

Lab 13

WORKING WITH AMAZON EC2 AUTO SCALING GROUPS

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.

The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, 'Services' dropdown, and user information 'Antonio Ang', 'Oregon', and 'Support'. Below the navigation bar, the main content area is titled 'Amazon Web Services' and is divided into several columns of service categories. On the right side, there's a section titled 'Additional Resources' with links to 'Getting Started', 'AWS Console Mobile App', 'AWS Marketplace', 'Service Health', and 'Set Start Page'. The 'Service Health' section shows a status of 'All services operating normally' as of 'Nov 20 2014 12:57:00 GMT-0800'. The 'Set Start Page' section has a dropdown menu set to 'Console Home'.

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 and the password



Account:

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.

Antonio Ang ▾ Oregon ▾ Support ▾

- US East (N. Virginia)
- | **US West (Oregon)**
- US West (N. California)
- EU (Ireland)
- EU (Frankfurt)
- Asia Pacific (Singapore)
- Asia Pacific (Tokyo)
- Asia Pacific (Sydney)
- South America (São Paulo)

STEP 2: Auto Scaling Overview

Before going to the AWS console and creating an Auto Scaling Group, let's take a quick look at the components of an Auto Scaling Group. AWS has done a great job defining them so we'll use the official definition:

Groups

Your EC2 instances are organized into groups so that they can be treated as a logical unit for the purposes of scaling and management. When you create a group, you can specify its minimum, maximum, and, desired number of EC2 instances. For more information, see Auto Scaling Groups.

Launch configurations

Your group uses a launch configuration as a template for its EC2 instances. When you create a launch configuration, you can specify information such as the AMI ID, instance type, key pair, security groups, and block device mapping for your instances. For more information, see Launch Configurations.

You can read the full documentation here

<http://docs.aws.amazon.com/autoscaling/latest/userguide/WhatIsAutoScaling.html>

In this lab, we will learn to create an Auto Scaling Group with these components and place it behind an Elastic Load Balancing (ELB). Don't worry if you don't fully understand all the components yet. We will talk in greater detail about each of the components as we create them.

At the end of this lab we'll have an Auto Scaling Group with some web server instances behind an ELB. Although this lab focuses on Auto Scaling, it is important to mention that to have an Auto Scaling Group behind an ELB, it is necessary to create the ELB first. In the next step, we will begin exploring elements in the AWS console by creating an ELB.

STEP 3: Create a load balancer using ELB

Elastic Load Balancing (ELB) automatically distributes incoming application traffic across multiple Amazon EC2 instances. It enables you to achieve greater fault tolerance in your applications and seamlessly provides the correct amount of load balancing capacity needed in response to incoming application traffic.

Elastic Load Balancing detects unhealthy instances within a pool and automatically reroutes traffic to healthy instances until the unhealthy instances have been restored to health. Customers can enable Elastic Load Balancing within a single Availability Zone or across multiple zones for greater consistent application performance.

You can create your first ELB by taking the following steps:

From the EC2 dashboard, click the **Load Balancers** link in the Load Balancing group. The list of all already-created Load Balancers appears--this list will most likely be empty.

The screenshot displays the AWS Management Console interface. At the top, the header includes the AWS logo, navigation tabs for 'AWS', 'Services', and 'Tools', and user information: 'student @ 3752-6316-1608', 'Oregon', and 'Support'. The left-hand navigation pane lists various services. Under the 'LOAD BALANCING' category, the 'Load Balancers' link is highlighted with a blue circle. The main content area features a 'Create Load Balancer' button and an 'Actions' dropdown menu. Below these, a search bar indicates 'None found'. A message states: 'You do not have any load balancers in this region. To learn about Elastic Load Balancing, see our FAQ and Getting Started Guide. Click "Create Load Balancer" to create a load balancer that distributes traffic across your instances.' At the bottom of the main area, there is a section titled 'Select a Load Balancer' with three placeholder icons. The footer contains a 'Feedback' button, a language selector set to 'English', and links to 'Privacy Policy' and 'Terms of Use'.

Click the blue **Create Load Balancer** button.

On the **Define Load Balancer** step, type a load balancer name (e.g., "web") and select **Enable advanced VPC configuration**

Step 1: Define Load Balancer

This wizard will walk you through setting up a new load balancer. Begin by giving your new load balancer a unique name so that you can identify it from other load balancers you might create. You will also need to configure ports and protocols for your load balancer. Traffic from your clients can be routed from any load balancer port to any port on your EC2 instances. By default, we've configured your load balancer with a standard web server on port 80.

Load Balancer name:

Create LB Inside:

Create an internal load balancer: ☒ (what's this?)

Enable advanced VPC configuration: ☒

Listener Configuration:

Load Balancer Protocol	Load Balancer Port	Instance Protocol	Instance Port
HTTP	80	HTTP	80

Select Subnets

You will need to select a Subnet for each Availability Zone where you wish traffic to be routed by your load balancer. If you have instances in only one Availability Zone, please select at least two Subnets in different Availability Zones to provide higher availability for your load balancer.

VPC vpc-6d1f5f08 (172.31.0.0/16)

Please select at least two Subnets in different Availability Zones to provide higher availability for your load balancer.

