TASK – 2

Step 1: Create a Target Group for Jenkins

1. Navigate to Target Groups

- Open the **AWS EC2 Console**.
- In the left sidebar, under **Load Balancing**, select **Target Groups**.
- Click **Create Target Group**.

2. Configure Basic Settings

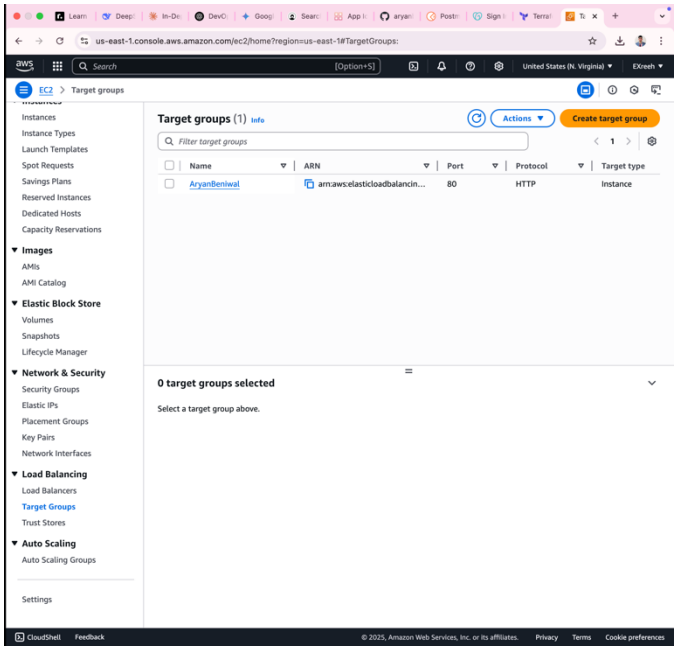
- **Target Type**: `Instances` (default).
- **Protocol**: `HTTP`.
- **Port**: `8080` (Jenkins default port).
- **VPC**: Select the **same VPC** as your EC2 instance.

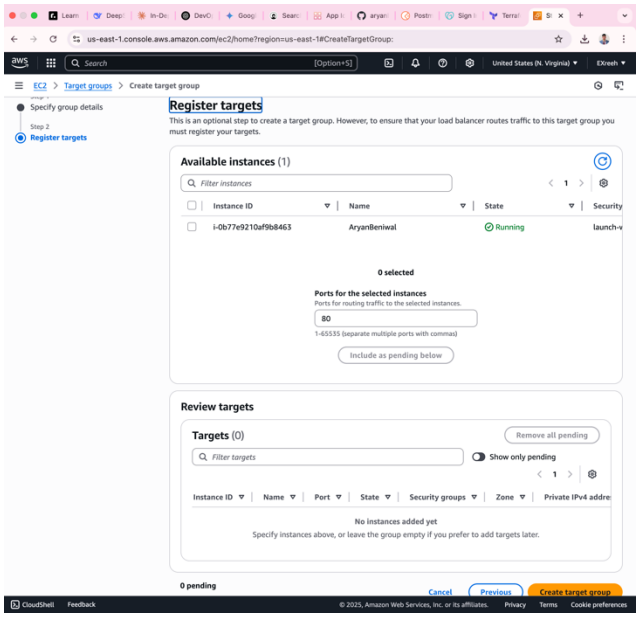
3. Name & Health Checks

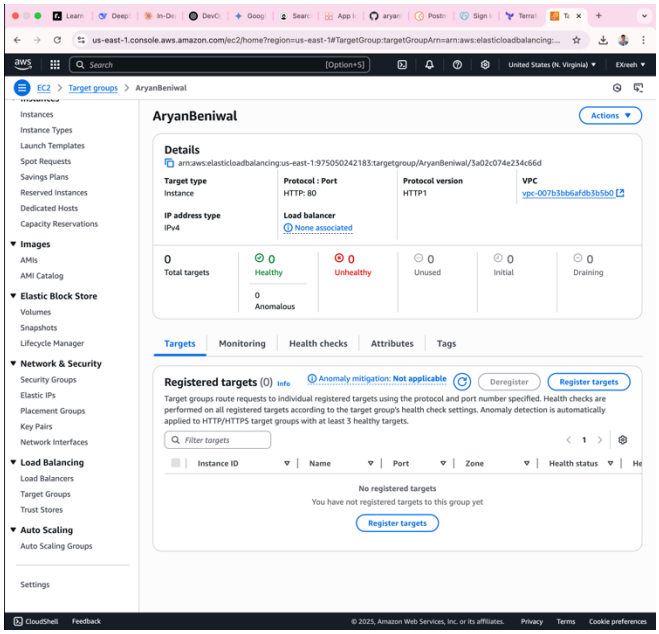
- **Name**: Enter a descriptive name (e.g., `jenkins-target-group`).
- **Health Check Path**: `/` (root path, or use `/login` for Jenkins).
- (Optional) Adjust advanced health check settings (e.g., interval, thresholds).

