

CLOUD CONCEPTS

(EC2 Launch Types)

- On Demand Instances: short workload, predictable pricing
- Reserved Instances: long workloads (≥ 1 year)
- Convertible Reserved Instances: long workloads with flexible instances
- Scheduled Reserved Instances: launch within time window you reserve
- Spot Instances: short workloads, for cheap, can lose instances
- Dedicated Instances: no other customers will share your hardware
- Dedicated Hosts: book an entire physical server, control instance placement

- Whenever server is launched the default instance is **On Demand Instance**. It is the most expensive as it is available immediately. **Reserved instances** are recommended for database usage whenever usage is 1 – 3 years. Types of reserved instances are **convertible** and **scheduled**. In **convertible** type of server can be changed e.g., T2 micro (1cpu 1gb) can be changed to any server with higher configuration if more traffic is to be entertained while in **scheduled** the time span or time slot is defined i.e., on Monday from 8am – 10am. **Spot instances** are used for bidding. To get the requested instance the bid must be higher if any bid is higher than yours even the instances in your use will be taken and granted to the highest bidder. In all the discussed instance types of portions in big machine is allocated and are said **dedicated Instances**. **Dedicated host** means you own entire physical server even if you are running only one application.

EC2 On Demand

- Pay for what you use (billing per second, after the first minute)
- Has the highest cost but no upfront payment
- No long term commitment
- Recommended for short-term and un-interrupted workloads, where you can't predict how the application will behave

EC2 Reserved Instances

- Up to 75% discount compared to On-demand
- Pay upfront for what you use with long term commitment
- Reservation period can be 1 or 3 years
- Reserve a specific instance type
- Recommended for steady state usage applications (think database)
- **Convertible Reserved Instance**
 - can change the EC2 instance type
 - Up to 54% discount
- **Scheduled Reserved Instances**
 - launch within time window you reserve
 - When you require a fraction of day / week / month

EC2 Spot Instances

- Can get a discount of up to 90% compared to On-demand
 - You bid a price and get the instance as long as its under the price
 - Price varies based on offer and demand
 - Spot instances are reclaimed with a 2 minute notification warning when the spot price goes above your bid
 - Used for batch jobs, Big Data analysis, or workloads that are resilient to failures.
- Not great for critical jobs or databases

Todo: Take the overview of all the other instance types from on amazon but do not launch the server as all are paid.

Lab (Creatin AMI copy from one region to other and launching AMI copy in same region)

What's an AMI?

- As we saw, AWS comes with base images such as:
 - Ubuntu
 - Fedora
 - RedHat
 - Windows
 - Etc
- Amazon machine image is base image or base AMI provided by Amazon. We simply select it and launch it with all the default services and configurations. We can create your own AMI. In this existing SG and key pair should be used. AMIs can be built for Windows and Linux machines.
- AMI are built in specific regions and have specific IDs in specific regions. AMI created in one region has ID A and same AMI created in other region will have ID B.
- AMI ID is unique.
- You can also use or rent your own created AMI with others. By default, the AMI is private it must be made public and then accessed publicly.
- Basic monitoring is free in AMAZON.

Why use AMI

Pre installed Packages

Faster boot time

Machine have basic monitoring configured

Security is managed

Maintainence, control and updates of AMI

Installing app ahead of time (auto scaling)

Using cutom made AMI otimized for running apps

- AMIs are stored on S3 as S3 is durable, cheap, and resilient storage where most of the back-ups reside. As it is in S3 you will be charged for the space it occupies in S3. Although it is inexpensive but it is better to delete them when not required.
- By default, AMIs are private and locked for account or region. These can be made public and shared with others. They can also be placed on Amazon marketplace.

Using Public AMIs

- You can leverage AMIs from other people
- You can also pay for other people's AMI by the hour
 - These people have optimized the software
 - The machine is easy to run and configure
 - You basically rent "expertise" from the AMI creator
- AMI can be found and published on the Amazon Marketplace
- Warning:
 - Do not use an AMI you don't trust!
 - Some AMIs might come with malware or may not be secure for your enterprise

- Check the latest price on AWS marketplace.

