

## 1. Configure Classic Load balancer.

Create 2 ec2 instances in public subnet and deploy sample appl to default path i.e., /var/www/html/index.html:

**EC2-1 [ in the user\_data give the below script]:**

```
#!/bin/bash

Sudo dnf install httpd -y

echo "Server-01" >> /var/www/html/index.html

sudo systemctl enable httpd

sudo systemctl start httpd
```

**EC2-2 [ in the user\_data write the below script]:**

```
#!/bin/bash

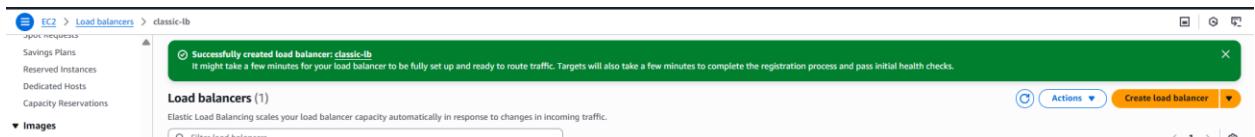
Sudo dnf install httpd -y

echo "Server-02" >> /var/www/html/index.html

sudo systemctl enable httpd

sudo systemctl start httpd
```

Create Load Balancer:



Select Classic Load Balancer:

<p>Choose an Application Load Balancer when you need a flexible feature set for your 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.</p> <p><a href="#">Create</a></p>	<p>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 applications. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies.</p> <p><a href="#">Create</a></p>	<p>Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-party virtual appliances that support GENEVE. These appliances enable you to improve security, compliance, and policy controls.</p> <p><a href="#">Create</a></p>
---	--	---

▼ Classic Load Balancer - previous generation

**Classic Load Balancer** [Info](#)

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

[Create](#)

[Close](#)

Give the name and Internet facing:

**Create Classic Load Balancer** [Info](#)

The Classic Load Balancer distributes incoming application traffic across multiple EC2 instance targets in multiple Availability Zones. This increases the fault tolerance of your application only to healthy instances.

► How Classic Load Balancers work

**Basic configuration**

**Load balancer name**  
Name must be unique within your AWS account and can't be changed after the load balancer is created.  
  
A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

**Scheme** [Info](#)  
Scheme can't be changed after the load balancer is created.

**Internet-facing**  
• Serves internet-facing traffic.  
• Has public IP addresses.  
• DNS name resolves to public IPs.  
• Requires a public subnet.

**Internal**  
• Serves internal traffic.  
• Has private IP addresses.  
• DNS name resolves to private IPs.

Select the VPC and the subnet in which instances are there:

### Network mapping Info

The load balancer routes traffic to targets in the selected subnets, and in accordance with your network settings.

VPC | [Info](#)

loadBalancers.vpcDescriptionCblInternetFacing [Learn more](#)

vpc-038d9ffa8289cfaf

172.31.0.0/16

(default) ▾



[Create VPC](#)

#### Availability Zones and subnets

Select at least one Availability Zone and one subnet for each zone. We recommend selecting at least two Availability Zones. The load balancer will route traffic only to targets in the selected Availability Zones. Availability Zones that are not available for selection.

us-east-1a (use1-az1)

Subnet

Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.

subnet-00a7a7abf3a434492

IPv4 subnet CIDR: 172.31.0.0/20

#### IPv4 address

Assigned by AWS

us-east-1b (use1-az2)

Subnet

Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.

subnet-06062b88e98ea381

IPv4 subnet CIDR: 172.31.80.0/20

#### IPv4 address

Assigned by AWS

us-east-1c (use1-az4)

us-east-1d (use1-az6)

us-east-1e (use1-az3)

us-east-1f (use1-az5)

## Select the ‘Security group’, ‘Listeners’ and health checks:

### Security groups Info

A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can [create a new security group](#).

#### Security groups

Select up to 5 security groups



default

sg-0589a658b873b7f69 VPC: vpc-038d9ffa8289cfaf

### Listeners and routing Info

A listener is a process that checks for connection requests using the protocol and port you configure. The settings you define for a listener determine how the load balancer routes requests to its registered targets.

▼ Listener HTTP:80

Instance HTTP:80

Listener protocol

HTTP

Listener port

80

1-65535

Instance protocol

HTTP

Instance port

80

1-65535

[Add listener](#)

You can add up to 99 more listeners.

### Health checks Info

Your load balancer automatically performs health checks to test the availability of all registered instances. Traffic is only routed to healthy instances, which is determined on their response to the health check.

#### Ping target

The health check ping is sent using the protocol and port you specify. If using HTTP/HTTPS protocol, you must also provide the destination path.

Ping protocol

HTTP

Ping port

80

Ping path

/index.html

## In advance health checks give your values:

**▼ Advanced health check settings**

**Response timeout**  
Time to wait for EC2 instances to respond to health checks.  
 seconds  
2-60 seconds. Must be less than the health check interval.

**Interval**  
Amount of time between health checks sent to EC2 instances.  
 seconds  
5-300 seconds. Must be greater than the health check response timeout.

**Unhealthy threshold**  
Number of consecutive health check failures before declaring an EC2 instance unhealthy.  
 2-10

**Healthy threshold**  
Number of consecutive health check successes before declaring an EC2 instance healthy.  
 2-10

**Restore defaults**

For adding the instances to this Load Balancer click on 'Add instances':

**Instances (0)**

You can add instances to register as targets of the load balancer. Alternatively, after your load balancer is created, you can add it to an Amazon EC2 Auto Scaling group to ensure you maintain the correct number of instances to handle the load for your application. For maximum fault tolerance, we recommend maintaining approximately equivalent numbers of instances in each Availability Zone.