4. Register EC2 Instance

- Under **Registered instances**, select your **Jenkins EC2 instance**.
- Click **Include as pending below → Create Target Group**.


us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#TargetGroups:
EC2 > Target groups
Target groups (1) Info
Actions
Create target group
Filter target groups
Name
ARN
Port
Protocol
Target type
AryanBeniwal
arn:aws:elasticloadbalancin...
80
HTTP
Instance
0 target groups selected
Select a target group above.
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores
Auto Scaling
Auto Scaling Groups
Settings
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateTargetGroup:
EC2 > Target groups > Create target group
Specify group details
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)
Filter instances
Instance ID
Name
State
Security
i-0b77e9210af9b8463
AryanBeniwal
Running
0 selected
Ports for the selected instances
Ports for routing traffic to the selected instances.
80
1-65535 (separate multiple ports with comma)
Include as pending below
Review targets
Targets (0)
Remove all pending
Filter targets
Show only pending
Instance ID
Name
Port
State
Security groups
Zone
Private IPv4 addre
No instances added yet
Specify instances above, or leave the group empty if you prefer to add targets later.
0 pending
Cancel
Previous
Create target group
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences



Step 2: Create an Application Load Balancer for Jenkins

1. Navigate to Load Balancers

- Go to **AWS EC2 Console**
- Under **Load Balancing**, select **Load Balancers**
- Click **Create Load Balancer**

2. Select Load Balancer Type

- Choose **Application Load Balancer (ALB) → Create**

3. Configure Basic Settings

- **Name**: AryanBeniwal(or your preferred name)
- **Scheme**: Internet-facing (publicly accessible)
- **IP address type**: IPv4

4. Network Mapping

- **VPC**: Select the **same VPC** as your instances
- **Availability Zones**:
    - Select **at least 2 subnets** in different AZs (e.g., `us-east-1a` and `us-east-1b`)
    - *Note: This ensures high availability*

5. Security Groups

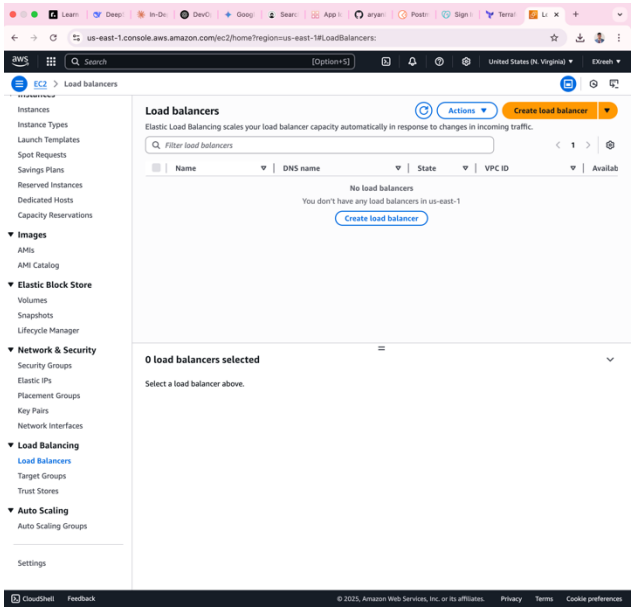
- Assign a security group that **allows HTTP (Port 80)** traffic
- *Recommended*: Create a new SG or modify existing to allow:
    - **Inbound**: HTTP (Port 80) from `0.0.0.0/0` (or restrict to your IP)
    - **Outbound**: All traffic (default)

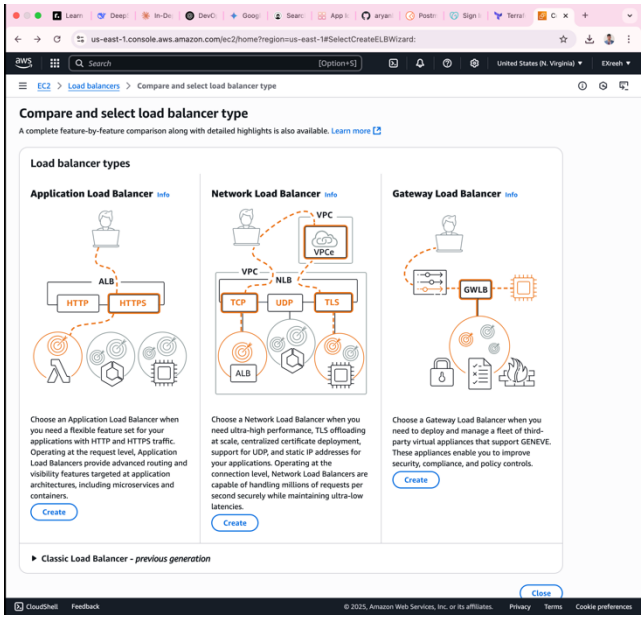
6. Listeners and Routing

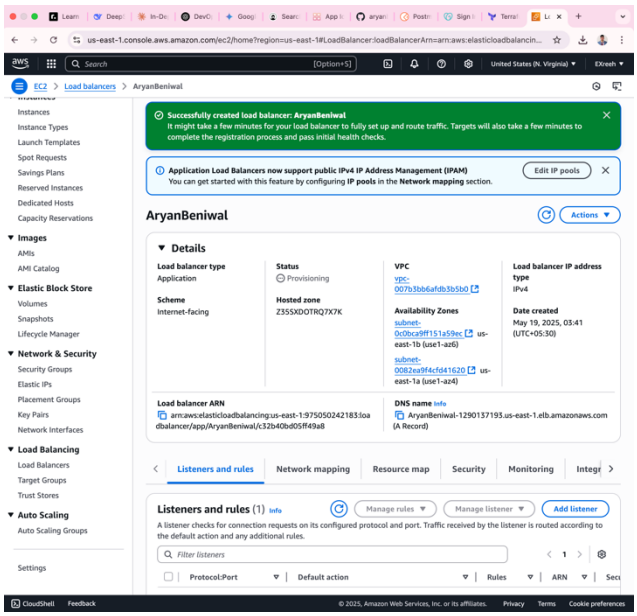
- **Listener**: HTTP on **Port 80**
- **Default action**: Forward to your **Jenkins Target Group** (created earlier)