TODO: Launch the Apache server using Linux AMI with the boot script given below in user data. Make sure HTTP and SSH (Port 80 and 22) are enabled. It is recommended to access the server when the server status is checked 2/2. Save the Public IP of server and open in browser.

```
#!/bin/bash

##### USE THIS FILE IF YOU LAUNCHED AMAZON LINUX 2 #####
#####

# get admin privileges
sudo su

# install httpd (Linux 2 version)
yum update -y
yum install -y httpd.x86_64
systemctl start httpd.service
systemctl enable httpd.service
echo "Hello World from $(hostname -f)" > /var/www/html/index.html
```

- To create the copy, go to actions and click create image. Set the image name and description and click on Create Image.

Launch Instance Connect Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
	i-0098a5312c58dbdcf	t2.micro	us-east-2a	terminated	2/2 checks ...	None	-
server	i-066db8e2b0297d7...	t2.micro	us-east-2c	stopped	2/2 checks ...	None	-
	i-071bcd59b22715516	t2.micro	us-east-2a	running	2/2 checks ...	None	ec2-3-138-114-219.us-east-2.compute.amazonaws.com

Instance: i-071bcd59b22715516 Public DNS (IPv4) ec2-3-138-114-219.us-east-2.compute.amazonaws.com

Launch Instance Connect Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
	i-0098a5312c58dbdcf	t2.micro	us-east-2a	terminated	2/2 checks ...	None	-
	i-066db8e2b0297d7...	t2.micro	us-east-2c	stopped	2/2 checks ...	None	-
	i-071bcd59b22715516	t2.micro	us-east-2a	running	2/2 checks ...	None	ec2-3-138-114-219.us-east-2.compute.amazonaws.com

Instance: i-071bcd59b22715516 (awais-2) Public DNS: ec2-3-138-114-219.us-east-2.compute.amazonaws.com

Description Status Checks Monitoring Tags

Instance ID i-071bcd59b22715516 Public DNS (IPv4) ec2-3-138-114-219.us-east-2.compute.amazonaws.com IPv4 Public ID 3 138 114 219

Create Image

Description of the image. This field is limited to 127 characters in length and cannot be modified.

Instance ID i-071bcd59b22715516

Image name

Image description

No reboot ☐

Instance Volumes

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/xvda	snap-05741358b44a33b45	8	General Purpose S	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

Add New Volume

Total size of EBS Volumes: 8 GiB

When you create an EBS image, an EBS snapshot will also be created for each of the above volumes.

Cancel Create image

Create Image

Instance ID

i-071bcd59b22715516

Image name

Image description

No reboot

☐

Instance Volumes

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/xvda	snap-05741358b44a33b45	8	General Purpose	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

Add New Volume

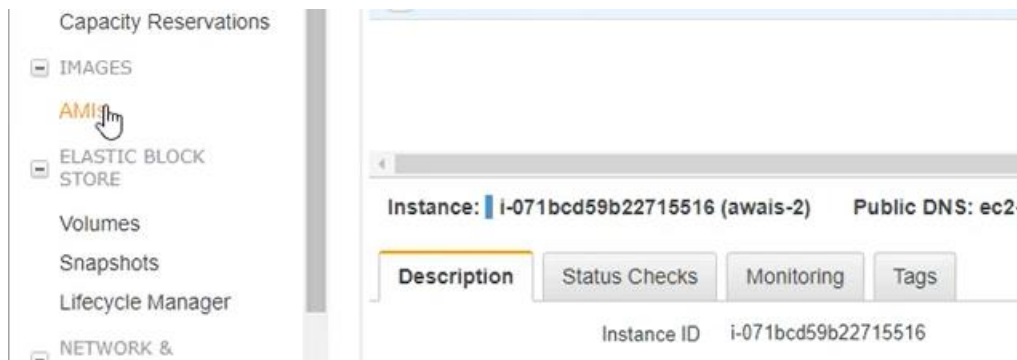
Total size of EBS Volumes: 8 GiB

When you create an EBS image, an EBS snapshot will also be created for each of the above volumes.

Cancel

Create Image

- Go to AMI on the left side and launch the server from there. Follow the steps and launch the server. Now the server is launched in the same region. In this case you will not be asked to select the AMI as the AMI is already copied. The new server will have Apache installed on it.
- You can make any changes if you want to. Same SG and Key pair are recommended to use.