<input type="button" value="Filter instances"/>	<	1	>	<input type="button" value="Remove"/>	<input type="button" value="Add instances"/>								
<input type="checkbox"/> Instance ID	<input type="button" value="▼"/>	Name	<input type="button" value="▼"/>	State	<input type="button" value="▼"/>	Security groups	<input type="button" value="▼"/>	Zone	<input type="button" value="▼"/>	Public IPv4 address	<input type="button" value="▼"/>	Subnet ID	<input type="button" value="▼"/>
No instances added													

Select your required instances:

**Add instances**

Select EC2 instances to register to your load balancer. Requests will be routed to registered instances that meet the health check requirements. For maximum fault tolerance, we recommend maintaining approximately equivalent numbers of instances in each Availability Zone enabled for the load balancer. If demand on your instances changes, you can register or deregister instances without disrupting the flow of requests to your application. [Learn more](#)

VPC  
vpc-038d9ffa8289cfac

**Available instances (2/2)**

<input type="checkbox"/> Instance ID	<input type="button" value="▼"/>	Name	<input type="button" value="▼"/>	State	<input type="button" value="▼"/>	Security groups	<input type="button" value="▼"/>	Zone	<input type="button" value="▼"/>	Public IPv4 address	<input type="button" value="▼"/>	Subnet ID	<input type="button" value="▼"/>	Launch time
<input checked="" type="checkbox"/> i-0e129113024d1ecca	<input type="button" value="▼"/>	Server-02	<input checked="" type="button" value="Running"/>	allow-all	<input type="button" value="▼"/>	us-east-1b	<input type="button" value="▼"/>	54.197.65.88	<input type="button" value="▼"/>	subnet-06062b88e98ea381	<input type="button" value="▼"/>	October 8, 2025,		
<input checked="" type="checkbox"/> i-046f0d95a5fa766dd	<input type="button" value="▼"/>	instance-1	<input checked="" type="button" value="Running"/>	allow-all	<input type="button" value="▼"/>	us-east-1a	<input type="button" value="▼"/>	35.170.63.80	<input type="button" value="▼"/>	subnet-00a7a7abf3a434492	<input type="button" value="▼"/>	October 8, 2025,		

Enable cross zone load balancing, and connection Draining:

**Attributes**

Creating your load balancer using the console gives you the opportunity to specify additional features at launch. You can also find and adjust these settings in the load balancer's "Attributes" section after your load balancer is created.

**Enable cross-zone load balancing**  
With cross-zone load balancing, each load balancer node for your Classic Load Balancer distributes requests evenly across the registered instances in all enabled Availability Zones. If cross-zone load balancing is disabled, each load balancer node distributes requests evenly across the registered instances in its Availability Zone only. Classic Load Balancers created with the API or CLI have cross-zone load balancing disabled by default. After you create a Classic Load Balancer, you can enable or disable cross-zone load balancing at any time.

**Enable connection draining**  
Applicable to instances that are deregistering, this feature allows existing connections to complete (during a specified draining interval) before reporting the instance as deregistered. [Learn more](#)

**Timeout (draining interval)**  
The maximum time for the load balancer to allow existing connections to complete. When the maximum time limit is reached, the load balancer forcibly closes any remaining connections and reports the instance as deregistered.

<input type="text" value="300"/> seconds
--

Valid values: 1-3600 (integers only)

And click on Create Load Balancer:

## Review

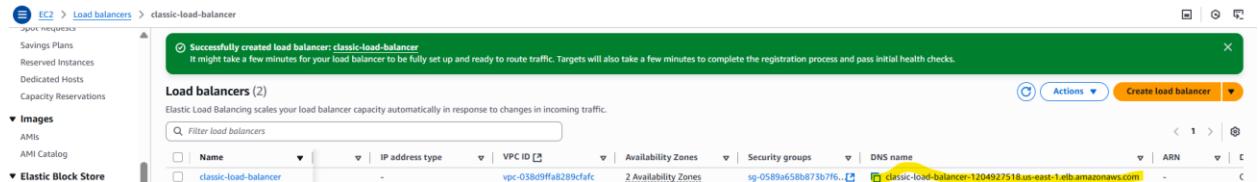
Review the load balancer configurations and make changes if needed. After you finish reviewing the configurations, choose **Create load balancer**.

Summary			
<b>Basic configuration</b> <a href="#">Edit</a> Name: classic-load-balancer Scheme: Internet-facing	<b>Network mapping</b> <a href="#">Edit</a> VPC: vpc-038d9ffa8289cfac Availability Zones and subnets: <ul style="list-style-type: none"><li>us-east-1a <a href="#">subnet-00a7a7abf3a434492</a></li><li>us-east-1b <a href="#">subnet-06062b88ea98ea381</a></li></ul>	<b>Security groups</b> <a href="#">Edit</a> default <a href="#">sg-0589a658b873b7f69</a>	<b>Listeners and routing</b> <a href="#">Edit</a> HTTP:80
<b>Health checks</b> <a href="#">Edit</a> HTTP:80/index.html <ul style="list-style-type: none"><li>Timeout: 2 seconds</li><li>Interval: 5 seconds</li><li>Unhealthy threshold: 2</li><li>Healthy threshold: 2</li></ul>	<b>Instances</b> <a href="#">Edit</a> 2 instances added <ul style="list-style-type: none"><li>1 instance in us-east-1a</li><li>1 instance in us-east-1b</li></ul>	<b>Attributes</b> <a href="#">Edit</a> <ul style="list-style-type: none"><li>Cross-zone load balancing: On</li><li>Connection draining: On</li><li>Connection draining timeout: 300 seconds</li></ul>	<b>Tags</b> <a href="#">Edit</a> <ul style="list-style-type: none"><li>-</li></ul>

[Cancel](#)

[Create load balancer](#)

Copy the load Balancer Domain Name:



The screenshot shows the AWS EC2 Load Balancers page. A green success message at the top states: "Successfully created load balancer: classic-load-balancer. It might take a few minutes for your load balancer to be fully set up and ready to route traffic. Targets will also take a few minutes to complete the registration process and pass initial health checks." Below the message, the "Load balancers (2)" section is displayed. The second load balancer listed is "classic-load-balancer". The "DNS name" column shows the value "classic-load-balancer-1204927518.us-east-1.elb.amazonaws.com".

