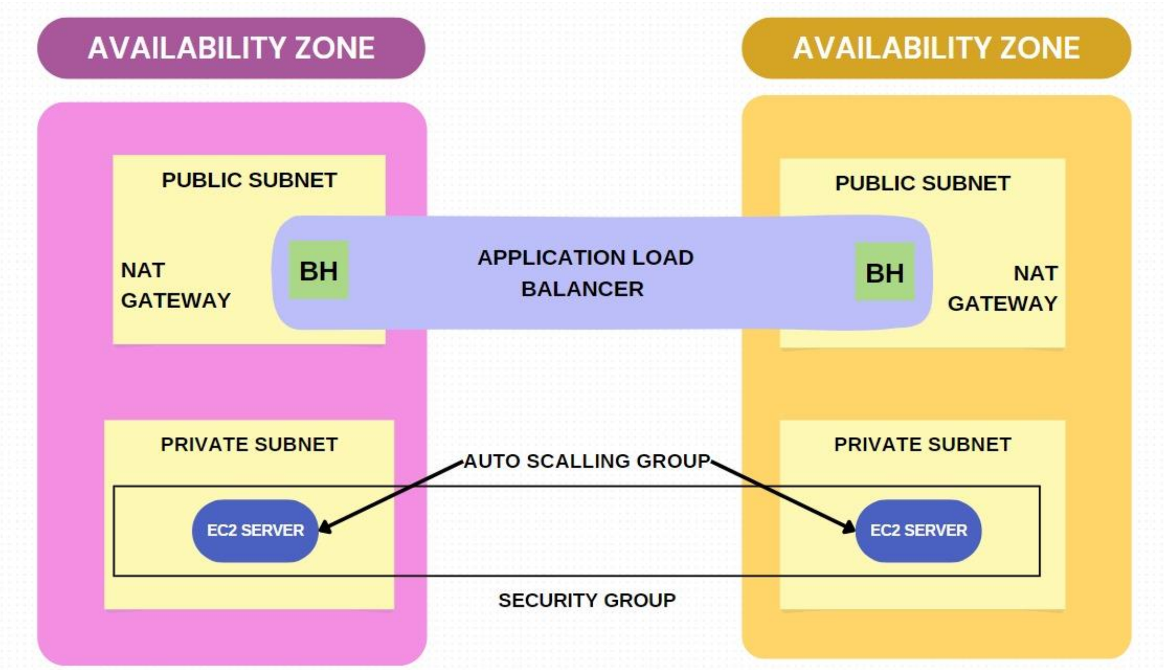
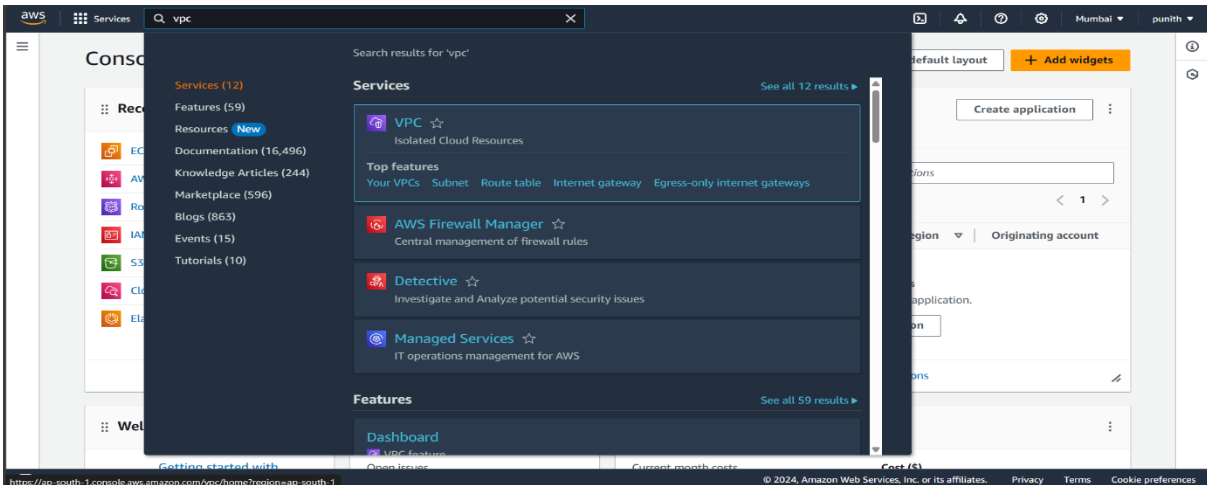
Scalable and Secure Web Hosting in AWS: Leveraging VPCs, Auto Scaling, and Load Balancing for High Availability

This AWS project explains establishing a robust infrastructure utilizing VPCs, Auto Scaling Groups, and Load Balancers for dynamic scaling and efficient traffic distribution through private subnets and a bastion host with secure access to instances while maintaining availability and reliability of the hosted websites.



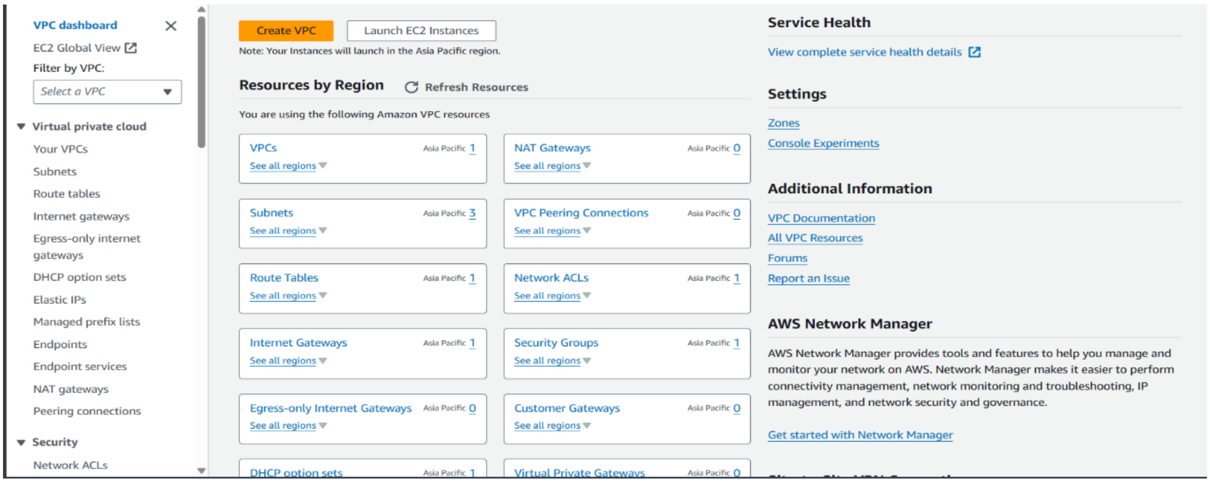
All the steps have been explained in detailed way with images attached below:

Login into AWS console. Search for VPC



https://ap-south-1.console.aws.amazon.com/vpc/home?region=ap-south-1

Choose Create VPC.



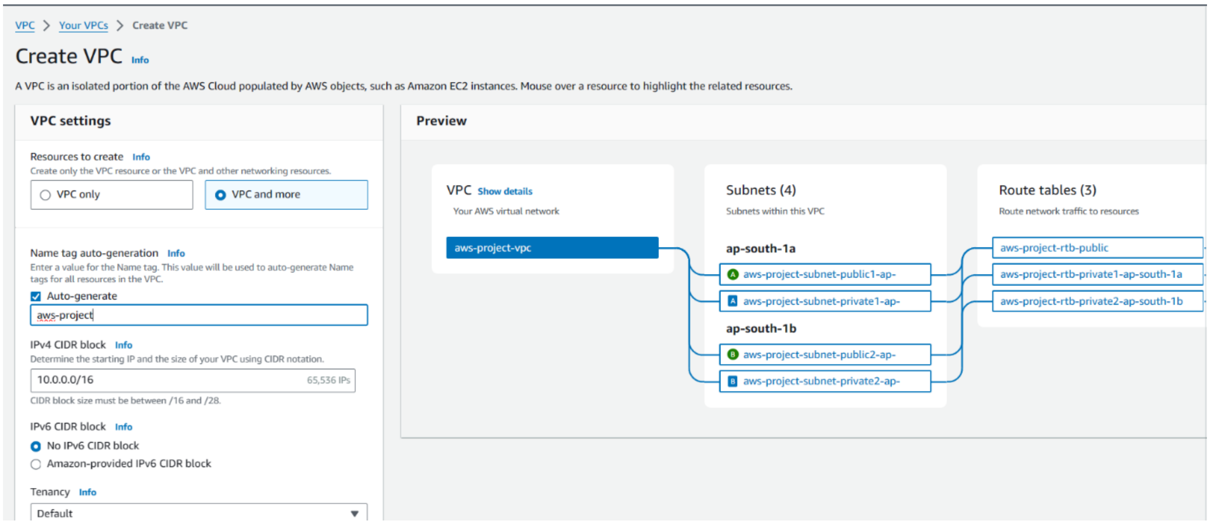
Note: Your Instances will launch in the Asia Pacific region.