Actions	Availability Zone	Subnet ID	Subnet CIDR	Name
<input checked="" type="checkbox"/>	us-west-2a	subnet-1eaafc7b	172.31.16.0/20	
<input checked="" type="checkbox"/>	us-west-2b	subnet-faadd98d	172.31.32.0/20	
<input type="checkbox"/>	us-west-2c	subnet-73149a2a	172.31.0.0/20	

Selected subnets

Actions	Availability Zone	Subnet ID	Subnet CIDR	Name
---------	-------------------	-----------	-------------	------

Cancel **Next: Assign Security Groups**

Select two subnets, one from the *us-west-2a* Availability Zone and one from the *us-west-2b* Availability Zone. Then click the **Next: Assign Security Groups** button.

AWS

Services

EC2

student 3752-6316-1608 Oregon Support

1. Define Load Balancer

2. Assign Security Groups

3. Configure Security Settings

4. Configure Health Check

5. Add EC2 Instances

Step 1: Define Load Balancer

Basic Configuration

This wizard will walk you through setting up a new load balancer. Begin by giving your new load balancer a unique name so that you can identify it from other load balancers you might create. You will also need to configure ports and protocols for your load balancer. Traffic from your clients can be routed from any load balancer port to any port on your EC2 instances. By default, we've configured your load balancer with a standard web server on port 80.

Load Balancer name:

web

Create LB inside:

My Default VPC (172.31.0.0/16)

Create an internal load balancer:

☒ (what's this?)

Enable advanced VPC configuration:

☒

Listener Configuration:

Load Balancer Protocol	Load Balancer Port	Instance Protocol	Instance Port
HTTP	80	HTTP	80

Add

Select Subnets

You will need to select a Subnet for each Availability Zone where you wish traffic to be routed by your load balancer. If you have instances in only one Availability Zone, please select at least two Subnets in different Availability Zones to provide higher availability for your load balancer.

VPC vpc-6d1f5f08 (172.31.0.0/16)

Available subnets

Actions	Availability Zone	Subnet ID	Subnet CIDR	Name
+	us-west-2c	subnet-73149a2a	172.31.0.0/20	

Selected subnets

Actions	Availability Zone	Subnet ID	Subnet CIDR	Name
-	us-west-2a	subnet-1eaafc7b	172.31.16.0/20	
-	us-west-2b	subnet-faadd98d	172.31.32.0/20	

Cancel

Next: Assign Security Groups

Feedback

English

Privacy Policy

Terms of Use

In the **Assign Security Groups** section, select **Create a new security group**, type a Security group name (e.g., "elb-webserver") and a description. Create a single firewall rule of type *HTTP*, protocol *TCP*, port range *80*, and source *Anywhere*. Click **Next: Configure Security Settings**.

AWS

Services

student

3752-6316-1508

Oregon

Support

1. Define Load Balancer

2. Assign Security Groups

3. Configure Security Settings

4. Configure Health Check

5. Add EC2 Instances

Step 2: Assign Security Groups

You have selected the option of having your Elastic Load Balancer inside of a VPC, which allows you to assign security groups to your load balancer. Please select the security groups to assign to this load balancer. This can be changed at any time.

Assign a security group:

☒ Create a new security group

☐ Select an existing security group

Security group name:

Description:

Type	Protocol	Port Range	Source
<input type="text" value="HTTP"/>	<input type="text" value="TCP"/>	<input type="text" value="80"/>	<input type="text" value="Anywhere 0.0.0.0/0"/>