And access from the browser:



Server-01

Refresh the browser:



Server-02

-----done-----

## 2. Configure Application Load balancer.

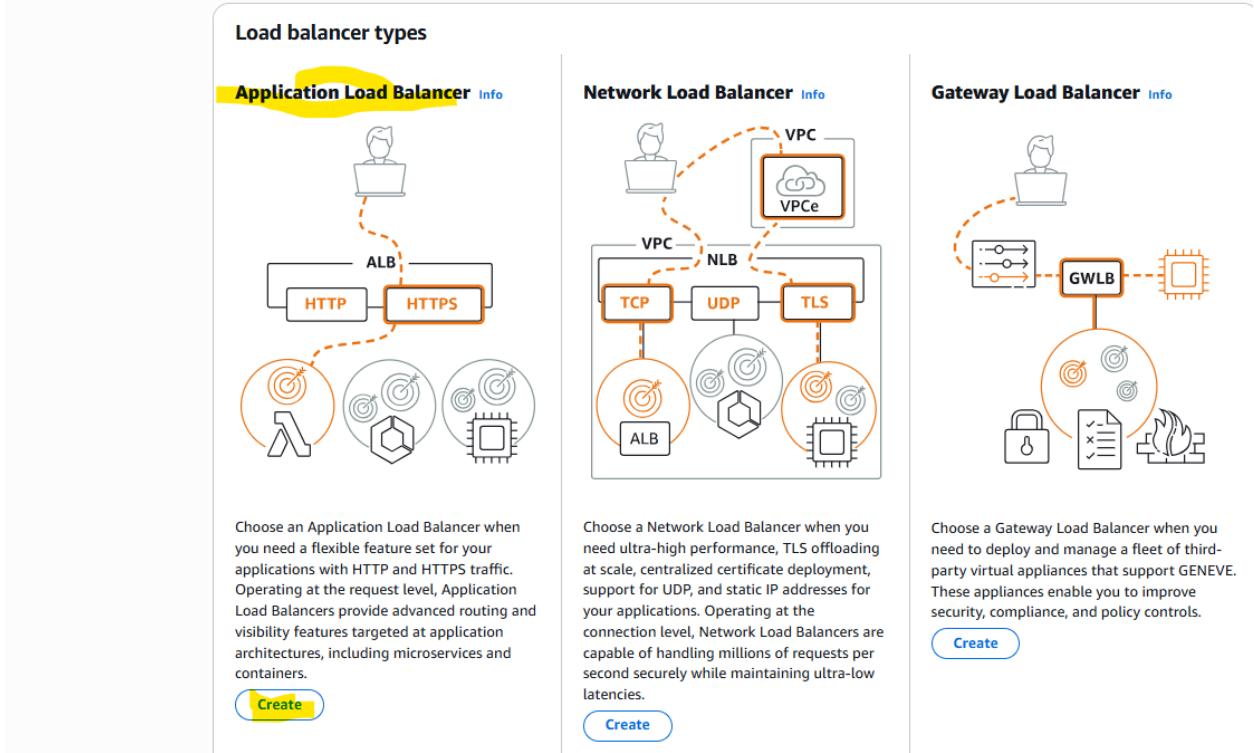
Create Load Balancer:



The screenshot shows the AWS EC2 Load Balancers page. The "Load balancers (1/1)" section displays one entry for "classic-load-balancer". The "DNS name" column shows the value "classic-load-balancer-1204927518.us-east-1.elb.amazonaws.com".

Create Application Load Balancer:

A complete feature-by-feature comparison along with detailed highlights is also available. [Learn more](#)



Give the name to application load balancer and select 'internet-facing':

#### Create Application Load Balancer [Info](#)

The Application Load Balancer distributes incoming HTTP and HTTPS traffic across multiple targets such as Amazon EC2 instances, microservices, and containers, based on request attributes. When the load balancer receives a connection request, it evaluates the listener rules in priority order to determine which rule to apply, and if applicable, it selects a target from the target group for the rule action.

##### ▶ How Application Load Balancers work

##### Basic configuration

###### Load balancer name

Name must be unique within your AWS account and can't be changed after the load balancer is created.

appl-lb

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

###### Scheme [Info](#)

Scheme can't be changed after the load balancer is created.

Internet-facing

- Serves internet-facing traffic.
- Has public IP addresses.
- DNS name resolves to public IPs.
- Requires a public subnet.

Internal

- Serves internal traffic.
- Has private IP addresses.
- DNS name resolves to private IPs.
- Compatible with the IPv4 and Dualstack IP address types.

###### Load balancer IP address type [Info](#)

Select the front-end IP address type to assign to the load balancer. The VPC and subnets mapped to this load balancer must include the selected IP address types. Public IPv4 addresses have an additional cost.

IPv4

Includes only IPv4 addresses.

Dualstack

Includes IPv4 and IPv6 addresses.

Dualstack without public IPv4

Includes a public IPv6 address, and private IPv4 and IPv6 addresses. Compatible with Internet-facing load balancers only.

Select your vpc and AZ:

**Network mapping** [Info](#)

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

**VPC** [Info](#)

The load balancer will exist and scale within the selected VPC. The selected VPC is also where the load balancer targets must be hosted unless routing to Lambda or on-premises targets, or if using VPC peering. To confirm the VPC for your targets, view [target groups](#).

vpc-038d9ffa8289cfac  
172.31.0.0/16 (default) ⟳ Create VPC

**IP pools** [Info](#)

You can optionally choose to configure an IPAM pool as the preferred source for your load balancers IP addresses. Create or view [Pools](#) in the [Amazon VPC IP Address Manager console](#).

Use IPAM pool for public IPv4 addresses

The IPAM pool you choose will be the preferred source of public IPv4 addresses. If the pool is depleted IPv4 addresses will be assigned by AWS.