You are using the following Amazon VPC resources

VPCs Asia Pacific 1 See all regions	NAT Gateways Asia Pacific 0 See all regions
Subnets Asia Pacific 3 See all regions	VPC Peering Connections Asia Pacific 0 See all regions
Route Tables Asia Pacific 1 See all regions	Network ACLs Asia Pacific 1 See all regions
Internet Gateways Asia Pacific 1 See all regions	Security Groups Asia Pacific 1 See all regions
Egress-only Internet Gateways Asia Pacific 0 See all regions	Customer Gateways Asia Pacific 0 See all regions
DHCP option sets Asia Pacific 1	Virtual Private Gateways Asia Pacific 0

AWS Network Manager provides tools and features to help you manage and monitor your network on AWS. Network Manager makes it easier to perform connectivity management, network monitoring and troubleshooting, IP management, and network security and governance.

Name VPC and side-by-side we can view the subnets and the route table that is going to be created in the VPC.


VPC
Your VPCs
Create VPC
Create VPC Info
VPC settings
Preview
VPC Show details
Your AWS virtual network
aws-project-vpc
Subnets (4)
Subnets within this VPC
ap-south-1a
aws-project-subnet-public1-ap-
aws-project-subnet-private1-ap-
ap-south-1b
aws-project-subnet-public2-ap-
aws-project-subnet-private2-ap-
Route tables (3)
Route network traffic to resources
aws-project-rtb-public
aws-project-rtb-private1-ap-south-1a
aws-project-rtb-private2-ap-south-1b

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances. Mouse over a resource to highlight the related resources.

Resources to create Info
Create only the VPC resource or the VPC and other networking resources.
○ VPC only ● VPC and more

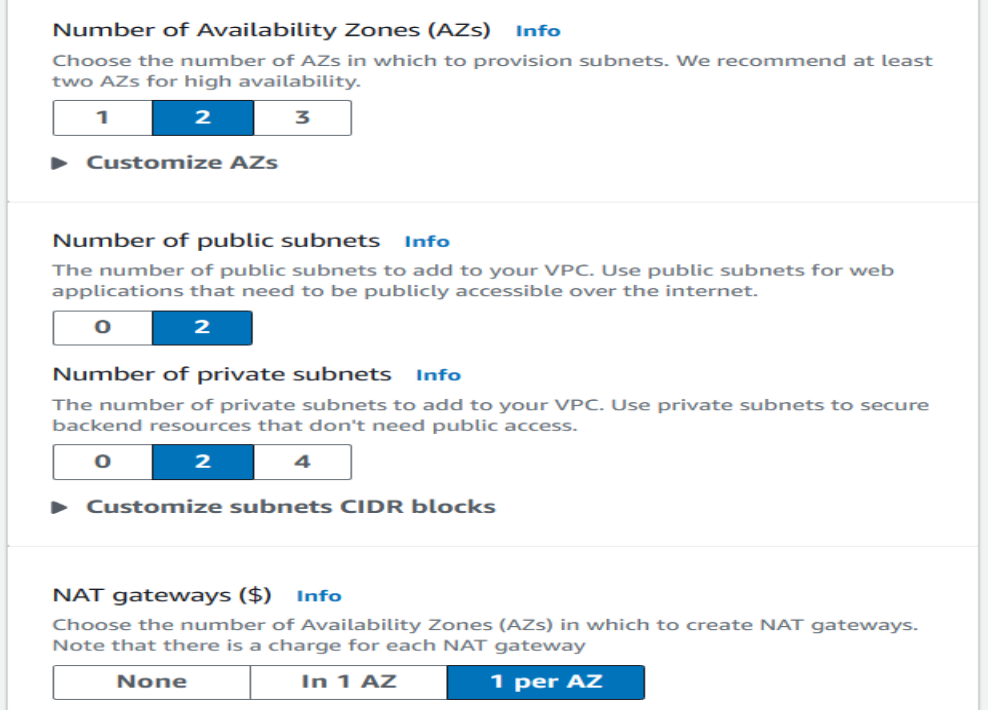
Name tag auto-generation Info
Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.
☑ Auto-generate
aws-project

IPv4 CIDR block Info
Determine the starting IP and the size of your VPC using CIDR notation.
10.0.0.0/16 65,536 IPs
CIDR block size must be between /16 and /28.

IPv6 CIDR block Info
● No IPv6 CIDR block
○ Amazon-provided IPv6 CIDR block

Tenancy Info
Default

Select no.of Availability Zones, Public, Private Subnets and Make sure to create one NAT gateway per Availability Zone.



Number of Availability Zones (AZs) Info

Choose the number of AZs in which to provision subnets. We recommend at least two AZs for high availability.

1 [✓ 2] 3

▶ Customize AZs

Number of public subnets Info

The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the internet.

0 [✓ 2]

Number of private subnets Info

The number of private subnets to add to your VPC. Use private subnets to secure backend resources that don't need public access.

0 [✓ 2] 4

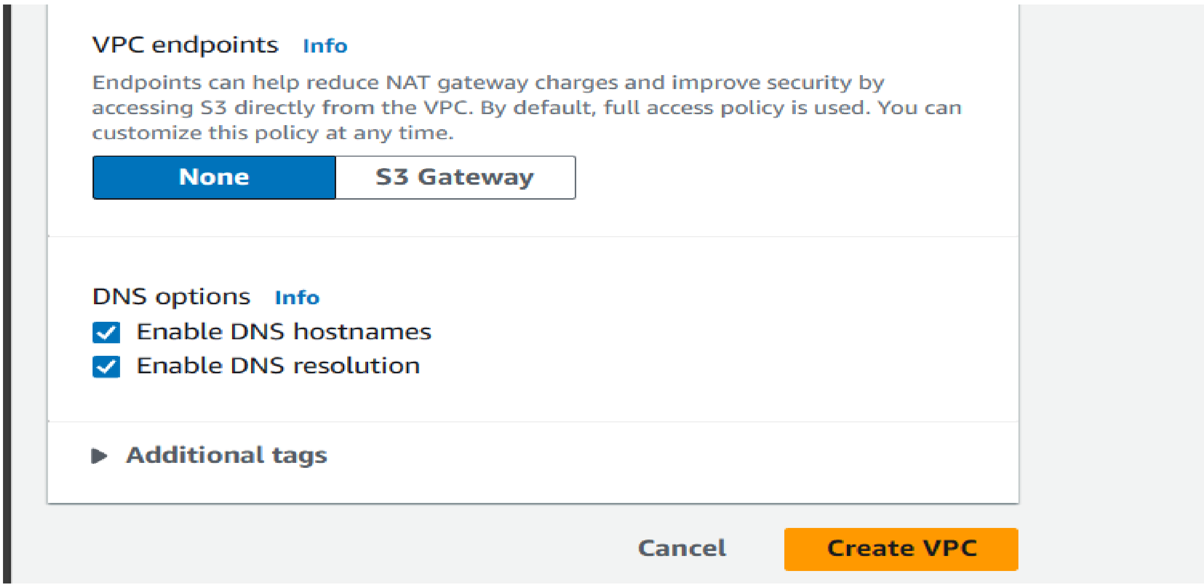
▶ Customize subnets CIDR blocks

NAT gateways ($) Info

Choose the number of Availability Zones (AZs) in which to create NAT gateways. Note that there is a charge for each NAT gateway

None In 1 AZ [✓ 1 per AZ]

No need of s3 Gateway, so choose None and Create VPC.



VPC endpoints Info

Endpoints can help reduce NAT gateway charges and improve security by accessing S3 directly from the VPC. By default, full access policy is used. You can customize this policy at any time.