Launch

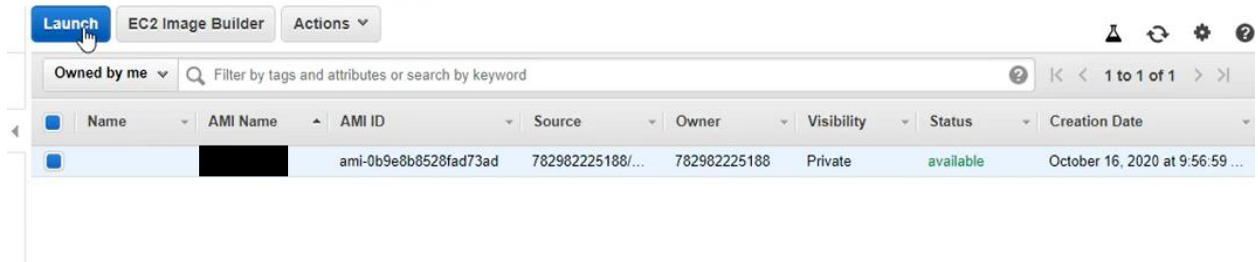
EC2 Image Builder

Actions

Owned by me

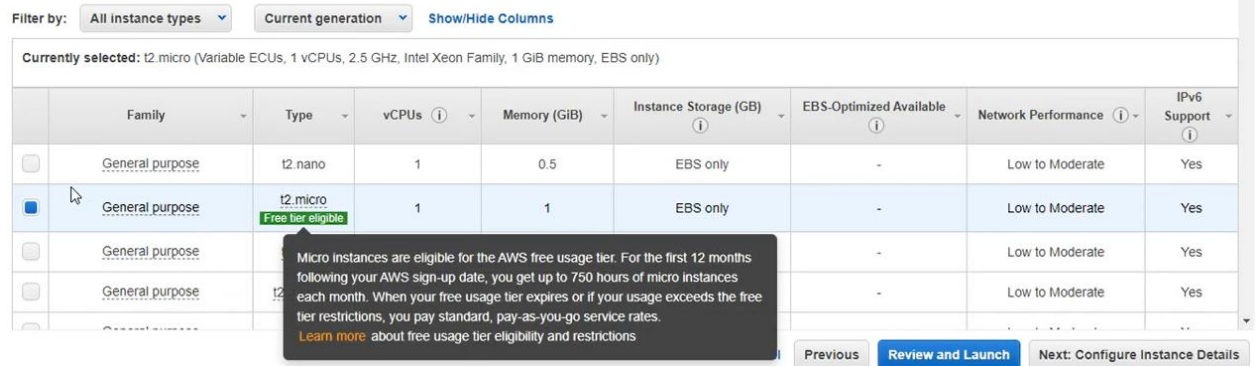
Filter by tags and attributes or search by keyword

Name	AMI Name	AMI ID	Source	Owner	Visibility	Status	Creation Date
		ami-0b9e8b8528fad73ad	782982225188/...	782982225188	Private	pending	October 16, 2020 at 9:56:59 ...

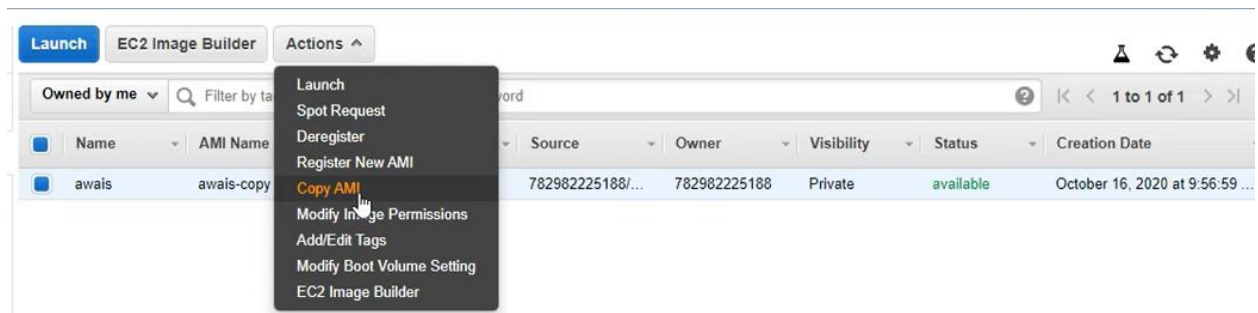


Step 2: Choose an Instance Type

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.