**Availability Zones and subnets** [Info](#)

Select at least two Availability Zones and a subnet for each zone. A load balancer node will be placed in each selected zone and will automatically scale in response to traffic. The load balancer routes traffic to targets in the selected Availability Zones only.

us-east-1a (use1-az1)  
Subnet  
Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.  
subnet-00a7a7abf5a434492  
IPv4 subnet CIDR: 172.31.0.0/20 ▼

us-east-1b (use1-az2)  
Subnet  
Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.  
subnet-06062b88ea98ea381  
IPv4 subnet CIDR: 172.31.80.0/20 ▼

us-east-1c (use1-az4)

us-east-1d (use1-az6)

us-east-1e (use1-az3)

us-east-1f (use1-az5)

## Select the Security group:

**Security groups** [Info](#)

A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can [create a new security group](#).

**Security groups**

Select up to 5 security groups ▼ ⟳

default ×  
sg-0589a658b873b7f69 VPC: vpc-038d9ffa8289cfac

Give protocol and port and select ‘Routing action’ “Forward to target groups” and click on ‘Create target groups’:

**Listeners and routing** [Info](#)

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

▼ Listener HTTP:80 Remove

Protocol	Port
HTTP	80 1-65535

**Default action** [Info](#)

The default action is used if no other rules apply. Choose the default action for traffic on this listener.

Forward to target groups  Redirect to URL  Return fixed response

**Forward to target group** [Info](#)

Choose a target group and specify routing weight [or create target group](#).

Target group	Weight	Percent
Select a target group <span>▼</span> <span>⟳</span>	1 0-999	100%

**+ Add target group**

You can add up to 4 more target groups.

**Target group stickiness** [Info](#)

Enables the load balancer to bind a user's session to a specific target group. To use stickiness the client must support cookies. If you want to bind a user's session to a specific target, turn on the Target Group attribute Stickiness.

Turn on target group stickiness

**Listener tags - optional**

Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

Add listener tag

Choose ‘target type’ as ‘instances’:

**Create target group**  
Your load balancer routes requests to the targets in a target group and performs health checks on the targets.

**Basic configuration**  
Settings in this section can't be changed after the target group is created.

**Choose a target type**

- Instances
  - Supports load balancing to instances within a specific VPC.
  - Facilitates the use of [Amazon EC2 Auto Scaling](#) to manage and scale your EC2 capacity.
- IP addresses
  - Supports load balancing to VPC and on-premises resources.
  - Facilitates routing to multiple IP addresses and network interfaces on the same instance.
  - Offers flexibility with microservice-based architectures, simplifying inter-application communication.
  - Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6 NAT.
- Lambda function
  - Facilitates routing to a single Lambda function.
  - Accessible to Application Load Balancers only.
- Application Load Balancer
  - Offers the flexibility for a Network Load Balancer to accept and route TCP requests within a specific VPC.
  - Facilitates using static IP addresses and PrivateLink with an Application Load Balancer.

Give target group name , Protocol, port, ip address type, vpc and protocol version:

**Target group name**

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

**Protocol**  
Protocol for load balancer-to-target communication. Can't be modified after creation.

HTTP

**Port**  
Port number where targets receive traffic. Can be overridden for individual targets during registration.

1-65535

**IP address type**  
Only targets with the indicated IP address type can be registered to this target group.

IPv4

Each instance has a default network interface (eth0) that is assigned the primary private IPv4 address. The instance's primary private IPv4 address is the one that will be applied to the target.

IPv6

Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). [Learn more](#)

**VPC**  
Select the VPC with the instances that you want to include in the target group. Only VPCs that support the IP address type selected above are available in this list.

(default) [Create VPC](#)

**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.

Give the health checks and health check path and click on Next:

**Health checks**

The associated load balancer periodically sends requests, per the settings below, to the registered targets to test their status.

**Health check protocol**

HTTP

**Health check path**

Use the default path of "/" to perform health checks on the root, or specify a custom path if preferred.

/

Up to 1024 characters allowed.

► Advanced health check settings

**Attributes**

① Certain default attributes will be applied to your target group. You can view and edit them after creating the target group.

► Tags - optional

Consider adding tags to your target group. Tags enable you to categorize your AWS resources so you can more easily manage them.

Cancel

Next

Select your required instances and click on 'include as pending below':

EC2 > Target groups > Create target group

Step 1 Create target group  
Step 2 Register targets

**Register targets**

This is an optional step to create a target group. However, to ensure that your load balancer routes traffic to this target group you must register your targets.

**Available instances (1/2)**

Instance ID	Name	State	Security groups	Zone	Private IPv4 address	Subnet ID
i-0e129113024d1ecca	Server-02	Running	allow-all	us-east-1b	172.31.86.156	subnet-06062b8bea98ea1
<input checked="" type="checkbox"/> i-046f0d95a5fa766dd	instance-1	Running	allow-all	us-east-1a	172.31.1.35	subnet-00a7a7abf3a4344

1 selected

**Ports for the selected instances**

Ports for routing traffic to the selected instances.

80

1-65535 (separate multiple ports with commas)

**Include as pending below**

As I want only 1 instance so selected one:

EC2 > Target groups > Create target group

**Review targets**

**Targets (1)**

Instance ID	Name	Port	State	Security groups	Zone	Private IPv4 address	Subnet ID	Launch time
i-046f0d95a5fa766dd	instance-1	80	Running	allow-all	us-east-1a	172.31.1.35	subnet-00a7a7abf3a434492	October 8, 2025, 11:28 (UTC+05:30)

1 pending

Remove all pending

Cancel Previous Create target group

And create target group:

Goto Load balancer and here the target group should available in the Target group section:

#### Listeners and routing Info

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

And click on Create , then you can see the status as Provisioning:

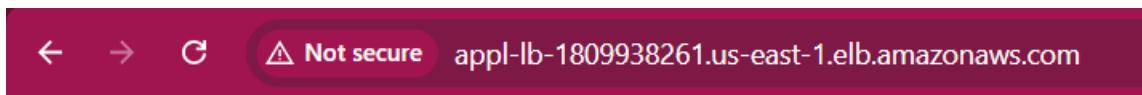
Then it should come to status as 'Active':

The screenshot shows the AWS EC2 Load Balancers console for the 'appl-lb' load balancer. The 'Resource map' tab is selected, providing a visual representation of the load balancer's architecture. It shows a single listener (HTTP:80) mapping to a rule (Priority default: Forward default to target group). This rule has a condition (If no other rule applies). The target group is an 'Instance: HTTP target-grp' with one target (I-046fd95a5fa76dd) which is healthy.