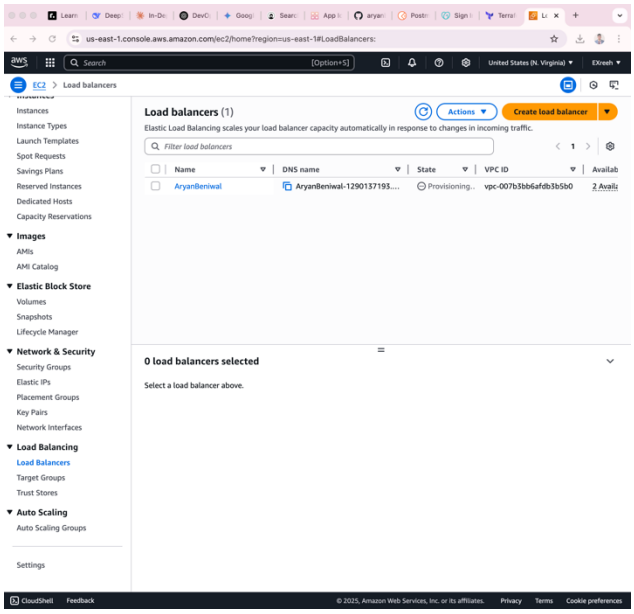
7. Review and Create

- Verify all settings
- Click **Create Load Balancer**




us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#SelectCreateELBWizard:
EC2 > Load balancers > Compare and select load balancer type
Compare and select load balancer type
A complete feature-by-feature comparison along with detailed highlights is also available. Learn more
Load balancer types
Application Load Balancer Info
ALB
HTTP
HTTPS
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.
Create
Network Load Balancer Info
VPC
VPCe
VPC
NLB
TCP
UDP
TLS
ALB
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.
Create
Gateway Load Balancer Info
GWLB
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.
Create
Classic Load Balancer - previous generation
Close
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LoadBalancer:loadBalancerArn=arn:aws:elasticloadbalancin...
EC2 > Load balancers > AryanBeniwal
Successfully created load balancer: AryanBeniwal
It might take a few minutes for your load balancer to fully set up and route traffic. Targets will also take a few minutes to complete the registration process and pass initial health checks.
Application Load Balancers now support public IPv4 IP Address Management (IPAM)
You can get started with this feature by configuring IP pools in the Network mapping section.
Edit IP pools
AryanBeniwal
Actions
Details
Load balancer type
Application
Status
Provisioning
VPC
vpc-007b3bb6afdb3b5b0
Load balancer IP address type
IPv4
Scheme
Internet-facing
Hosted zone
Z35SXDOTRQ7X7K
Availability Zones
subnet-0c0bca9ff151a59ec us-east-1b (use1-az6)
subnet-00b2ea9f4cfd41620 us-east-1a (use1-az4)
Date created
May 19, 2025, 03:41 (UTC+05:30)
Load balancer ARN
arn:aws:elasticloadbalancing:us-east-1:975050242183:loadbalancer/app/AryanBeniwal/c32b40bd05ff49a8
DNS name Info
AryanBeniwal-1290137193.us-east-1.elb.amazonaws.com (A Record)
Listeners and rules
Network mapping
Resource map
Security
Monitoring
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.
Manage rules
Manage listener
Add listener
Filter listeners
Protocol:Port
Default action
Rules
ARN
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores
Auto Scaling
Auto Scaling Groups
Settings
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences



Step 3: Configure Path-Based Routing Rules for ALB

1. Navigate to Load Balancer Listeners

- Go to **AWS EC2 Console**
- Under **Load Balancing**, select **Load Balancers**
- Choose your ALB (e.g., AryanBeniwal)
- Select the **Listeners** tab → Click on **View/Edit Rules** for HTTP:80

2. Modify Default Rule (Optional)

- **Option 1**: Delete the default forward rule to enforce strict path matching.
- **Option 2**: Keep it but set a fallback action (e.g., return fixed response for unmatched paths).