[✓ None] S3 Gateway

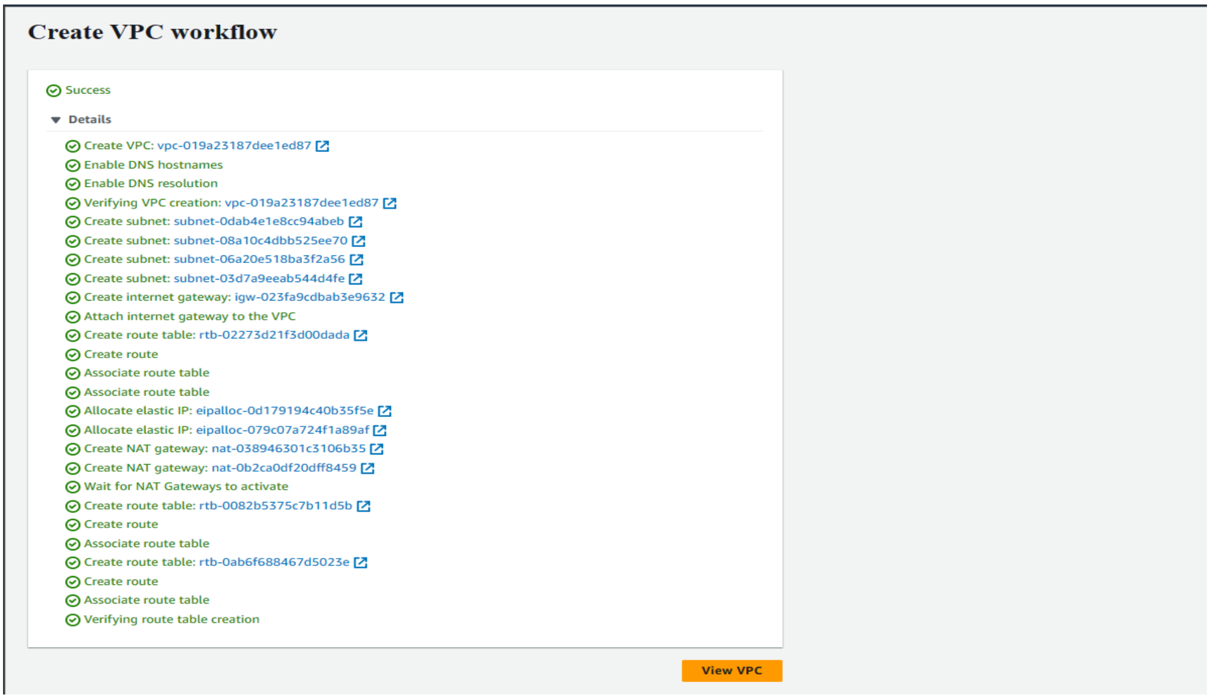
DNS options Info

☑ Enable DNS hostnames

☑ Enable DNS resolution

▶ Additional tags

One-by-one, the components of VPC gets created. Choose view VPC.



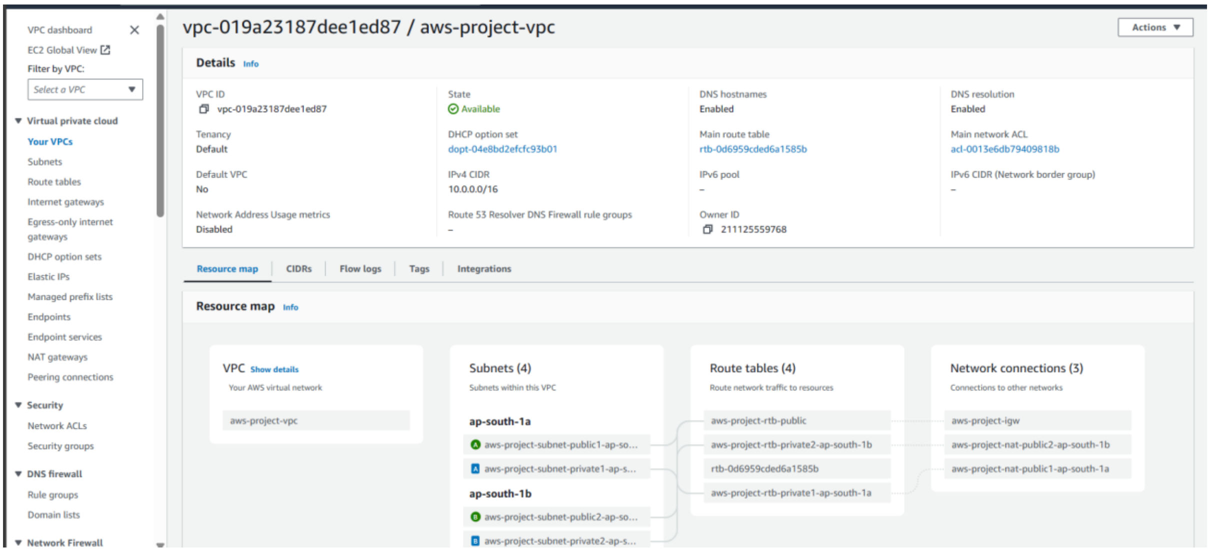
Create VPC workflow

✓ Success

▼ Details

- ✓ Create VPC: vpc-019a23187dee1ed87
- ✓ Enable DNS hostnames
- ✓ Enable DNS resolution
- ✓ Verifying VPC creation: vpc-019a23187dee1ed87
- ✓ Create subnet: subnet-0dab4e1e8cc94abeb
- ✓ Create subnet: subnet-08a10c4dbb525ee70
- ✓ Create subnet: subnet-06a20e518ba3f2a56
- ✓ Create subnet: subnet-03d7a9eeab544d4fe
- ✓ Create internet gateway: igw-023fa9cdbab3e9632
- ✓ Attach internet gateway to the VPC
- ✓ Create route table: rtb-02273d21f3d00dada
- ✓ Create route
- ✓ Associate route table
- ✓ Associate route table
- ✓ Allocate elastic IP: eipalloc-0d179194c40b35f5e
- ✓ Allocate elastic IP: eipalloc-079c07a724f1a89af
- ✓ Create NAT gateway: nat-038946301c3106b35
- ✓ Create NAT gateway: nat-0b2ca0df20dff8459
- ✓ Wait for NAT Gateways to activate
- ✓ Create route table: rtb-0082b5375c7b11d5b
- ✓ Create route
- ✓ Associate route table
- ✓ Create route table: rtb-0ab6f688467d5023e
- ✓ Create route
- ✓ Associate route table
- ✓ Verifying route table creation

The overview of the VPC.



vpc-019a23187dee1ed87 / aws-project-vpc

Details Info

VPC ID: vpc-019a23187dee1ed87

State: Available

DNS hostnames: Enabled

DNS resolution: Enabled

Tenancy: Default

DHCP option set: dopt-04e8bd2efcfc93b01

Main route table: rtb-0d6959cded6a1585b

Main network ACL: acl-0013e6db79409818b

Default VPC: No

IPv4 CIDR: 10.0.0.0/16

IPv6 pool: –

IPv6 CIDR (Network border group): –

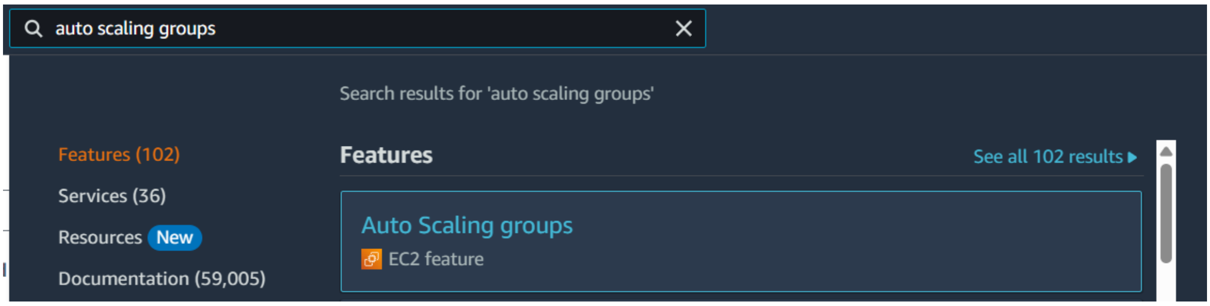
Network Address Usage metrics: Disabled

Route 53 Resolver DNS Firewall rule groups: –

Owner ID: 211125559768

Resource map
CIDRs
Flow logs
Tags
Integrations
Resource map Info
VPC Show details
Your AWS virtual network
aws-project-vpc
Subnets (4)
Subnets within this VPC
ap-south-1a
aws-project-subnet-public1-ap-so...
aws-project-subnet-private1-ap-s...
ap-south-1b
aws-project-subnet-public2-ap-so...
aws-project-subnet-private2-ap-s...
Route tables (4)
Route network traffic to resources
aws-project-rtb-public
aws-project-rtb-private2-ap-south-1b
rtb-0d6959cded6a1585b
aws-project-rtb-private1-ap-south-1a
Network connections (3)
Connections to other networks
aws-project-igw
aws-project-nat-public2-ap-south-1b
aws-project-nat-public1-ap-south-1a

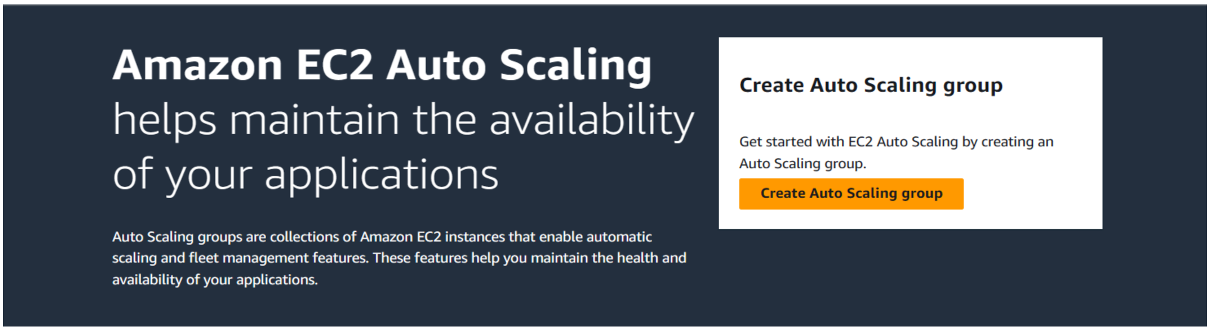
Now, Search for Auto Scaling Groups to launch instances and scale them automatically.



auto scaling groups

Search results for 'auto scaling groups'

Choose Create Auto Scaling Group.