Cancel

Previous

Next: Configure Security Settings

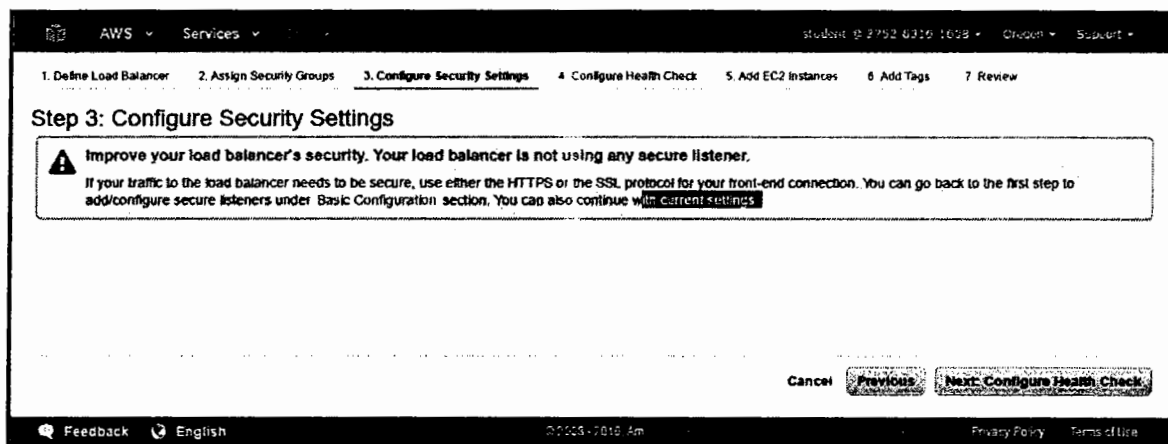
Feedback

English

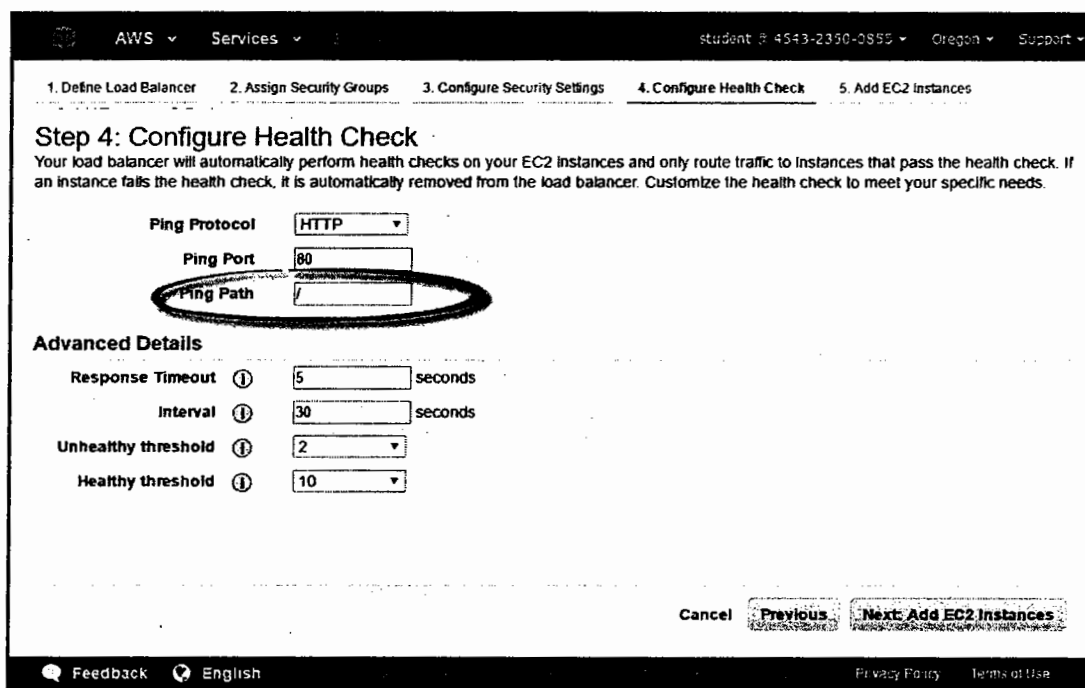
Privacy Policy

Terms of Use

Ignore the warning in the **Configure Security Settings** section. We are only serving the HTTP protocol in this exercise, so these settings are not required. Click **Next: Configure Health Check**.



In the **Configure Health Check** section, replace the default value of **Ping Path** with a single forward slash ("/") and click **Next: Add EC2 Instances**.



In the **Add EC2 Instances** section, you should see a "No instances available" message. This is because we have yet created and launched our Auto Scaling Group. Click **Next: Add Tags to continue**.

AWS Services student @ 4543-2350-0855 Oregon Support

1. Define Load Balancer 2. Assign Security Groups 3. Configure Security Settings 4. Configure Health Check 5. Add EC2 Instances

Step 5: Add EC2 Instances

The table below lists all your running EC2 Instances. Check the boxes in the Select column to add those Instances to this load balancer.

VPC vpc-2565d141 (172.31.0.0/16)

Instance	Name	State	Security groups	Zone	Subnet ID	Subnet CIDR
No instances available.						

Availability Zone Distribution

☒ Enable Cross-Zone Load Balancing ⓘ

☒ Enable Connection Draining ⓘ seconds

Cancel Previous Next: Add Tags

Feedback English

You may leave the fields blank in the **Add Tags** section. Click the **Review and Create** button to continue.

AWS Services student @ 4543-2350-0855 Oregon Support

1. Define Load Balancer 2. Assign Security Groups 3. Configure Security Settings 4. Configure Health Check 5. Add EC2 Instances

Step 6: Add Tags

Apply tags to your resources to help organize and identify them.

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. Learn more about tagging your Amazon EC2 resources.

Key	Value
<input type="text"/>	<input type="text"/>

Create Tag

Cancel Previous Review and Create

Feedback English

Review your settings, then click **Create** when ready.

AWS
Services
student @ 4543-2350-0855
Oregon
Support

1. Define Load Balancer
2. Assign Security Groups
3. Configure Security Settings
4. Configure Health Check
5. Add EC2 Instances

Step 7: Review

Please review the load balancer details before continuing

▼ Define Load Balancer

Edit load balancer definition

Load Balancer name: web
Scheme: internet-facing
Port Configuration: 80 (HTTP) forwarding to 80 (HTTP)

▼ Configure Health Check

Edit health check

Ping Target: HTTP:80/
Timeout: 5 seconds
Interval: 30 seconds
Unhealthy threshold: 2
Healthy threshold: 10

▼ Add EC2 Instances

Edit instances

Cross-Zone Load Balancing: Enabled
Connection Draining: Enabled, 300 seconds
Instances:

▼ VPC Information

Edit subnets

VPC: vpc-2565d141
Subnets: subnet-b4d603c2, subnet-e26bdb86

▼ Security groups

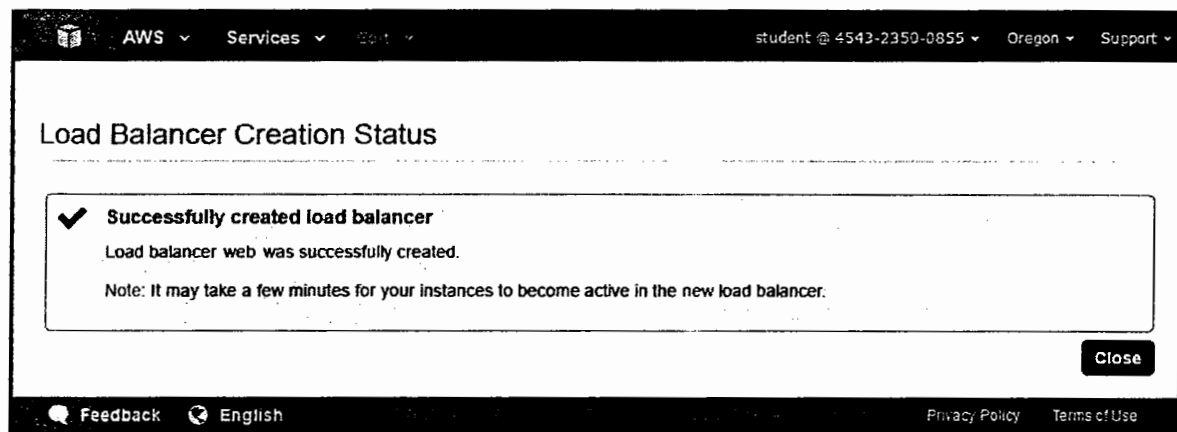
Edit security groups

Security groups: elb-webserver

Cancel
Previous
Create

Feedback
English
Privacy Policy
Terms of Use

Wait for the Load Balancer Creation Status to populate with the message, "Successfully created load balancer." Click **Close**.



STEP 4: Create a Launch Configuration

A **Launch Configuration** is a template that the Auto Scaling group uses to launch Amazon EC2 instances. If you've launched an individual EC2 instance before, you've already walked through the process of defining compute characteristics such as the instance type, security groups, and configuration scripts. A launch configuration allows you to define these same characteristics, which are then applied to any instances launched in the Auto Scaling group.

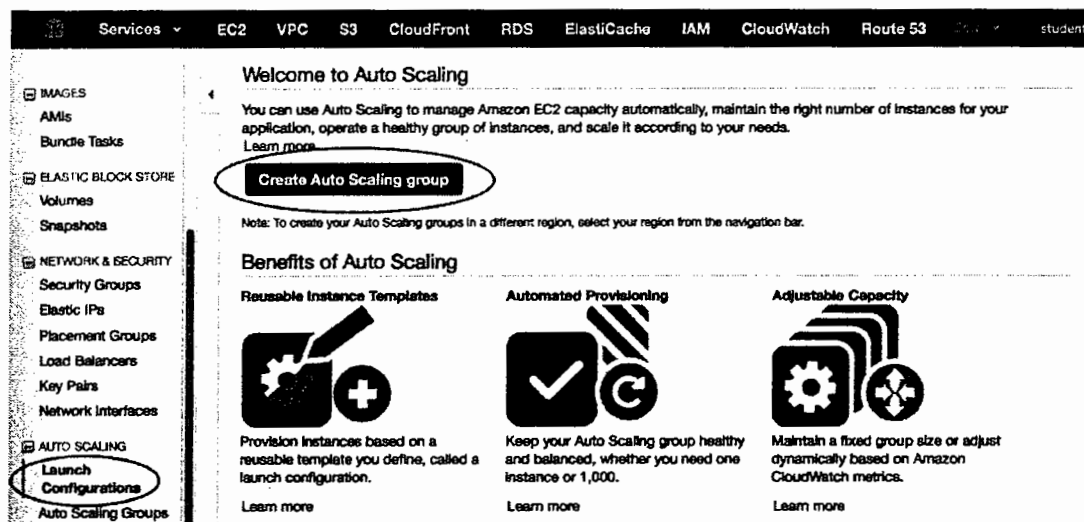
You create the launch configuration by including information such as the Amazon machine image ID to use for launching the EC2 instance, the instance type, key pairs, security groups, and block device mappings, among other configuration settings. When you create your Auto Scaling group, you must associate it with a launch configuration. You can attach only one launch configuration to an Auto Scaling group at a time and it cannot be modified.

Let's start creating our Auto Scaling Group by first defining a **Launch Configuration**.

Navigate to the EC2 service from the AWS dashboard:



Open the **Launch Configurations** page and click on the **Create Auto Scaling group** button.



This brings you to the Create Auto Scaling group wizard. Click on the **Create Launch configuration** button.

Create Auto Scaling Group

[Cancel and Exit](#)

To create an Auto Scaling group, you will first need to choose a template that your Auto Scaling group will use when it launches instances for you, called a launch configuration. Choose a launch configuration or create a new one, and then apply it to your group.

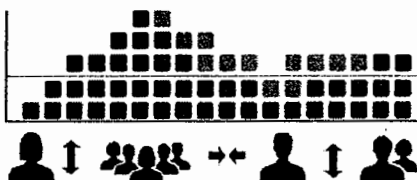
Later, if you want to use a different template, you can create another launch configuration and apply it to this group, even if you already have instances running in it. Using this method, you can update the software that your group uses when it launches new instances.



Step 1: Create launch configuration

First, define a template that your Auto Scaling group will use to launch instances.