- To launch or copy the server to any other region select copy AMI and specify the region. Set the destination and click Copy AMI. Encryption option can be selected if required.



Lab (Using Elastic Load Balancer for scalability)

Scalability & High Availability

- Scalability means that an application / system can handle greater loads by adapting.
- There are two kinds of scalability:
 - Vertical Scalability
 - Horizontal Scalability (= elasticity)

- Vertical Scaling** means to increase the size of the instance i.e., shifting from t2.micro to t2.large. It is very common for non-distributed systems like databases. RDS is an example of elastic cache that scales vertically. This scaling has a bound due to hardware limitation.
- Horizontal scaling** means to increase the number of instances. It is very common for applications like web sites. EC2 is an example of horizontal scaling.

High Availability

- High Availability usually goes hand in hand with horizontal scaling
- High availability means running your application / system in at least 2 data centers (== Availability Zones)
- The goal of high availability is to survive a data center loss

- Vertical Scaling: Increase instance size (= scale up / down)

- From: t2.nano - 0.5G of RAM, 1 vCPU
- To: u-12tb1.metal - 12.3 TB of RAM, 448 vCPUs

- Horizontal Scaling: Increase number of instances (= scale out / in)

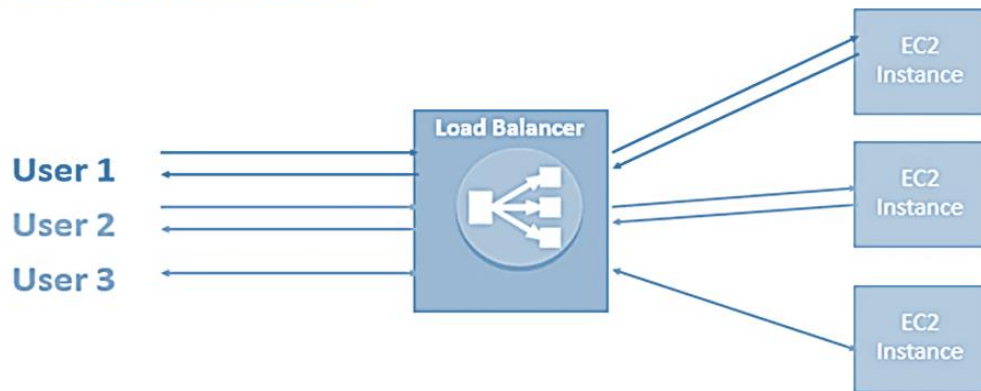
- Auto Scaling Group
- Load Balancer

High Availability: Run instances for the same application across multi AZ

- Auto Scaling Group multi AZ
- Load Balancer multi AZ

- **Load balancers** distribute the traffic amongst the instances and help in providing elasticity. Instances are attached with load balancer and the instances are then accessed through the load balancer via DNS providing single point of access to your application.

Load balancers are servers that forward internet traffic to multiple servers (EC2 Instances) downstream.



Why use
load
balancer?

Seamlessly handle failures of downstream instances

Do regular health checks of instances

Provide SSL termination (HTTPS) for your website

Enforce stickiness with cookies

High availability across zones

Separate public and private traffic

Why use an EC2 Load Balancer ?