3. Add Path-Based Routing Rule

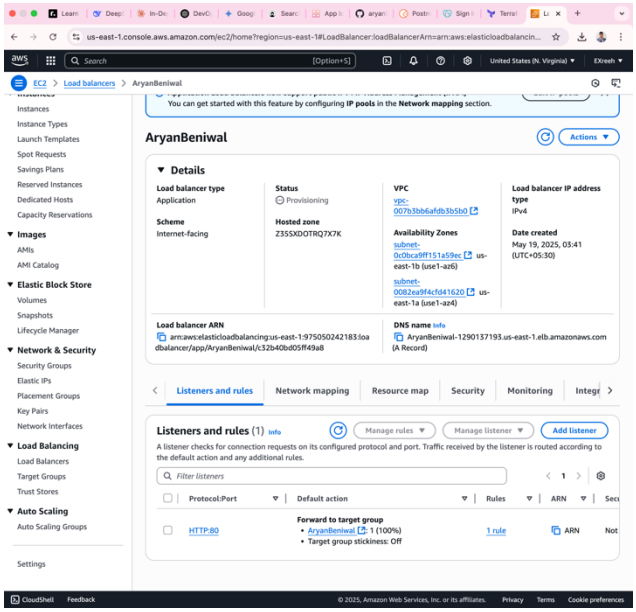
- Click **Add Rule → Insert Rule**
- **Condition**:

- Select **Path is** → Enter `/jenkins*`
  (Matches `/jenkins`, `/jenkins/`, and any subpaths)

- **Action**:
  - Select **Forward to** → Choose your Jenkins target group (e.g., `jenkins-tg`)
- **Priority**: Rules are evaluated top-down. Ensure this rule has higher priority than catch-all rules.

4. Save Rules

- Click **Save** to apply changes.



Step 4: Create an Alias A Record in Route 53

1. Access Route 53 Hosted Zone

- Go to **AWS Route 53 Console**
- Select **Hosted Zones** → Choose your domain (e.g., `ecliplearn.in`)

2. Create Record

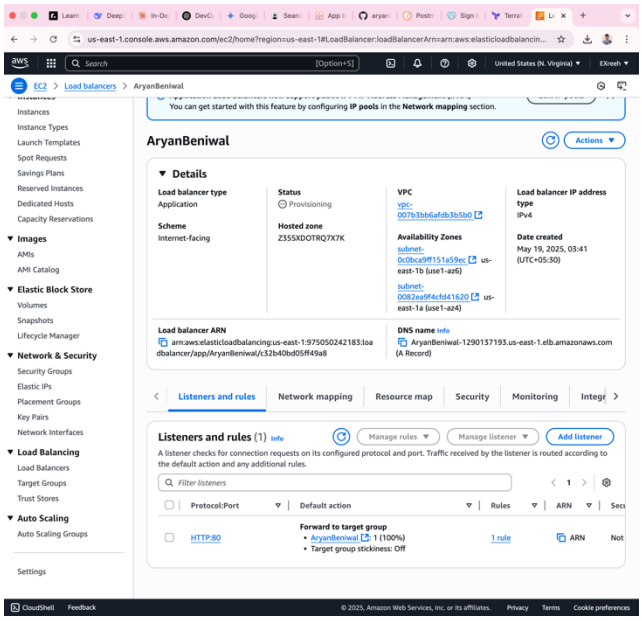
- Click **Create Record**
- **Record Name**: Enter subdomain (e.g., `www` for `www.ecliplearn.in`)
- **Record Type**: Select **A – IPv4 address**
- **Alias**: Toggle **ON**