You can change your group's launch configuration at any time. [Learn more](#)



Step 2: Create Auto Scaling group

Next, give your group a name and specify how many instances you want to run in it.

Your group will maintain this number of instances, and replace any that become unhealthy or impaired.

You can optionally configure your group to adjust in capacity according to demand, in response to Amazon CloudWatch metrics. [Learn more](#)

[Cancel](#)

Create launch configuration

From there the AWS Management Console guides you through each required step and displays a graphical interface that is similar to the Launch Instance Wizard.

The first step is the AMI selection. You have to select the AMI that will be used by all the EC2 instances of the Auto Scaling group. The Vepsun DevOps team created a specific AMI for this laboratory. You can find it among the Community AMIs by searching for the word "vepsun" in the AMI search box.

Select the **"vepsun-labs-webserver-basic"** AMI and click Select.

Create Launch Configuration

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.

Quick Start

My AMIs

AWS Marketplace

Community AMIs

Operating system

- Amazon Linux
- Cent OS
- Debian
- Fedora
- Genioo
- OpenSUSE
- Other Linux
- Red Hat
- SUSE Linux
- Ubuntu
- Windows

Architecture

- 32-bit
- 64-bit

Root device type

- EBS
- Instance store

cloudacademy-labs-webserver-basic - ami-d1792ce1

Ubuntu image with nginx, php, git, awscli

Root device type: ebs Virtualization type: hvm

Select

cloudacademy-labs-openswan-20160201 - ami-e021c080

Openswan AMI used by Cloud Academy Labs

Root device type: ebs Virtualization type: hvm

Select

Feedback English

Privacy Policy Terms of Use

The next step is choosing the instance type. Select the t2.micro type and click on the **Next: Configure details** button.

Create Launch Configuration

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 (GiB)	EBS-Optimized Available	Network Performance
<input checked="" type="radio"/>	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate
<input type="radio"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate
<input type="radio"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate
<input type="radio"/>	General purpose	m3.medium	1	3.75	1 x 4 (SSD)	-	Moderate
<input type="radio"/>	General purpose	m3.large	2	7.5	1 x 32 (SSD)	-	Moderate
<input type="radio"/>	General purpose	m3.xlarge	4	15	2 x 40 (SSD)	Yes	High
<input type="radio"/>	General purpose	m3.2xlarge	8	30	2 x 80 (SSD)	Yes	High

Cancel Previous **Next: Configure details**

The **Configure details** step asks you to name your launch configuration and asks if you want to enable the CloudWatch detailed monitoring for your future instances. Type a friendly name (e.g., "webserver-cluster") for the Launch Configuration and Enable detailed monitoring.

Click on **Next: Add Storage** after you have filled in the required fields.

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

Create Launch Configuration

Name ①

Purchasing option ① ☐ Request Spot Instances

IAM role ①

Monitoring ① ☒ Enable CloudWatch detailed monitoring
[Learn more](#)

► Advanced Details

Later, if you want to use a different launch configuration, you can create a new one and apply it to any Auto Scaling group. Existing launch configurations cannot be edited.

Cancel Previous Skip to review Next: Add Storage

The **Add Storage** step allows you to add or increment the size of any EBS volume linked to each EC2 instance that will be started by the Auto Scaling group.

In order to complete this laboratory exercise, leave the defaults and do not add any EBS volumes. Then click on **Next: Configure Security Group**.

N.B.: You should use big EBS volumes only if your software requires storage space to process the application data. If you need to store raw or processed data, you should use Amazon S3, Redshift, DynamoDB or another storage/database service provided by Amazon.

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

Create Launch Configuration

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. https://docs.aws.amazon.com/en_us/console/ec2/launchinstance/storage about storage options in Amazon EC2.

Type ①	Device ①	Snapshot ①	Size (GB) ①	Volume Type ①	IOPS ①	Delete on Termination ①
Root	/dev/sda1	snap-a1fc262d	8	General Purpose (SSD) ③	24 / 3000	<input checked="" type="checkbox"/>

[Add New Volume](#)

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

Cancel Previous Skip to review Next: Configure Security Group

Create a new Security Group for your Auto Scaling Group. Choose a name (e.g., Webserver-cluster) and description, and add the required rules to allow inbound SSH and HTTP.

The default Amazon VPC subnet range is **172.31.0.0/16**. You can use it to allow the HTTP traffic, so the Elastic Load Balancing instance will be able route the HTTP requests to the instances of the Auto Scaling group.

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

Create Launch Configuration

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
SSH	TCP	22	My IP
HTTP	TCP	80	Custom IP 172.31.0.0/16

Click the blue **Review** button.

Once you have reviewed the details for accuracy, click the blue **Create launch configuration** button.

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

Create Launch Configuration

Review the details of your launch configuration. You can go back to edit the details of each section before you finish.

AMI Details Edit AMI

cloudacademy-labs-webserver-basic - ami-d1792ee1
 Ubuntu image with nginx, php, git, awscli
 Root device type: ebs Virtualization Type: hvm

Instance Type Edit instance type

Instance Type	ECUs	CPU	Memory	Storage	EBS-Optimized	Network
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

Launch configuration details Edit details

You will be presented with the *Select an existing key pair or create a new key pair* dialogue box. Notice that you will use this Key Pair to access all the instances that are going to be launched by the Auto Scaling service with this Launch Configuration, so secure your Key Pair.