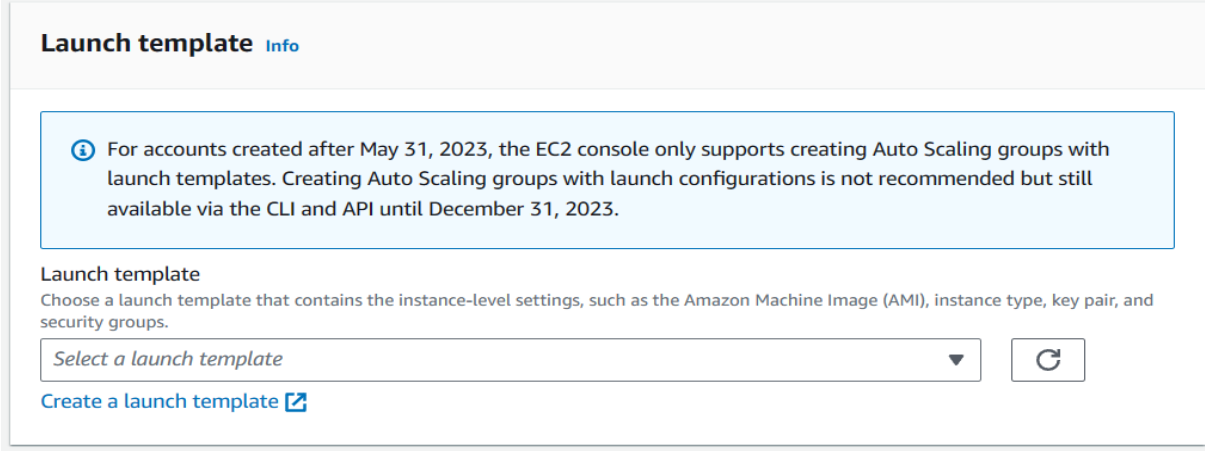
Amazon EC2 Auto Scaling helps maintain the availability of your applications

Auto Scaling groups are collections of Amazon EC2 instances that enable automatic scaling and fleet management features. These features help you maintain the health and availability of your applications.

Create Auto Scaling group

Get started with EC2 Auto Scaling by creating an Auto Scaling group.

To create ASG, we need a launch template. Choose Create a launch template.



Launch template Info

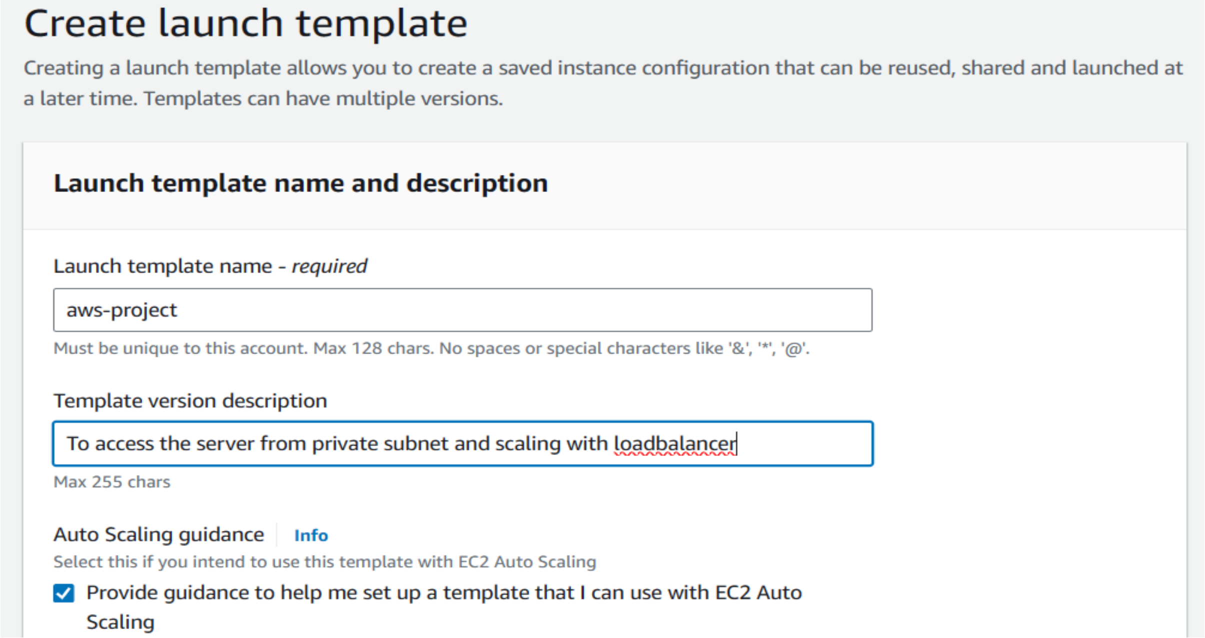
For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

Launch template

Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.

Select a launch template ▼

Name the template and give description to it.



Create launch template

Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple versions.

Launch template name and description

Launch template name - *required*

aws-project

Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '*', '@'.

Template version description

To access the server from private subnet and scaling with loadbalancer

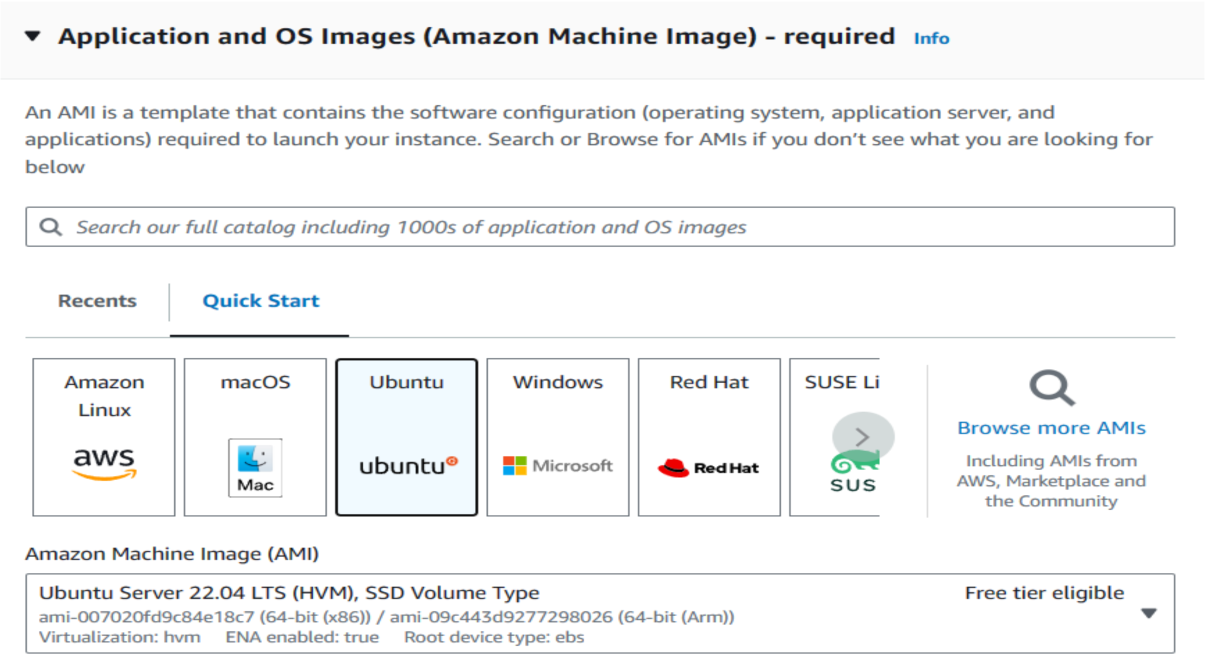
Max 255 chars

Auto Scaling guidance | Info

Select this if you intend to use this template with EC2 Auto Scaling

☑ Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

Choose the AMI.