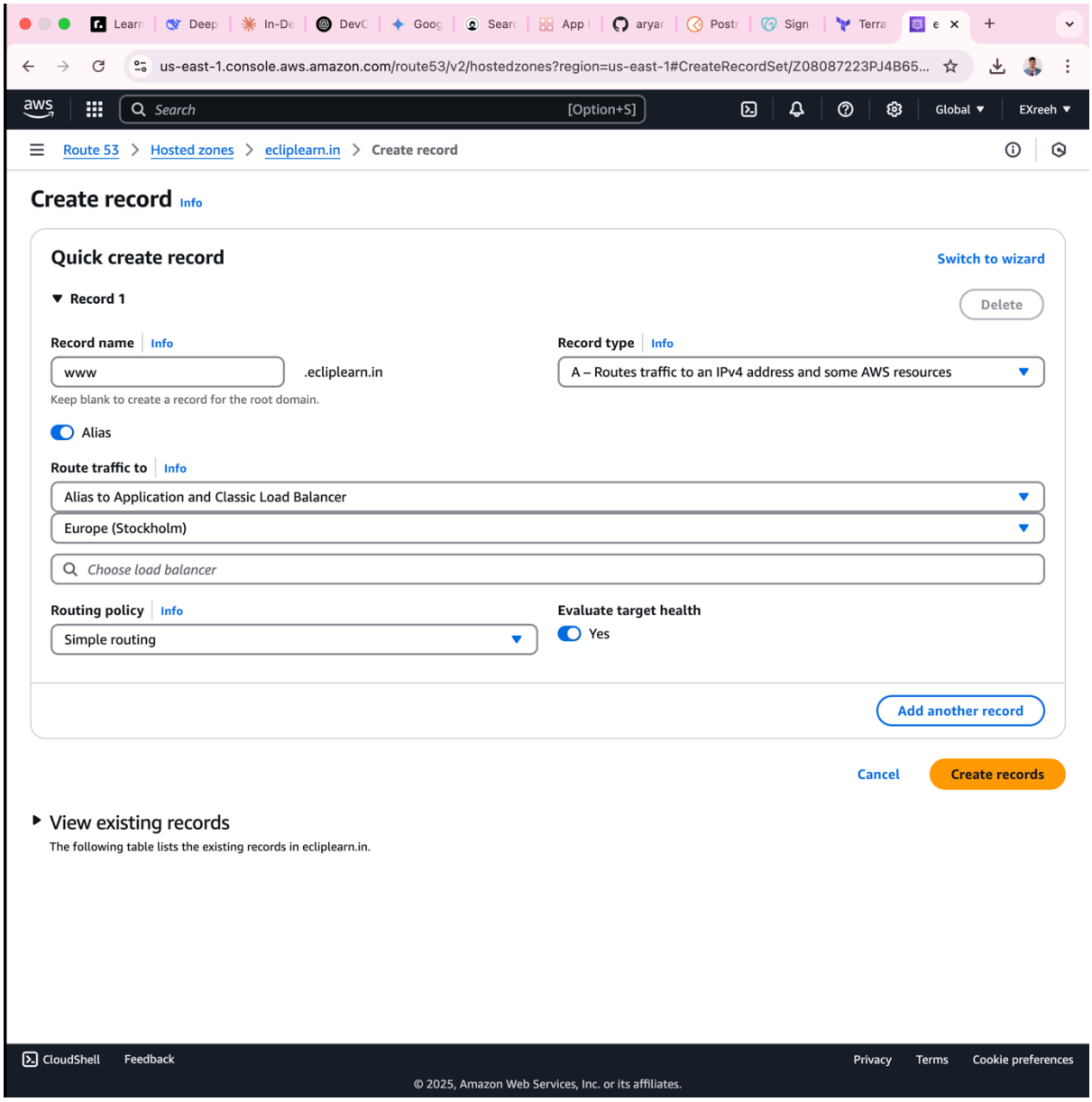
3. Configure Alias Target

- **Route traffic to**:
  - Choose **Alias to Application and Classic Load Balancer**
  - Select your **AWS Region** (e.g., `us-east-1`)
  - Pick your ALB from the dropdown (e.g., `8-SEM-Workshop`)
- **Routing Policy**: `Simple` (default)

4. Save Changes

- Click **Create Records**





Step 5: Request an SSL/TLS Certificate from AWS ACM

1. Navigate to AWS Certificate Manager

- Go to **AWS ACM Console**
- Ensure you're in the **correct region** (same as your ALB)

- Click **Request a certificate**

2. Select Certificate Type

- Choose **Request a public certificate** → Click **Next**

3. Specify Domain Names

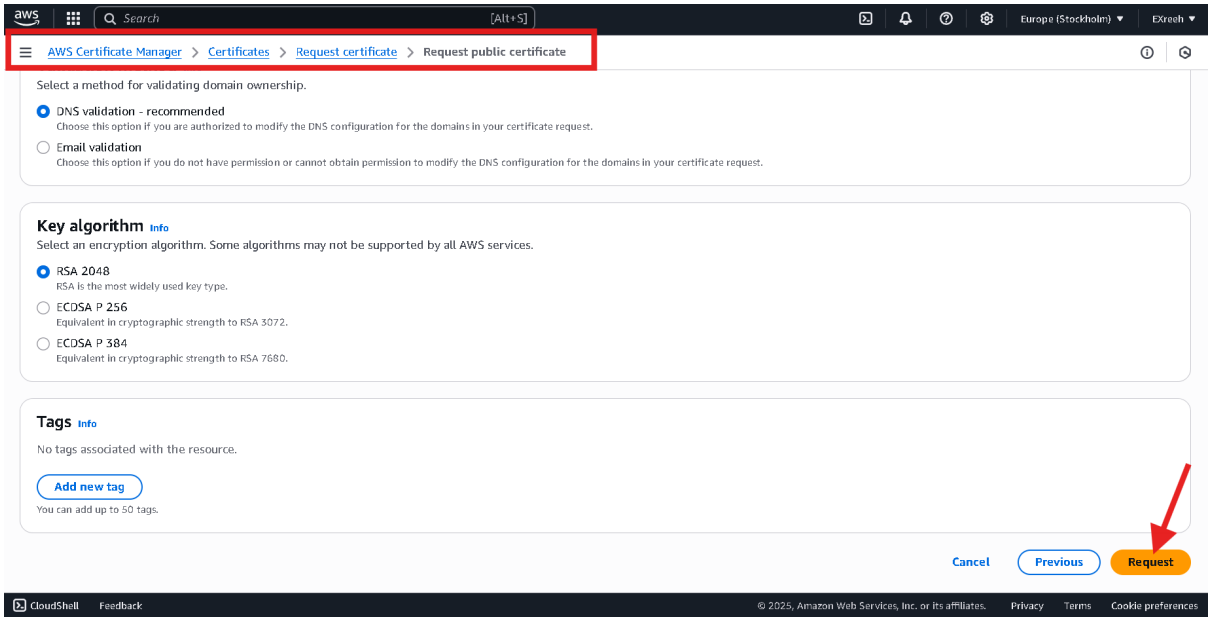
- **Fully Qualified Domain Names (FQDNs)**:
  - Primary domain: `ecliplearn.in`
  - Additional names (recommended):
    - `www.ecliplearn.in` (for www prefix)
    - `*.ecliplearn.in` (if needing wildcard for subdomains)
- Click **Next**