Select **Create a new key pair** from the first drop-down menu and type in a Key pair name (e.g., webserver-cluster). Click the **Download Key Pair** button. Then click the **Create Launch Configuration** button in this dialogue box.

Select an existing key pair or create a new key pairX

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.

Create a new key pair▼

Key pair name

webserver-cluster

Download Key Pair

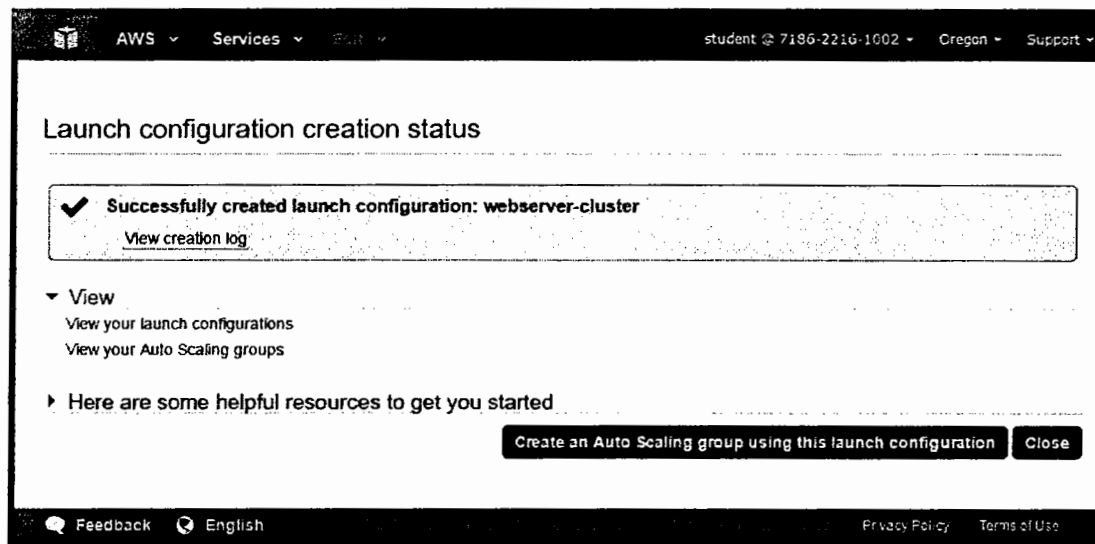
... You have to download the **private key file** (*.pem file) before you can continue. Store it in a **secure and accessible location**. You will not be able to download the file again after it's created.

CancelCreate launch configuration

Wait for the *Launch configuration creation status* to report, "Successfully created launch configuration." Congratulations! You created a new **Launch Configuration**.

We are going to continue creating our Auto Scaling Group in the next step. For now, just click **Close** in the AWS console and move on the next lab step.

340 | Page



STEP 5: Create an Auto Scaling Group

An Auto Scaling group is a representation of multiple Amazon EC2 instances that share similar characteristics and that are treated as a logical grouping for the purposes of instance scaling and management. For example, if a single application operates across multiple instances, you might want to increase or decrease the number of instances in that group to improve the performance of the application. You can use the Auto Scaling group to automatically scale the number of instances or maintain a fixed number of instances. You create Auto Scaling groups by defining the minimum, maximum, or desired number of running EC2 instances the group must have at any given point of time.

An Auto Scaling group starts by launching the minimum number (or the desired number, if specified) of EC2 instances and then increases or decreases the number of running EC2 instances automatically according to the conditions that you define. Auto Scaling also maintains the current instance levels by conducting periodic health checks on all the instances within the Auto Scaling group. If an EC2 instance within the Auto Scaling group becomes unhealthy, Auto Scaling terminates the unhealthy instance and launches a new one to replace the unhealthy instance. This automatic scaling and maintenance of the instance levels in an Auto Scaling group is the core value of the Auto Scaling service.

1. To create the Auto Scaling group, click on the **Auto Scaling Groups** link in the Auto Scaling menu group and then click the blue **Create Auto Scaling group** button.

Services ▾ EC2 VPC S3 CloudFront RDS ElastiCache IAM student @ 6725

Instances
Spot Requests
Reserved Instances

IMAGES
AMIs
Bundle Tasks

ELASTIC BLOCK STORE
Volumes
Snapshots

NETWORK & SECURITY
Security Groups
Elastic IPs
Placement Groups
Load Balancers
Key Pairs
Network Interfaces

AUTO SCALING
Launch Configurations
Auto Scaling Groups

Welcome to Auto Scaling

You can use Auto Scaling to manage Amazon EC2 capacity automatically, maintain the right number of instances for your application, operate a healthy group of instances, and scale it according to your needs. [Learn more](#)

You have the following Auto Scaling resources in the US West (Oregon) region


Auto Scaling Groups: 0 Launch Configuration: 1

[Create Auto Scaling group](#) [Create launch configuration](#)

Note: To create your Auto Scaling groups in a different region, select your region from the navigation bar.


Benefits of Auto Scaling

Reusable Instance Templates




Provision instances based on a reusable template you define, called a launch configuration. [Learn more](#)

Automated Provisioning



Keep your Auto Scaling group healthy and balanced, whether you need one instance or 1,000. [Learn more](#)

Adjustable Capacity



Maintain a fixed group size or adjust dynamically based on Amazon CloudWatch metrics. [Learn more](#)

2. Select **Create an Auto Scaling group from an existing launch configuration**, select the previously created launch configuration and click Next Step.