- An ELB (EC2 Load Balancer) is a managed load balancer
 - AWS guarantees that it will be working
 - AWS takes care of upgrades, maintenance, high availability
 - AWS provides only a few configuration knobs
- It costs less to setup your own load balancer but it will be a lot more effort on your end.
- It is integrated with many AWS offerings / services
- It is integrated with many AWS offerings / services

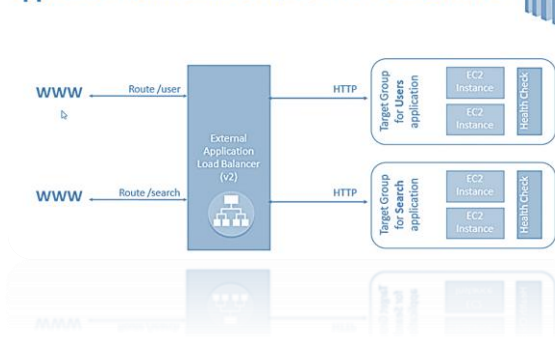
Application Load Balancer

- Used to route HTTP / HTTPs (layer 7) traffic
- Load balance across machine (target group)
- Load balance on same machine (containers)
- Best for micro services and container based applications e.g., Dockers and Amazon ECS
- Has port mapping feature to redirect to dynamic port

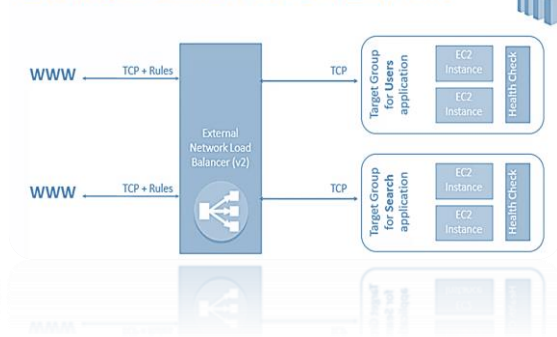
Network Load Balancer and Classic load balancer

- Used to route TCP (layer 5) traffic
- Handle millions of requests per second
- Support static and elastic IP
- Used for extreme performance
- Should not be the default load balancer

Application Load Balancer (v2) HTTP Based Traffic



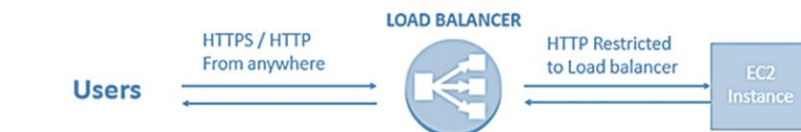
Network Load Balancer (v2) TCP Based Traffic



- Health Checks are crucial for Load Balancers
- They enable the load balancer to know if instances it forwards traffic to are available to reply to requests
- The health check is done on a port and a route (/health is common)
- If the response is not 200 (OK), then the instance is unhealthy



Load Balancer Security Groups



Load Balancer Security Group:

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	0.0.0.0/0	Allow HTTP from an...
HTTPS	TCP	443	0.0.0.0/0	Allow HTTPS from a...

Application Security Group: Allow traffic only from Load Balancer

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	sg-054b5ff5ea02f2b6e (load-b...	Allow Traffic only...

Application Load Balance

- Stickiness can be enabled at the target group level
 - Same request goes to the same instance
 - Stickiness is directly generated by the ALB (not the application)
- ALB support HTTP/HTTPS & Web sockets protocols



