Service starter page should load. If it does, you have successfully created an instance based on the launch template.

If the page fails to load, make sure that your browser is using HTTP not HTTPS. You created a security group rule that allows HTTP traffic from anywhere.

As before, you may need to wait for the value of the Status Check column to change to 2/2 checks passed before the webpage will appear.