AWS ▾ Services ▾ student @ 4683-2055-0067 ▾ Oregon ▾ Support ▾

Create Auto Scaling Group

[Cancel and Exit](#)

To create an Auto Scaling group, you will first need to choose a template that your Auto Scaling group will use when it launches instances for you, called a launch configuration. Choose a launch configuration or create a new one, and then apply it to your group.

Later, if you want to use a different template, you can create another launch configuration and apply it to this group, even if you already have instances running in it. Using this method, you can update the software that your group uses when it launches new instances.

☒ Create a new launch configuration

☐ Create an Auto Scaling group from an existing launch configuration

Filter launch configurations

Name	AMI ID	Instance Type	Spot Price	Security Groups
webserver-ca	ami-d1792ee1	t2.micro		sg-a20117c5

[Cancel](#) [Next Step](#)

Feedback English Privacy Policy Terms of Use

3. In the "Configure Auto Scaling group details" step, you should use the following settings:

Group name: webserver-cluster

Group size: 1

Network: default

Subnet: Select two. The default network in *us-west-2a* and the default network in *us-west-2b*.

Open the Advanced Details, then set as follows:

Load Balancing: Check Receive traffic from Elastic Load Balancer(s). Select the "web" ELB you created

Health Check Type: ELB

Monitoring: Check *Enable CloudWatch detailed monitoring*

Once all fields are complete, click **Next: Configure Scaling Policies**.

The screenshot shows the AWS Management Console interface for creating an Auto Scaling Group. The top navigation bar includes the AWS logo, 'Services', and a user profile. The main content area is titled 'Create Auto Scaling Group' and has a progress bar with five steps: 1. Configure Auto Scaling group details (active), 2. Configure scaling policies, 3. Configure Notifications, 4. Configure Tags, and 5. Review. The 'Launch Configuration' section includes fields for 'Group name' (webserver-cluster), 'Group size' (1 instance), 'Network' (vpc-9741f7f2), and 'Subnet' (two subnets in us-west-2a and us-west-2b). The 'Advanced Details' section is expanded, showing 'Load Balancing' (checked), 'Health Check Type' (ELB), 'Health Check Grace Period' (300 seconds), and 'Monitoring' (checked). The 'Next: Configure scaling policies' button is highlighted.

1. Configure Auto Scaling group details 2. Configure scaling policies 3. Configure Notifications 4. Configure Tags 5. Review

Create Auto Scaling Group Cancel and Exit

Launch Configuration

Group name: webserver-cluster

Group size: Start with 1 instances

Network: vpc-9741f7f2 (172.31.0.0/16) (default) Create new VPC

Subnet: subnet-72913d05 (172.31.16.0/20) Default in us-west-2a
subnet-c4b03aa1 (172.31.32.0/20) Default in us-west-2b Create new subnet

Each instance in this Auto Scaling group will be assigned a public IP address.

Advanced Details

Load Balancing ☒ Receive traffic from Elastic Load Balancer(s)
web

Health Check Type ☒ ELB ☐ EC2

Health Check Grace Period 300 seconds

Monitoring ☒ Enable CloudWatch detailed monitoring
Learn more

Cancel **Next: Configure scaling policies**

4. In this step, you must *Configure scaling policies*, which determine how and when your infrastructure will scale out and scale back.

Select the *Use scaling policies to adjust the capacity of this group* button. For this lab you should set your group to scale between **1** and **5** instances.

The screenshot shows the 'Create Auto Scaling Group' wizard in the AWS Management Console, specifically the '2. Configure scaling policies' step. The page has a top navigation bar with 'AWS', 'Services', and a user profile. Below the navigation bar are tabs for '1. Configure Auto Scaling group details', '2. Configure scaling policies', '3. Configure Notifications', '4. Configure Tags', and '5. Review'. The main heading is 'Create Auto Scaling Group'. There are two radio buttons: 'Keep this group at its initial size' (which is selected) and 'Use scaling policies to adjust the capacity of this group'. Below these, it says 'Scale between 1 and 5 instances. These will be the minimum and maximum size of your group.' The 'Increase Group Size' section is expanded, showing fields for 'Name' (Increase Group Size), 'Execute policy when' (No alarm selected), 'Take the action' (Add 0 instances), and 'Instances need' (300 seconds to warm up after each step). There is an 'Add new alarm' button. The 'Decrease Group Size' section is also visible but not expanded. At the bottom, there are 'Cancel', 'Previous', 'Review', and 'Next: Configure Notifications' buttons. The footer includes 'Feedback', 'English', 'Privacy Policy', and 'Terms of Use'.

5. The Auto Scaling group policies allow you to automatically increase or decrease the group size based upon policies you define. In order to establish an Increase Group size or Decrease Group Size policy, you must create a CloudWatch Alarm and then define which action should be taken if it is triggered.

Click *Add new alarm* under the *Increase Group Size* section. A *Create Alarm* dialogue box will pop up.

If you want to receive a notification when the alarm is triggered, you need to set up an **SNS topic**. Check the *Send a notification to:* checkbox. Type in a name (e.g., "autoscaling-alarm-up") for the SNS topic and enter at least one email address in the recipients box.