Then copy the application load balancer DNS name:

Name	Scheme	IP address type	VPC ID	Availability Zones	Security groups	DNS name	ARN
appl-lb	Internet-facing	IPv4	vpc-038d9ffa8289cfaf	2 Availability Zones	sg-0589a658b875b7f6	appl-lb-1809938261.us-east-1.elb.amazonaws.com	arn:aws:elasticloadbalancing:us-east-1:931208603575:loadbalancer/app/appl-lb/68e4f713366a84e4

And access from the browser:



Server-01

Note: this is the default response means if anyone is trying to access our DNS without giving any path prefix then he/she will get this response.

Next, if anyone is giving the path prefix like after DNS/images then it should route to that particular path so for that login to server and create an images directory in the `/var/www/html/` and goto images directory and create an 'index.html' file and give sample content in it:

```

[ec2-user@ip-172-31-1-35 ~]$ cd
[ec2-user@ip-172-31-1-35 ~]$ ls
[ec2-user@ip-172-31-1-35 ~]$ ls -l /var/www/html/
total 4
drwxr-xr-x. 2 root root 24 oct 8 06:47 images
-rw-r--r--. 1 root root 10 oct 8 05:59 index.html
[ec2-user@ip-172-31-1-35 ~]$ ls -l /var/www/html/images/
total 4
-rw-r--r--. 1 root root 29 oct 8 06:47 index.html
[ec2-user@ip-172-31-1-35 ~]$ cat /var/www/html/images/
cat: /var/www/html/images/: Is a directory
[ec2-user@ip-172-31-1-35 ~]$ cat /var/www/html/images/index.html
<h1>Welcome to Images </h1>

```

Select your Load Balancer and goto 'Listeners and Rules' tab and click on Manage rules->Add rule:

The screenshot shows the AWS CloudFormation console with the path: EC2 > Load balancers > appl-lb. The 'Listeners and rules' tab is selected. A yellow box highlights the 'Manage rules' button. Below it, a table lists a single rule for port 80, which forwards requests to a target group named 'target-gp'.

Protocol:Port	Rules	ARN	Security policy	Actions
HTTP:80	Forward to target group target-gp (1 (100%)) Target group stickiness: Off	1 rule	Not applicable	Manage rule ▲   Add rule ▼   Edit rules   Reprioritize rules   Default security configuration   mTLS   Trust store

Click on Add condition and select 'Path':

The screenshot shows the 'Add rule' wizard, Step 1: Add rule. The 'Add condition' button is highlighted. The interface includes fields for Listener details (HTTP:80), Name and tags, and Conditions (0 values).

Give the priority as any number:

EC2 > Load balancers > app-lb > HTTP:80 listener > Add rule

Step 1: Add rule  
Step 2: Set rule priority (selected)  
Step 3: Review and create

**Set rule priority** Info  
Each rule has a priority. The default rule is evaluated last. You can change the priority of a non-default rule at any time. You can't change the priority of the default rule.

**Listener details: HTTP:80**

**Listener rules (2) Info**  
Traffic received by the listener is routed according to the default action and any additional rules. Rules are evaluated in priority order from the lowest value to the highest value.

Priority	Name tag	Conditions (If)	Actions (Then)	ARN	Tags
2	-	Path = /images Priority value must be 1-50,000.	<ul style="list-style-type: none"> <li>Forward to target group target-grp [1] 1 (100%) Target group stickiness: Off</li> </ul>	Pending	0 tags
Last (default)	Default	If no other rule applies	<ul style="list-style-type: none"> <li>Forward to target group target-grp [1] 1 (100%) Target group stickiness: Off</li> </ul>	ARN	0 tags

Rule limits: Reset priorities Add gap between priorities

Cancel Previous Next

Click on Add rule:

EC2 > Load balancers > app-lb > HTTP:80 listener > Add rule

Step 1: Add rule  
Step 2: Set rule priority  
Step 3: Review and create (selected)

**Review and create**

**Listener details: HTTP:80**

**Rule details**

Priority 2	Conditions If request matches all: Path = /images	Actions Forward to target group target-grp [1] 1 (100%) Target group stickiness: Off
---------------	---	---

**Rule ARN**  
Pending

**Rule tags (0)**  
Edit  
Tags can help you manage, identify, organize, search for and filter resources.  
Key Value  
No tags found

**Server-side tasks and status**  
After completing and submitting the above steps, all server-side tasks and their statuses become available for monitoring.

Cancel Previous Add rule

We can see there are 2 rules:

EC2 > Load balancers

**Load balancers (1/2)**  
Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Name	State	Type	Scheme	IP address type	VPC ID	Availability Zones	Security groups	DNS name
app-lb	Active	application	Internet-facing	IPv4	vpc-038d9ffea8289cfac	2 Availability Zones	sg-0589a658b873b7f6...	alb-1809958261.us-east-1.elb.a...
alb-1	Active	application	Internet-facing	IPv4	vpc-038d9ffea8289cfac	2 Availability Zones	sg-03bf157b19a55e3b...	alb-1-49239446.us-east-1.elb.amaz...