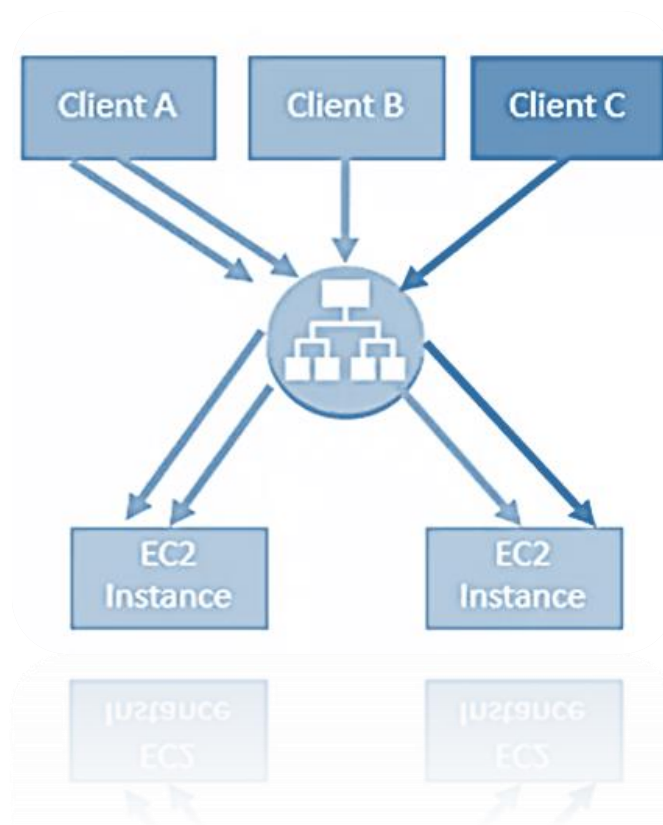
Load Balancer Stickiness

Load balancer can be made sticky

stickiness means that same client is always redirected to same instance

Cookie is used for stickiness and has expiration date

Enabling stickiness may bring imbalance to load balancer over backend EC2 instances



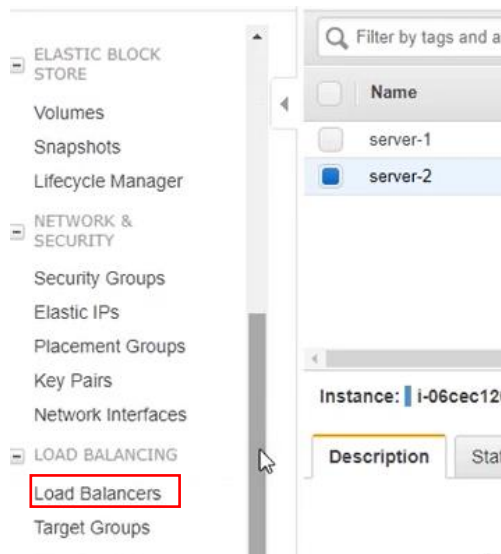
- Launch the Linux Server with boot strap code given below. You must launch two servers, just to make the difference change the echo statement of bash file for both servers. Your SG must have HTTP and SSH port open. Use public IPs of server to open in browser.

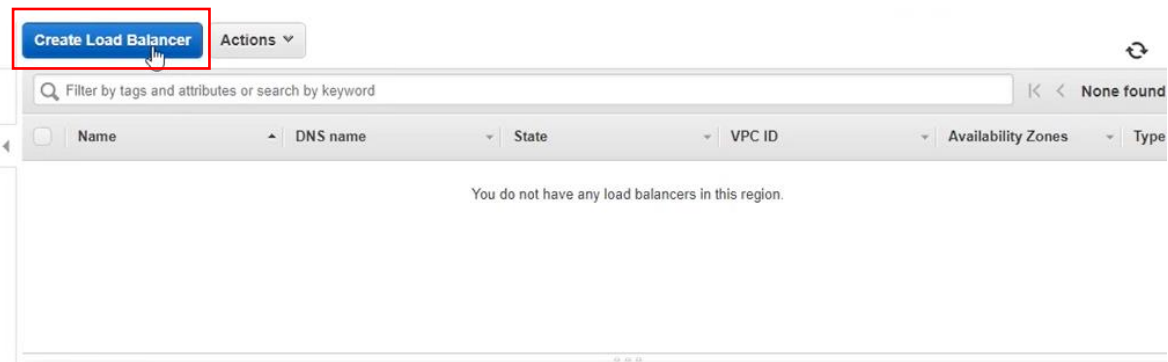
The image shows the AWS IAM console with two security group rules: SSH (TCP, Port 22) and HTTP (TCP, Port 80). Below this is a terminal window with the following bootstrap code:

```
File Edit Format View Help
#!/bin/bash
yum update -y
yum install httpd -y
service httpd start
chkconfig httpd on
cd /var/www/html
echo "<html><h1>This is WebServer-01</h1></html>" > index.html
```

Below the terminal are two browser screenshots. The left browser shows the output of the bootstrap script on a server with public IP 18.191.183.64, displaying "This is WebServer-01". The right browser shows the output of the bootstrap script on a server with public IP 18.217.126.173, displaying "This is WebServer-02".

- On left side scroll to Load balancer and click create load balancer and select Application load balancer.





Learn more about which load balancer is right for you

Application Load Balancer

HTTP
HTTPS

Create

Choose an Application Load Balancer when you need a flexible feature set for your web applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers.