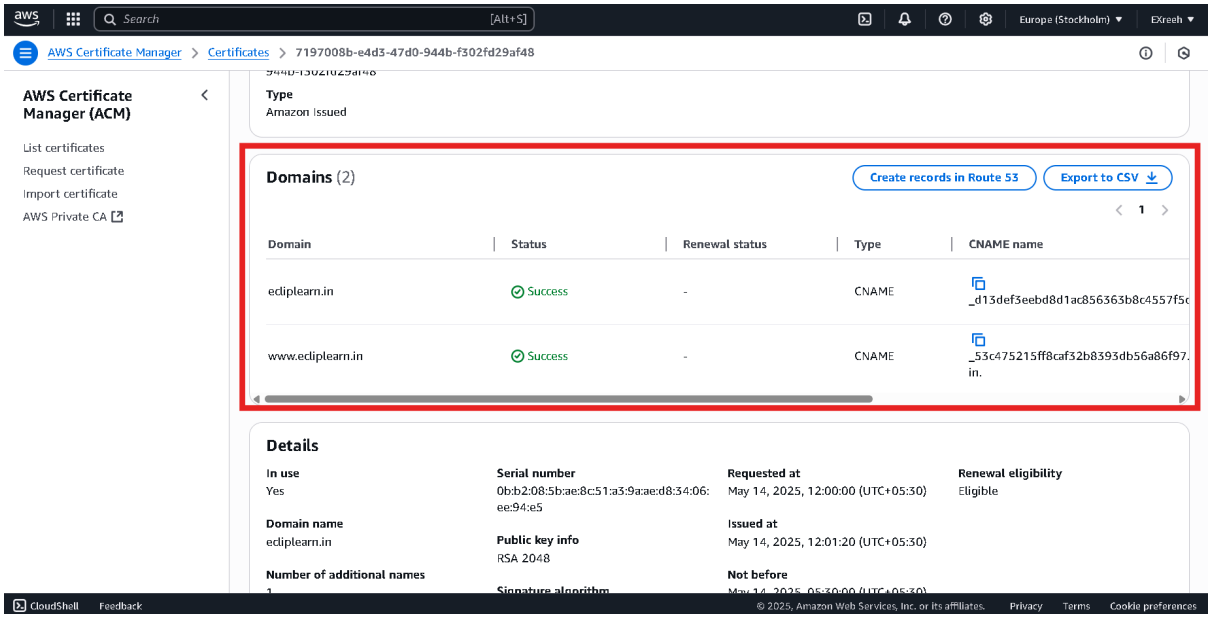
4. Choose Validation Method

- **DNS validation** (Recommended):
  - Automatically creates Route 53 CNAME records for verification
  - Faster than email validation
- Click **Next**

5. Review and Request

- Verify domain names and validation method
- Click **Confirm and request**





Step 6: Validate the SSL Certificate Using DNS (Route 53)

1. Locate the CNAME Record in ACM

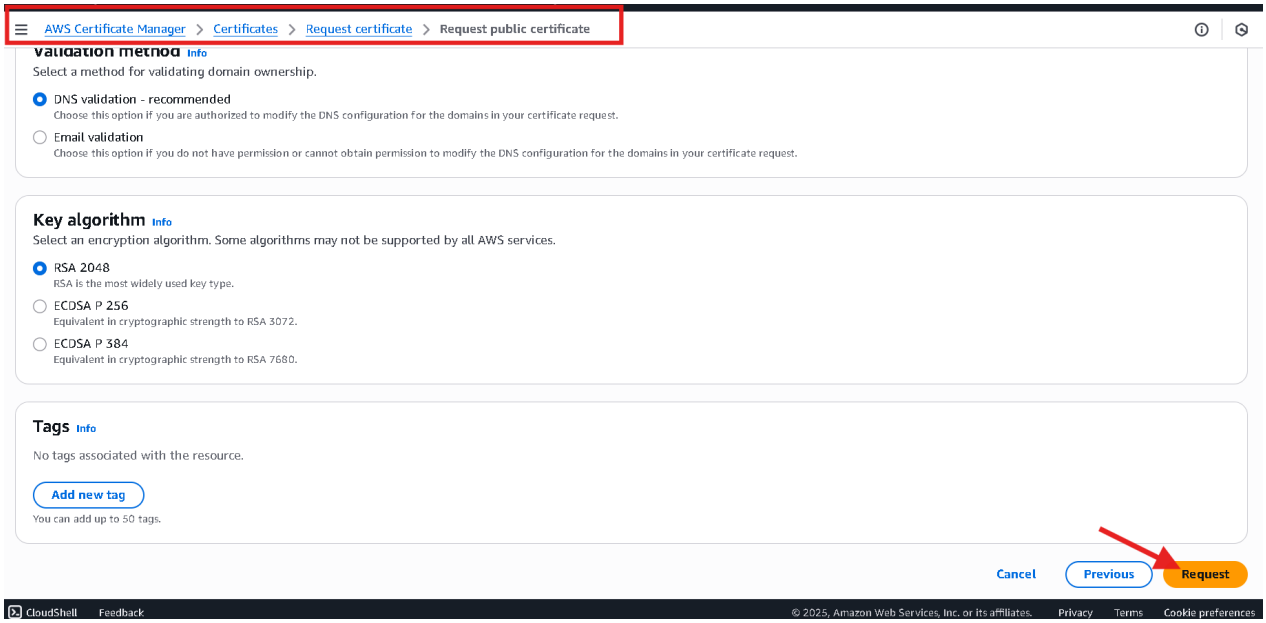
- After requesting the certificate, go to **AWS ACM Console**.
- Under **Certificates**, select your pending certificate.
- Copy the **CNAME name** and **CNAME value** provided under **Domain validation options**.