**Load balancer: app-lb**

Details **Listeners and rules** Network mapping Resource map Security Monitoring Integrations Attributes Capacity Tags

**Listeners and rules (1) Info**  
A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.

Protocol:Port	Default action	Rules	ARN	Security policy	Default SSL/TLS certificate	mTLS	Trust store
HTTP:80	<ul style="list-style-type: none"> <li>Forward to target group target-grp [1] 1 (100%) Target group stickiness: Off</li> </ul>	2 rules	ARN	Not applicable	Not applicable	Not applicable	Not applicable

The screenshot shows the AWS CloudFormation console with the following details:

- Stack Name:** app-lb
- Status:** CREATE\_COMPLETE
- Region:** us-east-1
- Outputs:**
  - app-lb: arn:aws:elasticloadbalancing:us-east-1:931208603575:listener/app/app-lb/68e4f713366a84e4/da23050ae22bb5ea

Copy the DNS of load balancer and browse with /images:

The browser window displays the following information:

- Address Bar:** appl-lb-1809938261.us-east-1.elb.amazonaws.com/images/
- Page Content:** Welcome to Images

Without /images when you browse the load balancer DNS:

The browser window displays the following information:

- Address Bar:** appl-lb-1809938261.us-east-1.elb.amazonaws.com
- Page Content:** Server-01

-----done-----

### 3. Configure Network Load balancer.

Goto Load Balancer and click on Create Load Balancer, and give the name and select 'Internet facing' or 'internal':

## Create Network Load Balancer Info

The Network Load Balancer distributes incoming TCP and UDP traffic across multiple targets such as Amazon EC2 instances, microservices, and containers. When the load balancer receives a connection port that are specified in the listener configuration, and the routing rule specified as the default action.

### ► How Network Load Balancers work

#### Basic configuration

##### Load balancer name

Name must be unique within your AWS account and can't be changed after the load balancer is created.

my-network-lb

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

##### Scheme

Scheme can't be changed after the load balancer is created.

Internet-facing

- Serves internet-facing traffic.
- Has public IP addresses.
- DNS name resolves to public IPs.
- Requires a public subnet.

Internal

- Serves internal traffic.
- Has private IP addresses.
- DNS name resolves to private IPs.

##### Load balancer IP address type Info

Select the front-end IP address type to assign to the load balancer. The VPC and subnets mapped to this load balancer must include the selected IP address types.

IPv4

Includes only IPv4 addresses.

Dualstack

Includes IPv4 and IPv6 addresses.

## Select the VPC and the subnet where instances are deployed:

#### Network mapping Info

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

##### VPC

The load balancer will exist and scale within the selected VPC. The selected VPC is also where the load balancer targets must be hosted unless routing to on-premises targets or using VPC peering. To confirm the VPC for your targets, view target groups Info.

vpc-05f40aba9acea42ad (default-vpc)

172.31.0.0/16

(default) ▾



Create VPC Info

##### Availability Zones and subnets

Select one or more Availability Zones and corresponding subnets. Enabling multiple Availability Zones increases the fault tolerance of your applications. The load balancer routes traffic to targets in the selected Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

us-east-1a (use1-az2)

Subnet

Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.

subnet-0d2d45ea99987b45

IPv4 subnet CIDR: 172.31.80.0/20

IPv4 address

The front-end IPv4 address of the load balancer in the selected Availability Zone.

Assigned by AWS

Use an Elastic IP address

us-east-1b (use1-az4)

Subnet

Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.

subnet-0242f46eea951417e

IPv4 subnet CIDR: 172.31.16.0/20

IPv4 address

The front-end IPv4 address of the load balancer in the selected Availability Zone.

Assigned by AWS

Use an Elastic IP address

us-east-1c (use1-az6)

## Select security group:

#### Security groups Info

A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can [create a new security group](#) Info.

##### Security groups - recommended

Security groups support on Network Load Balancers can only be enabled at creation by including at least one security group. You can change security groups after creation. The security groups for your load balancer must allow it to communicate with registered targets on both the listener port and the health check port. For PrivateLink Network Load Balancers, security group rules are enforced on PrivateLink traffic; however, you can turn off inbound rule evaluation after creation within the load balancer's Security tab or using the API.

Select up to 5 security groups

▼



default

sg-0e64ae6a546f0ad43 VPC: vpc-05f40aba9acea42ad

## Provide protocol and port:

**Listeners and routing** [Info](#)

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

▼ Listener TCP:80

Protocol	Port
TCP	80 1-65535

Default action [Info](#)

Forward to [Select a target group](#) [C](#)

Create target group [A](#)

**Listener tags - optional**

Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

Add listener tag

You can add up to 50 more tags.

Add listener

You can add up to 49 more listeners.

And click on Create target group and select 'Instances':

EC2 > Target groups > Create target group

Step 1 [Create target group](#)

Step 2 [Register targets](#)

**Create target group**

Your load balancer routes requests to the targets in a target group and performs health checks on the targets.

**Basic configuration**

Settings in this section can't be changed after the target group is created.

**Choose a target type**

Instances

- Supports load balancing to instances within a specific VPC.
- Facilitates the use of [Amazon EC2 Auto Scaling](#) to manage and scale your EC2 capacity.

IP addresses

- Supports load balancing to VPC and on-premises resources.
- Facilitates routing to multiple IP addresses and network interfaces on the same instance.
- Offers flexibility with microservice based architectures, simplifying inter-application communication.
- Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6 NAT.

Lambda function

- Facilitates routing to a single Lambda function.
- Accessible to Application Load Balancers only.

Application Load Balancer

- Offers the flexibility for a Network Load Balancer to accept and route TCP requests within a specific VPC.
- Facilitates using static IP addresses and PrivateLink with an Application Load Balancer.

And give the target group name:

**Target group name**

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

**Protocol**

Protocol for load balancer-to-target communication. Can't be modified after creation.

TCP	Port
80 1-65535	

**IP address type**

Only targets with the indicated IP address type can be registered to this target group.

IPv4

Each instance has a default network interface (eth0) that is assigned the primary private IPv4 address. The instance's primary private IPv4 address is the one that will be applied to the target.