[Learn more >](#)

Network Load Balancer

TCP
TLS
UDP

Create

Choose a Network Load Balancer when you need ultra-high performance, TLS offloading at scale, centralized certificate deployment, support for UDP, and static IP addresses for your application. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies.

[Learn more >](#)

Classic Load Balancer

PREVIOUS GENERATION
for HTTP, HTTPS, and TCP

Create

Choose a Classic Load Balancer when you have an existing application running in the EC2-Classical network.

[Learn more >](#)

- Configure the load balancer set the name and select internet facing to manage the traffic from public internet. To make high availability select all the regions. The listener must be port 80 for HTTP traffic.

Step 1: Configure Load Balancer

Basic Configuration

To configure your load balancer, provide a name, select a scheme, specify one or more listeners, and select a network. The default configuration is an Internet-facing load balancer in the selected network with a listener that receives HTTP traffic on port 80.

Name ⓘ

Scheme ⓘ ☒ Internet-facing
☐ Internal

IP address type ⓘ

Listeners

A listener is a process that checks for connection requests, using the protocol and port that you configured.

Load Balancer Protocol	Load Balancer Port
<input type="text" value="HTTP"/>	<input type="text" value="80"/>

Availability Zones

Specify the Availability Zones to enable for your load balancer. The load balancer routes traffic to the targets in these Availability Zones only. You can specify only one subnet per Availability Zone. You must specify subnets from at least two Availability Zones to increase the availability of your load balancer.

VPC ⓘ vpc-029c3269 (172.31.0.0/16) (default) ▾

Availability Zones

<input type="checkbox"/>	us-east-2a	subnet-242ef84f	▾
<input type="checkbox"/>	us-east-2b	subnet-a7b9b5dd	▾
<input type="checkbox"/>	us-east-2c	subnet-b66314fa	▾

Availability Zones

Specify the Availability Zones to enable for your load balancer. The load balancer routes traffic to the targets in these Availability Zones only. You can specify only one subnet per Availability Zone. You must specify subnets from at least two Availability Zones to increase the availability of your load balancer.

VPC ⓘ vpc-029c3269 (172.31.0.0/16) (default) ▾

Availability Zones

<input checked="" type="checkbox"/>	us-east-2a	subnet-242ef84f	▾	IPv4 address ⓘ	Assigned by AWS
<input checked="" type="checkbox"/>	us-east-2b	subnet-a7b9b5dd	▾	IPv4 address ⓘ	Assigned by AWS
<input checked="" type="checkbox"/>	us-east-2c	subnet-b66314fa	▾	IPv4 address ⓘ	Assigned by AWS

Additional AWS services can be integrated with this load balancer at launch when you enable them below. You can also add these and other services after your load balancer is created by reviewing the "Integrated Services" tab for the selected load balancer.

AWS Global Accelerator ☐ Create an accelerator to get static IP addresses and improve the performance and availability of your application. [Learn more](#)
Additional charges apply

Your Accelerator will be created with the following name that you can customize. Once your Accelerator is created you can manage it from the Global Accelerator console.

Accelerator name

Maximum 64 characters. Letters and numbers only.

Tags

[Cancel](#) [Next: Configure Security Settings](#)

- Click Next and configure the security group.

Step 3: Configure Security Groups

A security group is a set of firewall rules that control the traffic to your load balancer. On this page, you can add rules to allow specific traffic to reach your load balancer. First, decide whether to create a new security group or select an existing one.

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

Security group name:

Description:

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ
Custom TCP F ▾	TCP	80	Custom ▾ 0.0.0.0/0, ::/0
Add Rule			

- Now configure target groups.

E & OE

Handouts: Drakhshan Bokhat

Step 4: Configure Routing

Your load balancer routes requests to the targets in this target group using the protocol and port that you specify, and performs health checks on the targets using these health check settings. The target group specify in this step will apply to all of the listeners configured on this load balancer; you can edit the listeners and add listeners after the load balancer is created.