▼ **Application and OS Images (Amazon Machine Image) - required** Info

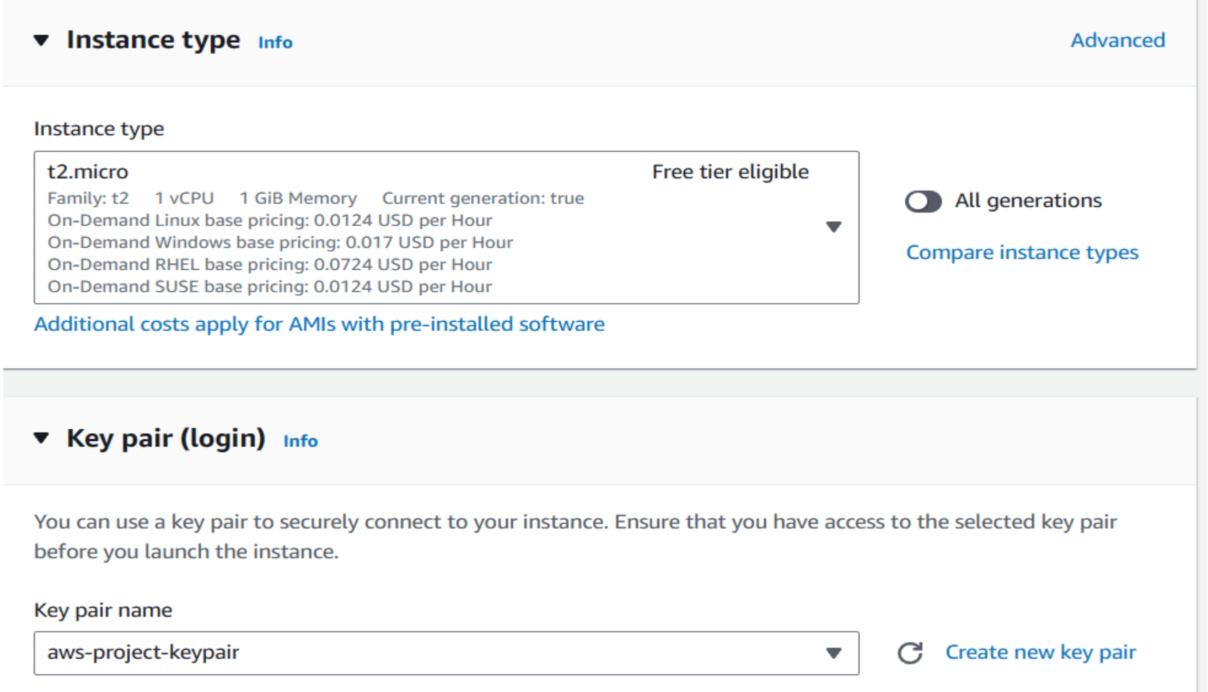
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type — Free tier eligible
ami-007020fd9c84e18c7 (64-bit (x86)) / ami-09c443d9277298026 (64-bit (Arm))
Virtualization: hvm   ENA enabled: true   Root device type: ebs

Choose the instance type and keypair. Make sure to use the same keypair.



▼ **Instance type** Info

Instance type

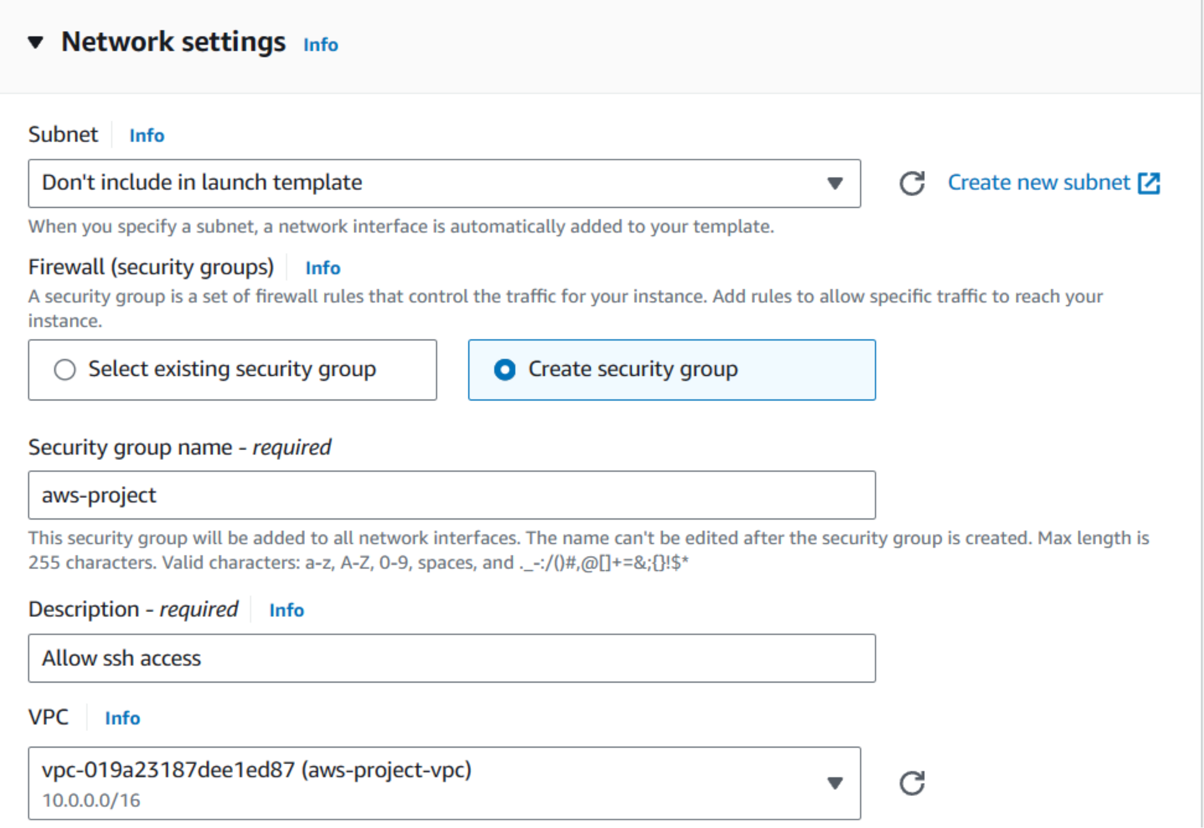
t2.micro — Free tier eligible
Family: t2   1 vCPU   1 GiB Memory   Current generation: true
On-Demand Linux base pricing: 0.0124 USD per Hour
On-Demand Windows base pricing: 0.017 USD per Hour
On-Demand RHEL base pricing: 0.0724 USD per Hour
On-Demand SUSE base pricing: 0.0124 USD per Hour

▼ **Key pair (login)** Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name: aws-project-keypair

Create security group and name it. Make sure to create in the same VPC that is created.



▼ Network settings Info

Subnet Info

Don't include in launch template ▼

Create new subnet

When you specify a subnet, a network interface is automatically added to your template.

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

○ Select existing security group ● Create security group

Security group name - *required*

aws-project

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and ._-:/()#,@[]+=&;{}!$*

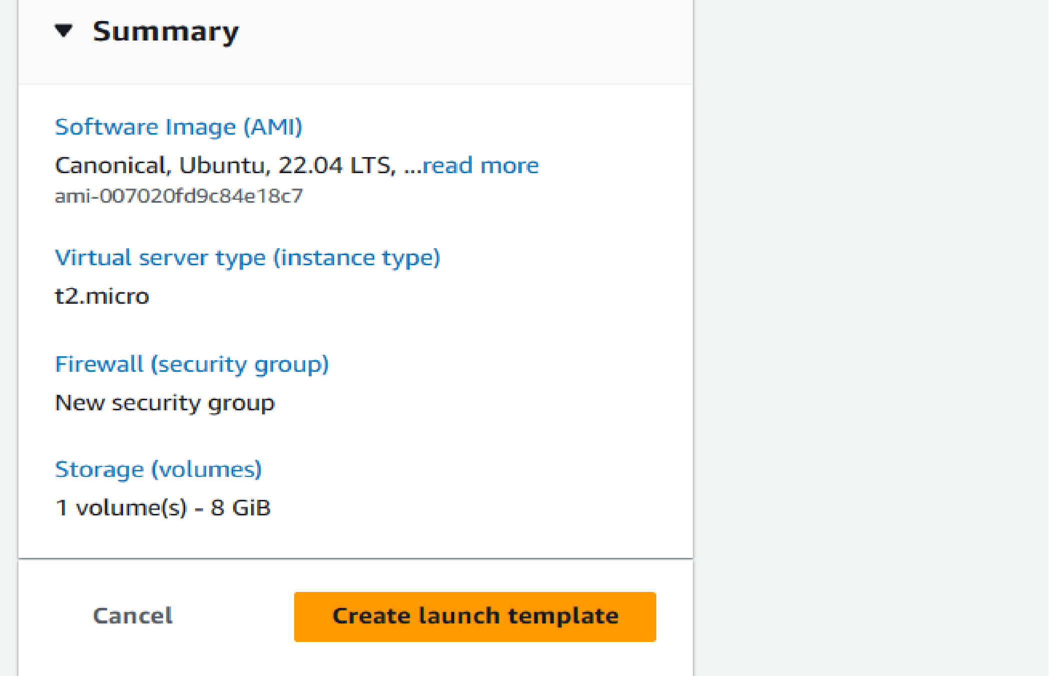
Description - *required* Info

Allow ssh access

VPC Info

vpc-019a23187dee1ed87 (aws-project-vpc)
10.0.0.0/16

Review the configurations and Choose Create Launch Template.