IPv6

Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). [Learn more](#) [A](#)

**VPC**

Select the VPC with the instances that you want to include in the target group. Only VPCs that support the IP address type selected above are available in this list.

vinc-05f40aba9acea42ad (default-vpc) 172.31.0.0/16	(default) <a href="#">C</a>	<a href="#">Create VPC</a> <a href="#">A</a>
---	-----------------------------	--

Select Health checks:

## Health checks

The associated load balancer periodically sends requests, per the settings below, to the registered targets to test their status.

### Health check protocol

HTTP



### Health check path

Use the default path of "/" to perform health checks on the root, or specify a custom path if preferred.

/



Up to 1024 characters allowed.

### ▼ Advanced health check settings

[Restore defaults](#)

#### Health check port

The port the load balancer uses when performing health checks on targets. By default, the health check port is the same as the target group's traffic port. However, you can specify a different port as an override.

Traffic port

Override

#### Healthy threshold

The number of consecutive health checks successes required before considering an unhealthy target healthy.

5

2-10

#### Unhealthy threshold

The number of consecutive health check failures required before considering a target unhealthy.

2

2-10

#### Timeout

The amount of time, in seconds, during which no response means a failed health check.

6

seconds

2-120

#### Interval

The approximate amount of time between health checks of an individual target.

30

seconds

5-300

#### Success codes

The HTTP codes to use when checking for a successful response from a target. You can specify multiple values (for example, "200,202") or a range of values (for example, "200-299").

200-399

And click on Next

## Attributes

ⓘ Certain default attributes will be applied to your target group. You can view and edit them after creating the target group.

## ► Tags - optional

Consider adding tags to your target group. Tags enable you to categorize your AWS resources so you can more easily manage them.

[Cancel](#)

[Next](#)

And add the required instances by clicking on 'Include as pending below':

EC2 > Target groups > Create target group

**Step 1**  
Create target group  
**Step 2**  
**Register targets**

### Register targets

This is an optional step to create a target group. However, to ensure that your load balancer routes traffic to this target group you must register your targets.

Available instances (2/2)							
Instance ID	Name	State	Security groups	Zone	Private IPv4 address	Subnet ID	
i-0d80bee0b52c6929	instance-2	Running	default	us-east-1b	172.31.21.163	subnet-0242f46eea95141	
i-0a9b39b2828f0e620	instance-1	Running	default	us-east-1a	172.31.82.208	subnet-0d2d45ea999877	

**2 selected**

**Ports for the selected instances**  
Ports for routing traffic to the selected instances.  
80  
1-45555 (separate multiple ports with commas)  
**Include as pending below**

### Review targets

**Targets (0)**

Instance ID	Name	Port	State	Security groups	Zone	Private IPv4 address	Subnet ID	Launch time
-------------	------	------	-------	-----------------	------	----------------------	-----------	-------------

**0 selected**

**Ports for the selected instances**  
Ports for routing traffic to the selected instances.  
80  
1-45555 (separate multiple ports with commas)  
**Include as pending below**

2 selections are now pending below. Include more or register targets when ready.

And click on Target group:

EC2 > Target groups > Create target group

Instance ID	Name	State	Security groups	Zone	Private IPv4 address	Subnet ID
i-0d80bee0b52c6929	instance-2	Running	default	us-east-1b	172.31.21.163	subnet-0242f46eea95141
i-0a9b39b2828f0e620	instance-1	Running	default	us-east-1a	172.31.82.208	subnet-0d2d45ea999877

**0 selected**

**Ports for the selected instances**  
Ports for routing traffic to the selected instances.  
80  
1-45555 (separate multiple ports with commas)  
**Include as pending below**

2 selections are now pending below. Include more or register targets when ready.

### Review targets

**Targets (2)**

Instance ID	Name	Port	State	Security groups	Zone	Private IPv4 address	Subnet ID	Launch time
i-0d80bee0b52c6929	instance-2	80	Running	default	us-east-1b	172.31.21.163	subnet-0242f46eea95141e	October 9, 2025, 11:59 (UTC+05:30)
i-0a9b39b2828f0e620	instance-1	80	Running	default	us-east-1a	172.31.82.208	subnet-0d2d45ea999877b45	October 9, 2025, 11:57 (UTC+05:30)

**2 pending**

**Create target group**

Goto load balancer and select your target group:

EC2 > Load balancers > Create Network Load Balance

### Listeners and routing info

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

**Listener TCP:80**

Protocol	Port
TCP	80

Default action: **Select a target group**

**Listener tags - optional**  
Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.  
**Add listener tag**  
You can add up to 50 more tags.

**Add listener**  
You can add up to 49 more listeners.

**Load balancer tags - optional**  
Consider adding tags to your load balancer. Tags enable you to categorize your AWS resources so you can more easily manage them. The 'Key' is required, but 'Value' is optional. For example, you can have Key = production-webserver, or Key = webserver, and Value = production.

**Optimize with service integrations - optional**

Optimize your load balancing architecture by integrating AWS services with this load balancer at launch. You can also add these and other services after your load balancer is created by reviewing the load balancer's "Integrations" tab.

And click on 'Create Load Balancer':

The screenshot shows the 'Review' step of creating a load balancer. It includes a 'Summary' section with basic configuration (Name: my-network-lb, Scheme: Internet-facing, IP address type: IPv4), Network mapping (VPC: vpc-05f40aba9ace42ad, Availability Zones and subnets: us-east-1a [subnet-012d45ea999877b45], us-east-1b [subnet-0242f46eea951417e]), Security groups (default sg-0e64ae6a346f0ad43), and Listeners and routing (TCP:80 | Target group: my-tg). Below this, there are sections for Service integrations (AWS Global Accelerator: -) and Attributes, with a note that certain default attributes will be applied. At the bottom, there's a 'Creation workflow and status' section with a 'Server-side tasks and status' box indicating completion, and a 'Cancel' and 'Create load balancer' button.