2. Add the CNAME Record in Route 53

- Go to **Route 53 Hosted Zones**.
- Select your domain's hosted zone (e.g., `ecliplearn.in`).
- Click **Create Record**.
- Configure the record:
  - **Record Type**: `CNAME`
  - **Name**: Paste the **CNAME name** from ACM (e.g., `_abcdef1234567890.ecliplearn.in`).
  - **Value**: Paste the **CNAME value** from ACM (e.g., `_xyz0987654321.acm-validations.aws.`).
  - **TTL**: Keep default (or set to 300 seconds).
- Click **Create Records**.

3. Wait for Validation (≈5-30 min)

- ACM automatically checks DNS validation.
- Certificate status changes from **Pending Validation** → **Issued** once verified.



Step 7: Configure HTTPS Listener (Port 443) for ALB

1. Navigate to Your Load Balancer

- Go to **EC2 Load Balancers Console**.
- Select your **Application Load Balancer** (e.g., `8-SEM-Workshop`).

2. Add HTTPS Listener

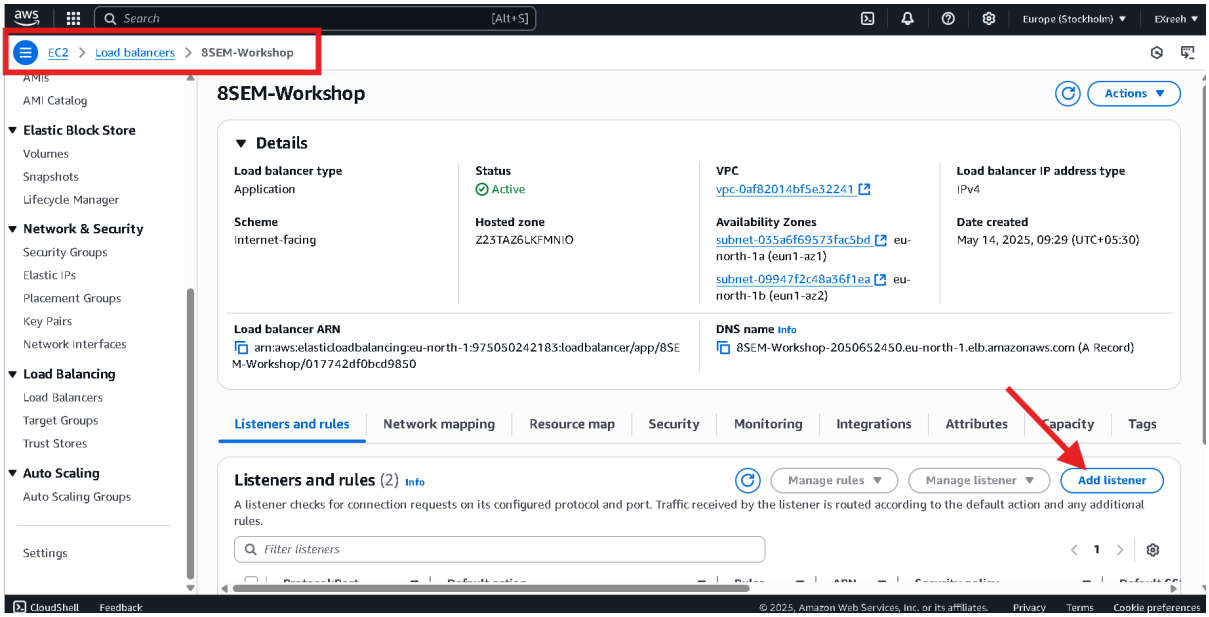
- Under the **Listeners** tab → Click **Add listener**.
- Configure:
    - **Protocol**: `HTTPS`
    - **Port**: `443`
    - **Default action**:
        - **Forward to** → Select your Jenkins **Target Group** (e.g., `jenkins-tg`).