Target group

Target group ⓘ

New target group

Name ⓘ

Target type

☒ Instance
☐ IP
☐ Lambda function

Protocol ⓘ

HTTP

Port ⓘ

80

Protocol version ⓘ

☒ HTTP1
Send requests to targets using HTTP/1.1. Supported when the request protocol is HTTP/1.1 or HTTP/2.
☐ HTTP2
Send requests to targets using HTTP/2. Supported when the request protocol is HTTP/2 or gRPC, but gRPC-specific features are not available.
☐ gRPC
Send requests to targets using gRPC. Supported when the request protocol is gRPC.

- You can set the path with any error.html file prompting any error say server down etc. The value for unhealthy threshold is 2 and for healthy its kept 5 i.e., no response is given by the server after 2 requests.

Health checks

Protocol ⓘ

HTTP

Path ⓘ

/

▼ Advanced health check settings

Port ⓘ

☒ traffic port
☐ override

Healthy threshold ⓘ

5

Unhealthy threshold ⓘ

2

Timeout ⓘ

5

seconds

Interval ⓘ

30

seconds

Success codes ⓘ

200

Cancel

Previous

Next: Register Targets

Step 5: Register Targets

Register targets with your target group. If you register a target in an enabled Availability Zone, the load balancer starts routing requests to the targets as soon as the registration process completes and the target passes the initial health checks.

Registered targets

To deregister instances, select one or more registered instances and then click Remove.

Remove

☐

Instance

▼

Name

▼

Port

▼

State

▼

Security groups

▼

Zone

▼

No instances available.

Instances

To register additional instances, select one or more running instances, specify a port, and then click Add. The default port is the port specified for the target group. If the instance is already registered on the specified port, you must specify a different port.

Add to registered

on port

80

E & OE

Handouts: Drakhshan Bokhat

- After registering the target attach the instances (running) and click review and create.

Step 5: Register Targets

<input type="checkbox"/>	Instance	Name	Port	State
<input type="checkbox"/>	i-00f4abdd4ad5f5e6	server-1	80	running
<input type="checkbox"/>	i-06cec12066938727b	server-2	80	running

Instances

To register additional instances, select one or more running instances, specify a port, you must specify a different port.

on port

<input type="checkbox"/>	Instance	Name	State
<input type="checkbox"/>	i-00f4abdd4ad5f5e6	server-1	running
<input type="checkbox"/>	i-06cec120669387...	server-2	running

Load Balancer Creation Status

Created security group	✓ Completed
Authorized security groups	✓ Completed
Create Load Balancer	✓ Completed
Create target group	✓ Completed
Add to registered	✓ Completed
Create Listener	⌛

Load Balancer Creation Status

- ✓ **Successfully created load balancer**
Load balancer `awais-elb` was successfully created.
Note: It might take a few minutes for your load balancer to be fully set up and ready to route traffic, and for the targets to complete the registration process and pass the initial health checks.
- Suggested next steps**
- Discover other services that you can integrate with your load balancer. Visit the **Integrated services** tab within `awais-elb`
 - Consider using AWS Global Accelerator to further improve the availability and performance of your applications. [AWS Global Accelerator console](#)

Close

The screenshot displays the AWS Management Console interface for an Elastic Load Balancing (ELB) instance. At the top, there are buttons for 'Create Load Balancer' and 'Actions'. Below this is a search bar and a table with columns: Name, DNS name, State, VPC ID, and Availability Zones. A single instance, 'awais-elb-794341748.us-eas...', is listed with a state of 'provisioning' and VPC ID 'vpc-029c3269'. The 'Load balancer: awais-elb' section is expanded, showing tabs for 'Description', 'Listeners', 'Monitoring', 'Integrated services', and 'Tags'. The 'Description' tab is active, displaying the 'Basic Configuration' section. This section includes fields for Name, ARN, DNS name, State, and Type. The 'DNS name' field is highlighted with a red box and contains the text 'awais-elb-794341748.us-east-2.elb.amazonaws.com' with a copy icon. The 'State' is 'provisioning' and the 'Type' is 'application'.

- Now use the DNS address of Load balancer and browse it. Refresh the page to see different servers responding to the request.

🔴🔴 How to avoid usage of public IP of servers if DNS of load balancer is used? *Hint in Security group timeout can be used.*