▼ Summary

Software Image (AMI)
Canonical, Ubuntu, 22.04 LTS, ...read more
ami-007020fd9c84e18c7

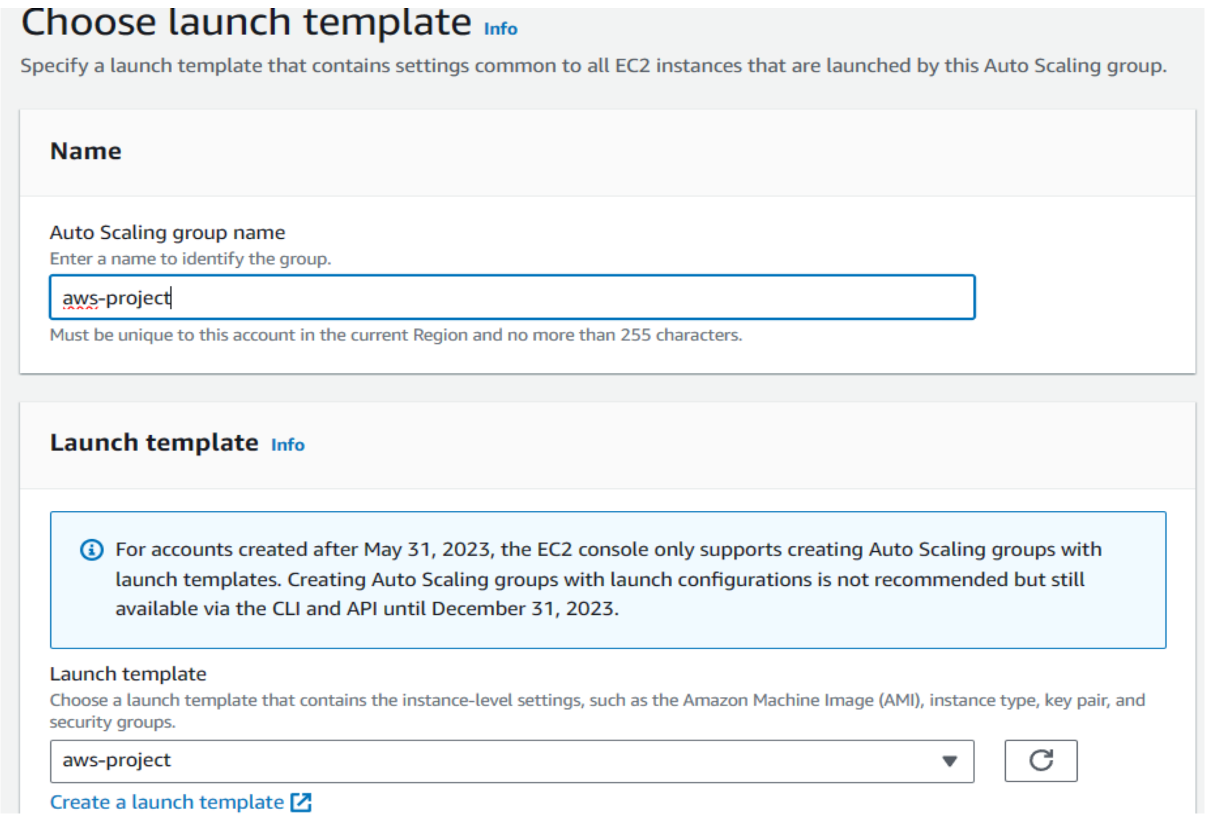
Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Cancel | Create launch template

Name the ASG and select the launch template that is created.



Choose launch template

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.

Name

Auto Scaling group name

Enter a name to identify the group.

aws-project

Must be unique to this account in the current Region and no more than 255 characters.

Launch template

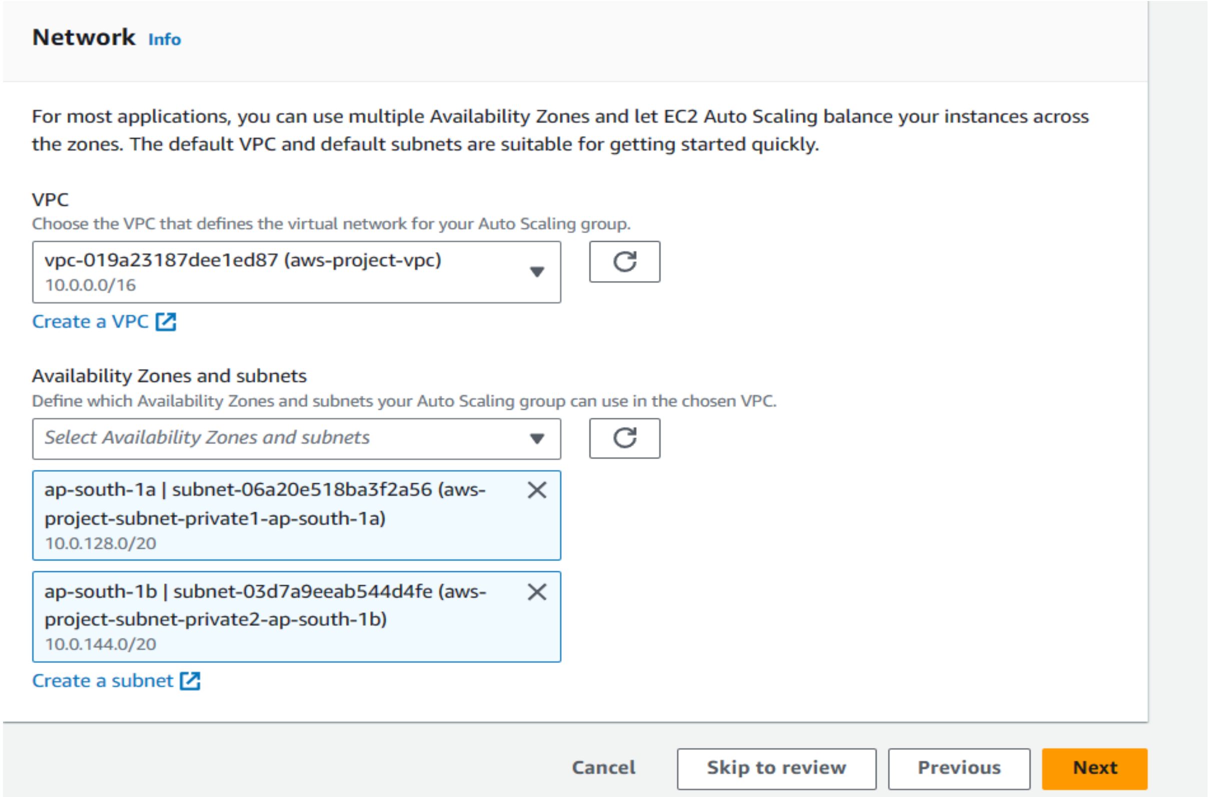
For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

Launch template

Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.

aws-project

Choose the VPC and make sure to create in both availability zones with private subnet. Choose next



Network

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC

Choose the VPC that defines the virtual network for your Auto Scaling group.

vpc-019a23187dee1ed87 (aws-project-vpc)
10.0.0.0/16

Availability Zones and subnets

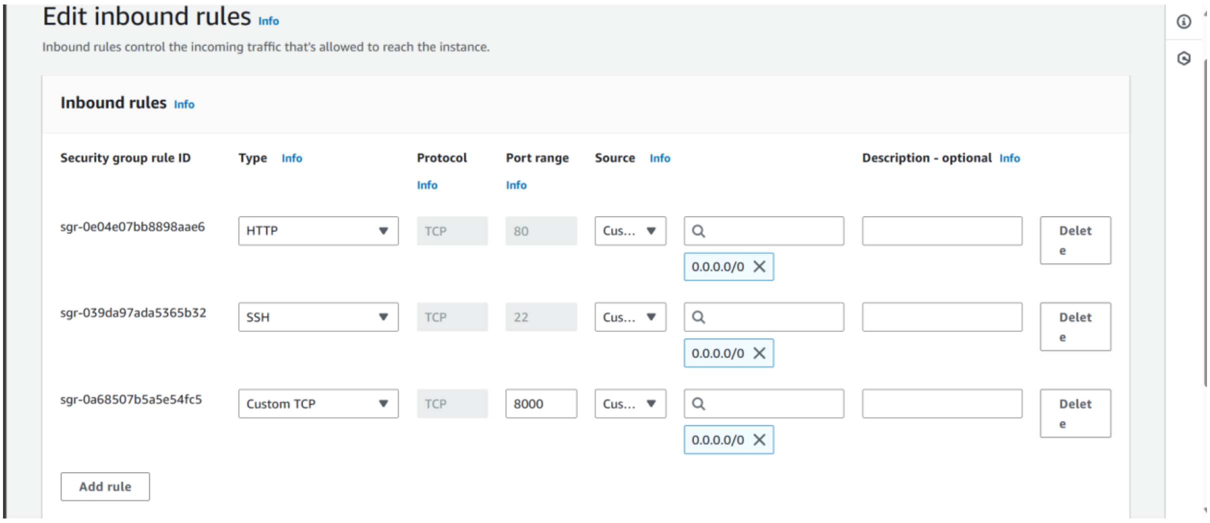
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.