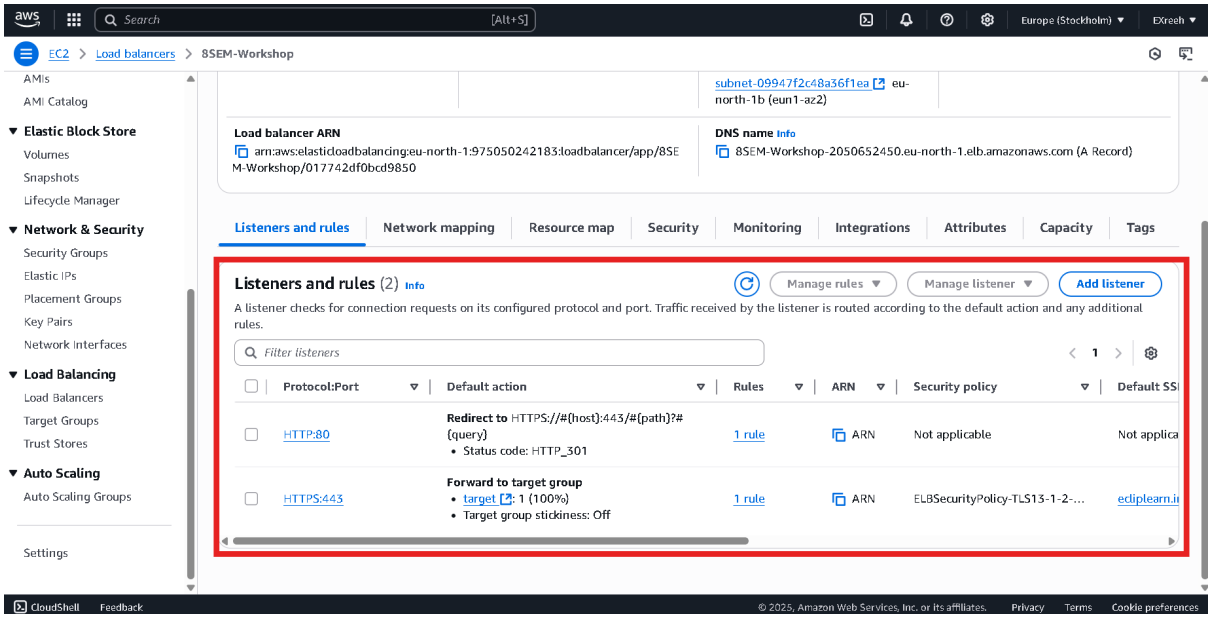
3. Attach SSL Certificate

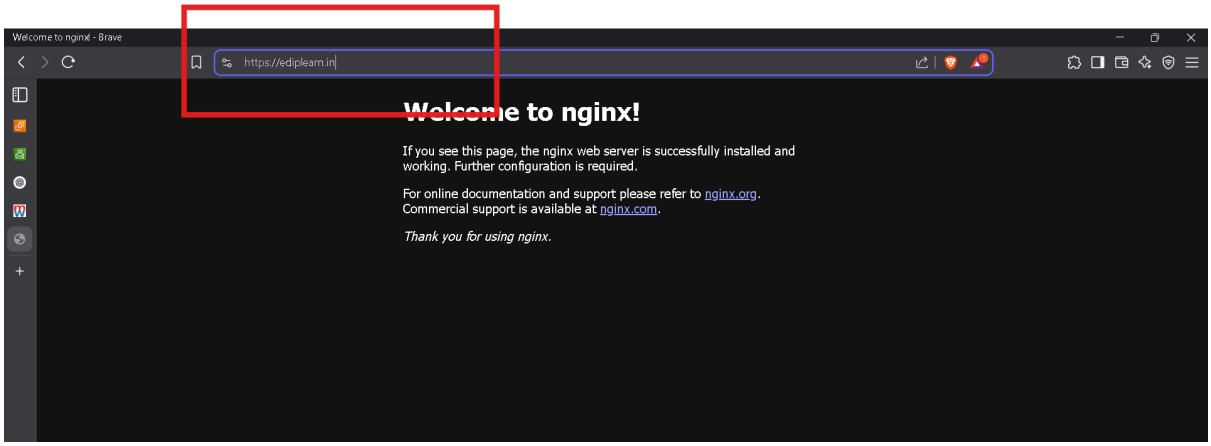
- Under **SSL Certificate**:
    - Select **From ACM** (AWS Certificate Manager).
    - Choose your issued certificate for `ecliplearn.in`.

4. Save Changes

- Click **Save** to apply the HTTPS listener.


EC2 > Load balancers > 8SEM-Workshop
8SEM-Workshop
Actions
Details
Load balancer type
Application
Status
Active
VPC
vpc-0af82014bf5e32241
Load balancer IP address type
IPv4
Scheme
Internet-facing
Hosted zone
Z23TAZ6LKFMNIO
Availability Zones
subnet-035a6f69573fac5bd eu-north-1a (eun1-az1)
subnet-09947f2c48a36f1ea eu-north-1b (eun1-az2)
Date created
May 14, 2025, 09:29 (UTC+05:30)
Load balancer ARN
arn:aws:elasticloadbalancing:eu-north-1:975050242183:loadbalancer/app/8SEM-Workshop/017742df0bcd9850
DNS name Info
8SEM-Workshop-2050652450.eu-north-1.elb.amazonaws.com (A Record)
Listeners and rules
Network mapping
Resource map
Security
Monitoring
Integrations
Attributes
Capacity
Tags
Listeners and rules (2) Info
Manage rules
Manage listener
Add listener
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.
Filter listeners


EC2 > Load balancers > 8SEM-Workshop
subnet-09947f2c48a36f1ea eu-north-1b (eun1-az2)
Load balancer ARN
arn:aws:elasticloadbalancing:eu-north-1:975050242183:loadbalancer/app/8SEM-Workshop/017742df0bcd9850
DNS name Info
8SEM-Workshop-2050652450.eu-north-1.elb.amazonaws.com (A Record)
Listeners and rules
Network mapping
Resource map
Security
Monitoring
Integrations
Attributes
Capacity
Tags
Listeners and rules (2) Info
Manage rules
Manage listener
Add listener
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.
Filter listeners
Protocol:Port
Default action
Rules
ARN
Security policy
HTTP:80
Redirect to HTTPS://#{host}:443/#{path}?#{query}
Status code: HTTP_301
1 rule
ARN
Not applicable
HTTPS:443
Forward to target group
target: 1 (100%)
Target group stickiness: Off
1 rule
ARN
ELBSecurityPolicy-TLS13-1-2-...


Welcome to nginx! - Brave
https://ediplearn.in
Welcome to nginx!
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.
For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.
Thank you for using nginx.