Access from the browser using load balancer DNS:

The screenshot shows a browser window with the URL my-nlb-23c42cbfaaa4c4a.elb.us-east-1.amazonaws.com. The page displays the text "Welcome to nginx! Server-01". Below it, a message states: "If you see this page, the nginx web server is successfully installed and working. Further configuration is required." It also provides links to online documentation at nginx.org and commercial support at nginx.com, ending with a thank you message.

The screenshot shows a browser window with the same URL. The page displays the text "Welcome to nginx! Server-02". Below it, a message states: "If you see this page, the nginx web server is successfully installed and working. Further configuration is required." It also provides links to online documentation at nginx.org and commercial support at nginx.com, ending with a thank you message.

-----done-----

4. Attach SSL for application load balancer.

Select your Load Balancer and goto ‘Integration’ tab and ->WAF and click on ‘Manage CloudFront+WAF Integration’:

The screenshot shows the AWS EC2 Load Balancers page. On the left, there's a sidebar with navigation links for EC2, Dashboard, Events, Instances, Images, Elastic Block Store, Network & Security, and more. The main area displays a table titled 'Load balancers (1/2)' with two entries: 'appl-lb' and 'alb-1'. The 'alb-1' row is selected and highlighted with yellow. Below the table, under the heading 'Load balancer: alb-1', there are two sections: 'Amazon CloudFront + AWS Web Application Firewall (WAF)' and 'AWS Global Accelerator'. Both sections have a 'Manage CloudFront + WAF integration' button.

And select your SSL certificate:

The screenshot shows the 'Manage integration' page for the selected load balancer 'alb-1'. The left sidebar is identical to the previous screenshot. The main content area has several sections: 'Amazon CloudFront + AWS Web Application Firewall (WAF)', 'CloudFront distributions', and 'Add distribution'. Under 'CloudFront distributions', there's a note about associating up to 1 TLS certificate per CloudFront distribution. A 'SSL/TLS certificate - optional' dropdown menu is open, showing 'Select a us-east-1 certificate' with a dropdown arrow. There are also 'Add distribution' and 'Creation workflow and status' sections.

And follow the steps what has shown then your certificate will get added and DNS will be created in CloudFront:

The screenshot shows the AWS EC2 Load Balancers console. On the left, there's a navigation sidebar with options like Dashboard, EC2 Global View, Events, Instances, Images, Elastic Block Store, Network & Security, and more. The main area displays 'Load balancers (1/2)' with two entries: 'appl-lb' (Active, application, Internet-facing, IPv4, VPC ID: vpc-038d9ffa8289cfac, 2 Availability Zones: sg-0589a6588b73b7f6, sg-03bf157819a5343b, DNS name: appl-lb-1809938261.us-east-1.elb.amazonaws.com) and 'alb-1' (Active, application, Internet-facing, IPv4, VPC ID: vpc-038d9ffa8289cfac, 2 Availability Zones: sg-03bf157819a5343b, sg-049239446.us-east-1.elb.amazonaws.com). Below this, the 'Load balancer: appl-lb' details page is shown, featuring tabs for Details, Listeners and rules, Network mapping, Resource map, Security, Monitoring, Integrations (selected), Attributes, Capacity, and Tags. The Integrations section lists 'Amazon Application Recovery Controller (ARC)' (Not integrated) and 'Amazon CloudFront + AWS Web Application Firewall (WAF)' (Integrated). The WAF section shows CloudFront distribution E38APR0LJS4HSQ0 is Enabled and WAF enabled.

-----done-----

## 5. Map Application load balancer to R53.

The screenshot shows the AWS Route 53 Create record page for the domain 'mohdshuja.com'. The 'Create record' section is open, showing a 'Quick create record' form. The 'Record name' field contains 'subdomain' and the 'Record type' field is set to 'A – Routes traffic to an IPv4 address and some AWS resources'. Under 'Route traffic to', 'Alias' is selected, and the target is 'US East (N. Virginia)' with the alias host ID Z355XKDOTRQ7X7K. The 'Routing policy' is set to 'Simple routing'. The 'Evaluate target health' option is selected. At the bottom right, there are 'Cancel' and 'Create records' buttons, along with a link to 'View existing records'.

**Record details**

- Edit record**
- Record name**: mohdshuja.com
- Record type**: A
- Value**: dbqbttxwivjs4.cloudfront.net
- Alias**: Yes
- TTL (seconds)**: -
- Routing policy**: Simple

6. Push the application load balancer logs to S3.
  7. Select already created bucket
  8. Set Bucket Policy for ALB Logging
- Go to the bucket → Permissions → Bucket Policy → Add

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AWSALBLoggingPermissions",
      "Effect": "Allow",
      "Principal": [
        "logdelivery.elasticloadbalancing.amazonaws.com"
      ],
      "Action": "S3:PutObject",
      "Resource": "arn:aws:s3:::elb-application-flowlogs-01/AWSLogs/085794372055/*"
    }
  ]
}
```

**Copy**

**CloudShell** **Feedback** © 2025, Amazon Web Services, Inc. or its affiliates. **Privacy** **Terms** **Cookie preferences**

26°C Mostly cloudy **Search** ENG IN 19:19 07-10-2025

The screenshot shows the AWS EC2 Load Balancers console. The user is on the 'Edit load balancer attributes' page for a specific load balancer. In the 'Access logs' section, there is a checked checkbox for 'Access logs' which enables detailed logs of all requests made to the Elastic Load Balancer. The logs are stored in an S3 bucket named 'elb-application-flowlogs-01'. The 'Save changes' button at the bottom right is highlighted.

## 9. Go to EC2 Dashboard → Load Balancers → Select your ALB

1. Description tab → “Edit attributes”
2. Enable Access logs

The screenshot shows the AWS S3 Buckets console. The user is viewing the 'elb-application-flowlogs-01' bucket. The 'Objects' tab is selected, showing a single folder named 'AWSLogs/'. The 'Upload' button at the top right of the object list is highlighted.

-----completed-----