Select Availability Zones and subnets

ap-south-1a | subnet-06a20e518ba3f2a56 (aws-project-subnet-private1-ap-south-1a)
10.0.128.0/20

ap-south-1b | subnet-03d7a9eeab544d4fe (aws-project-subnet-private2-ap-south-1b)
10.0.144.0/20

Make sure to open the port for 80,8000 and 22.



Edit inbound rules Info

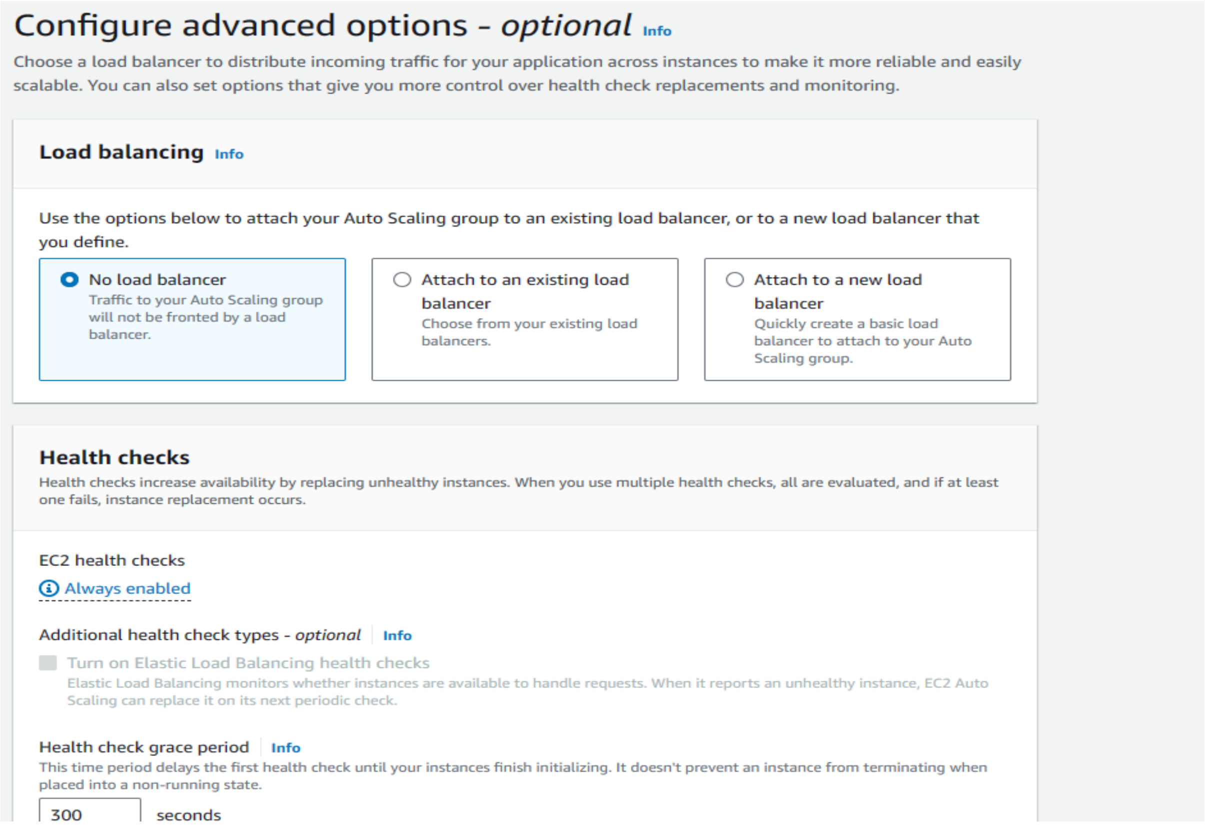
Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
sgr-0e04e07bb8898aae6	HTTP ▼	TCP	80	Cus... ▼	0.0.0.0/0 ✕		Delete
sgr-039da97ada5365b32	SSH ▼	TCP	22	Cus... ▼	0.0.0.0/0 ✕		Delete
sgr-0a68507b5a5e54fc5	Custom TCP ▼	TCP	8000	Cus... ▼	0.0.0.0/0 ✕		Delete

Add rule

As of now, no need to create the loadbalancer. Choose next.



Configure advanced options - *optional* Info

Choose a load balancer to distribute incoming traffic for your application across instances to make it more reliable and easily scalable. You can also set options that give you more control over health check replacements and monitoring.

Load balancing Info

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

● No load balancer
Traffic to your Auto Scaling group will not be fronted by a load balancer.

○ Attach to an existing load balancer
Choose from your existing load balancers.

○ Attach to a new load balancer
Quickly create a basic load balancer to attach to your Auto Scaling group.

Health checks

Health checks increase availability by replacing unhealthy instances. When you use multiple health checks, all are evaluated, and if at least one fails, instance replacement occurs.

EC2 health checks

Always enabled

Additional health check types - *optional* Info

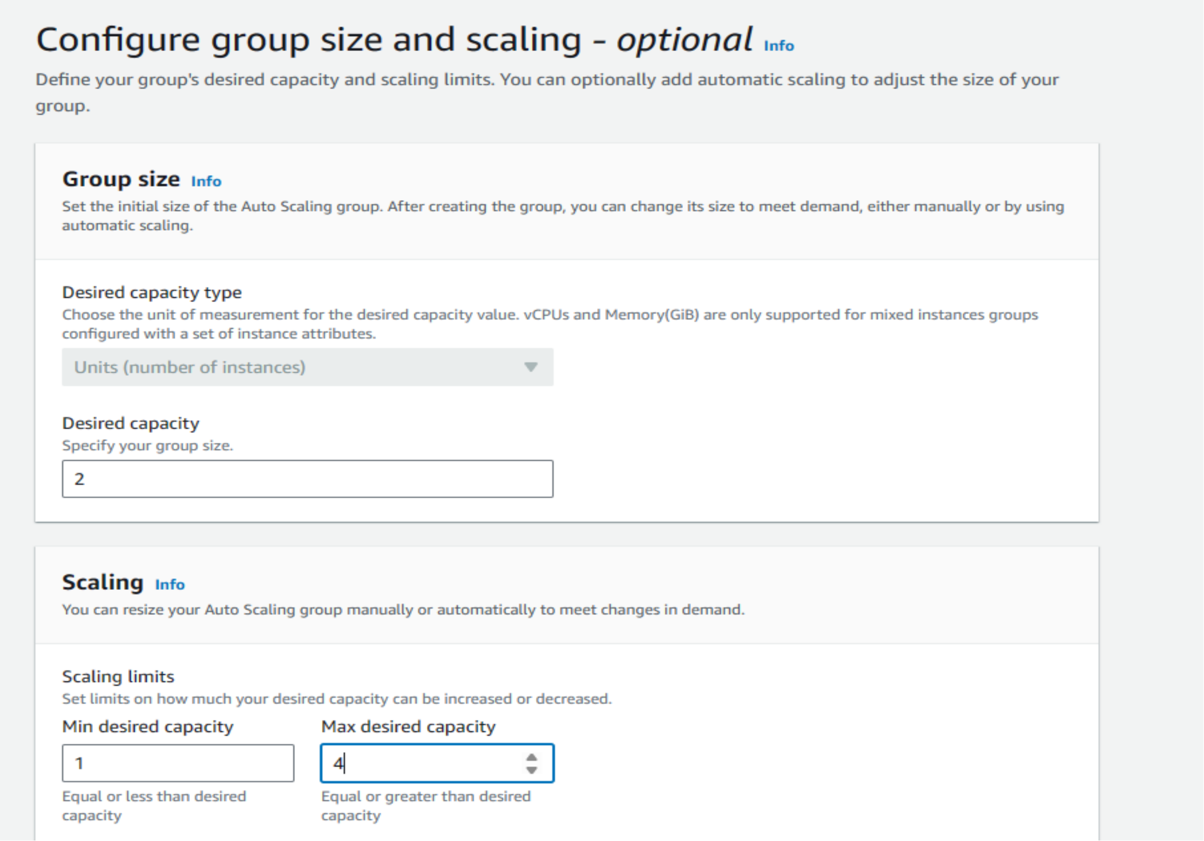
□ Turn on Elastic Load Balancing health checks
Elastic Load Balancing monitors whether instances are available to handle requests. When it reports an unhealthy instance, EC2 Auto Scaling can replace it on its next periodic check.

Health check grace period Info

This time period delays the first health check until your instances finish initializing. It doesn't prevent an instance from terminating when placed into a non-running state.

300 seconds

Enter the desired capacity that instances has to be created.Set the limit for scaling.



Configure group size and scaling - *optional* Info

Define your group's desired capacity and scaling limits. You can optionally add automatic scaling to adjust the size of your group.

Group size Info

Set the initial size of the Auto Scaling group. After creating the group, you can change its size to meet demand, either manually or by using automatic scaling.