Select a metric (e.g., Average, CPU Utilization) and a constraint (e.g., ≥ 80 percent). Select a count and an interval (e.g., For at least **1** consecutive period of **5 minutes**). Choose a name for the alarm, and then click **Create Alarm**.

Create Alarm

You can use CloudWatch alarms to be notified automatically whenever metric data reaches a level you define.
To edit an alarm, first choose whom to notify and then define when the notification should be sent.

☒ Send a notification to: [cancel](#)

With these recipients:

Whenever: Average of CPU Utilization

Is: \geq Percent

For at least: consecutive period(s) of 5 Minutes

Name of alarm:

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

CPU Utilization Percent

Time	CPU Utilization Percent
5/11 14:00	~75
5/11 16:00	~10
5/11 18:00	~10

webserver-cluster

Create another alarm with whatever settings you choose for the Decrease Group Size.
Click **Next: Configure Notifications**.

6. *Configure Notifications* will notify you whenever an Auto Scaling Group instance is launched or terminated -- with or without success.
7. Click **Add notification**. You can use one of the same SNS topics previously created for the CloudWatch alarms. When you're done, click the blue **Review** button.

1. Configure Auto Scaling group details 2. Configure scaling policies 3. Configure Notifications 4. Configure Tags 5. Review

Create Auto Scaling Group

Configure your Auto Scaling group to send notifications to a specified endpoint, such as an email address, whenever a specified event takes place, including: successful launch of an instance, failed instance launch, instance termination, and failed instance termination.

If you created a new topic, check your email for a confirmation message and click the included link to confirm your subscription. Notifications can only be sent to confirmed addresses.

Send a notification to: [create topic](#)

Whenever instances:

- ☒ launch
- ☒ terminate
- ☒ fail to launch
- ☒ fail to terminate

[Add notification](#)

[Cancel](#) [Previous](#) [Review](#) [Next: Configure Tags](#)

8. The **Review** tab allows you to review all the selected options. When you are satisfied, start the creation of your cluster by clicking on **Create Auto Scaling group**.

1. Configure Auto Scaling group details 2. Configure scaling policies 3. Configure Notifications 4. Configure Tags 5. Review

Create Auto Scaling Group

Please review your Auto Scaling group details. You can go back to edit changes for each section. Click **Create Auto Scaling group** to complete the creation of an Auto Scaling group.

▼ Auto Scaling Group Details Edit details

Group name	webservers-cluster
Group size	1
Minimum Group Size	1
Maximum Group Size	5
Subnet(s)	subnet-0e95216b, subnet-8dde14fa
Load Balancers	web
Health Check Type	ELB
Health Check Grace Period	300
Detailed Monitoring	Yes

Cancel Previous Create Auto Scaling group

9. In a few minutes your cluster will be deployed and your EC2 instances will be ready to.

Filter: Q. Filter Auto Scaling groups... X 1 to 1 of 1 Auto Scaling Groups

Name	Launch Configuration	Instances	Desired	Min	Max	Availability Zone(s)	Default Cooldown	Health Check Type
webservers-cluster	webservers-cluster	1	1	1	5	us-west-2b, us-west-2a	300	300

Auto Scaling Group: webservers-cluster Edit

Details Scaling History Scaling Policies Instances Notifications Tags

Launch Configuration	webservers-cluster
Load Balancers	web
Desired	1
Min	1
Max	5
Health Check Type	ELB
Health Check Grace Period	300
Termination Policies	Default
Availability Zone(s)	us-west-2b, us-west-2a
Subnet(s)	subnet-0e95216b, subnet-8dde14fa
Default Cooldown	300
Placement Group	
Suspended Processes	
Enabled Metrics	GroupMaxSize, GroupTerminatingInstances, GroupMinSize, GroupInServiceInstances, GroupDesiredCapacity, GroupPendingInstances, GroupTotalInstances

Creation Time Tue Dec 02 20:48:01 GMT-0800 2014

10. By opening the **Load Balancers** section, selecting your previously created ELB, and then opening the **Instances** tab, you can see the new Auto Scaling instance(s) automatically added to the ELB configuration.

Create Load Balancer

Actions

Filter: Q Search Load Balancers

< 1 to 1 of 1 >

Load Balancer Name	DNS Name	Port Configuration	Availability Zones	Instance Count	Health Check
web	web-1306826351.us-west-2...	80 (HTTP) forwarding to 80 (...)	us-west-2c, us-west-2b...	1 Instance	HTTP:80/index.html

Load balancer: web

Description

Instances

Health Check

Monitoring

Security

Listeners

Tags

Connection Draining: Enabled, 300 seconds (Edit)

Edit Instances

Instance ID	Name	Availability Zone	Status	Actions
i-e0dd50ea		us-west-2b	InService ①	Remove from Load Balancer

Edit Availability Zones

Availability Zone	Subnet ID	Subnet CIDR	Instance Count	Healthy?	Actions
us-west-2c	subnet-4bd03b12	172.31.0.0/20	0	No (Availability Zone contains no healthy instances)	Remove from Load Balancer
us-west-2b	subnet-0e95218b	172.31.32.0/20	1	Yes	Remove from Load Balancer
us-west-2a	subnet-8dde148a	172.31.16.0/20	0	No (Availability Zone contains no healthy instances)	Remove from Load Balancer