Desired capacity type
Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.
Units (number of instances)

Desired capacity
Specify your group size.
2

Scaling Info

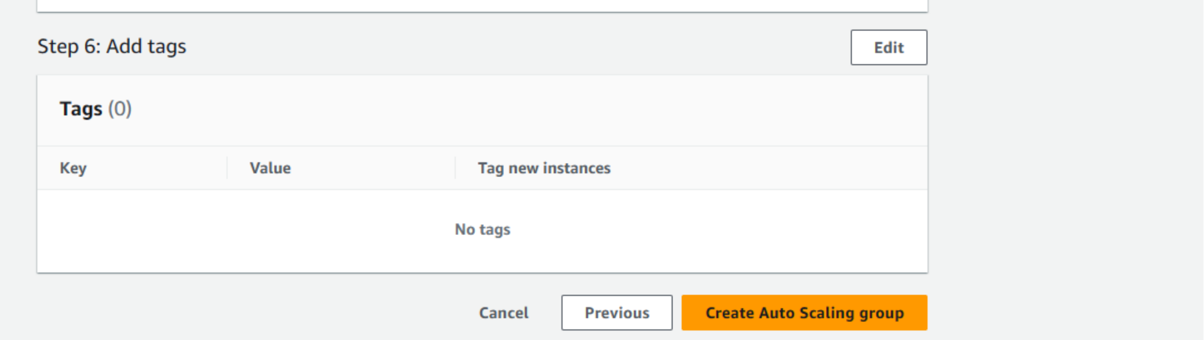
You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits
Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity: 1
Equal or less than desired capacity

Max desired capacity: 4
Equal or greater than desired capacity

Choose Create Auto Scaling Group.

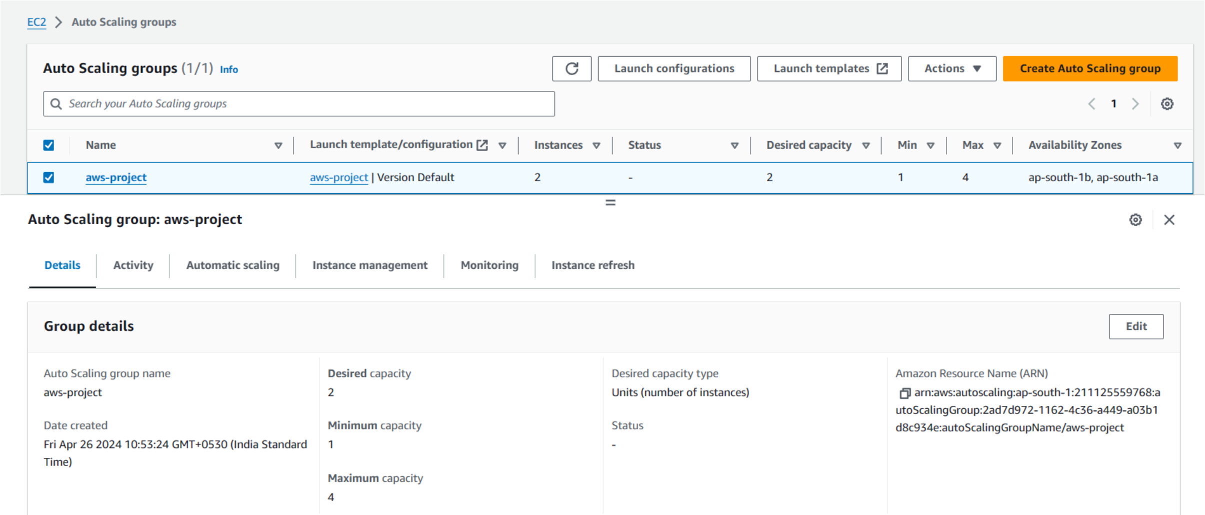


Step 6: Add tags

Tags (0)

Key	Value	Tag new instances
	No tags	

We can see the ASG is created. Navigate to instances to check whether the instances created or not.



EC2 > Auto Scaling groups

Auto Scaling groups (1/1) Info

☑	Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones
☑	aws-project	aws-project | Version Default	2	-	2	1	4	ap-south-1b, ap-south-1a

Auto Scaling group: aws-project

Group details

Auto Scaling group name
aws-project

Date created
Fri Apr 26 2024 10:53:24 GMT+0530 (India Standard Time)

Desired capacity
2

Minimum capacity
1

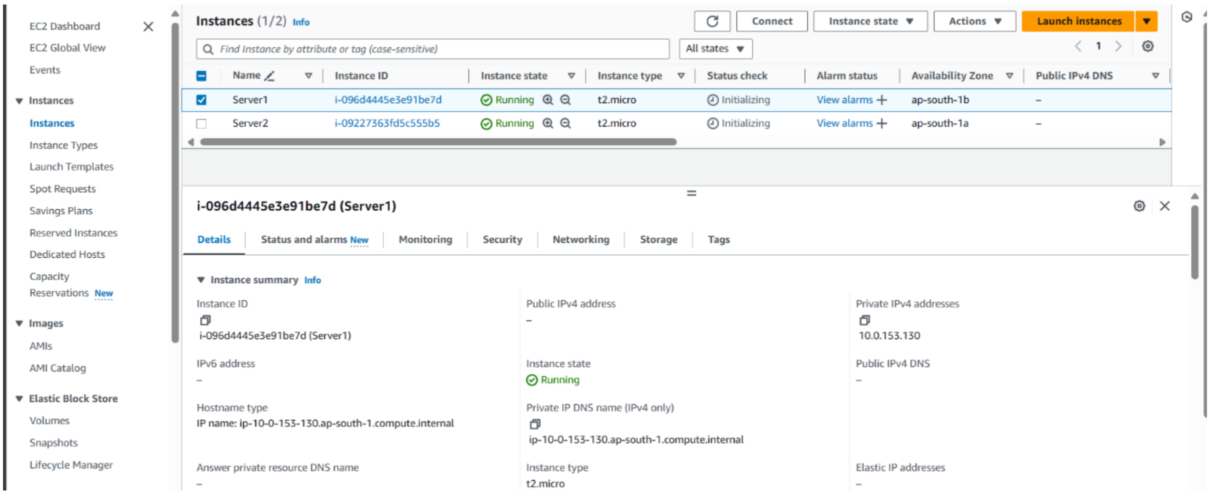
Maximum capacity
4

Desired capacity type
Units (number of instances)

Status
-

Amazon Resource Name (ARN)
arn:aws:autoscaling:ap-south-1:211125559768:autoScalingGroup:2ad7d972-1162-4c36-a449-a03b1d8c934e:autoScalingGroupName/aws-project

We can see two instances are running and I named them as Server1 and Server2 to recognise them w.r.t availability zones and we can see that instances are not assigned with public IP.



Instances (1/2) Info

▬	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
☑	Server1	i-096d4445e3e91be7d	Running	t2.micro	Initializing	View alarms +	ap-south-1b	–
□	Server2	i-09227363fd5c555b5	Running	t2.micro	Initializing	View alarms +	ap-south-1a	–

i-096d4445e3e91be7d (Server1)

▼ Instance summary Info

Instance ID
i-096d4445e3e91be7d (Server1)

IPv6 address
–

Hostname type
IP name: ip-10-0-153-130.ap-south-1.compute.internal

Answer private resource DNS name
–

Public IPv4 address
–

Instance state
Running

Private IP DNS name (IPv4 only)
ip-10-0-153-130.ap-south-1.compute.internal

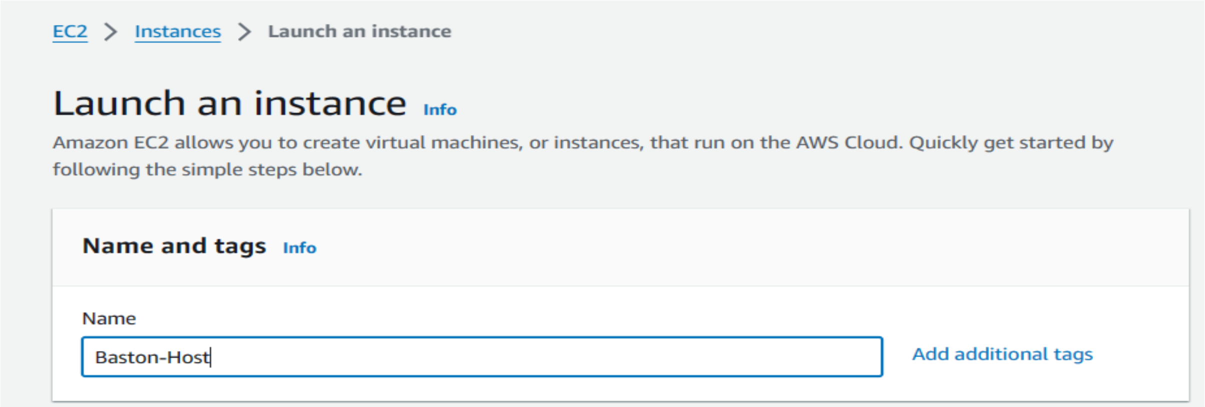
Instance type
t2.micro

Private IPv4 addresses
10.0.153.130

Public IPv4 DNS
–

Elastic IP addresses
–

Now create the Bastion-Host. Bastion-Host is an instance created to access the private instances from our local machine.



EC2 > Instances > Launch an instance

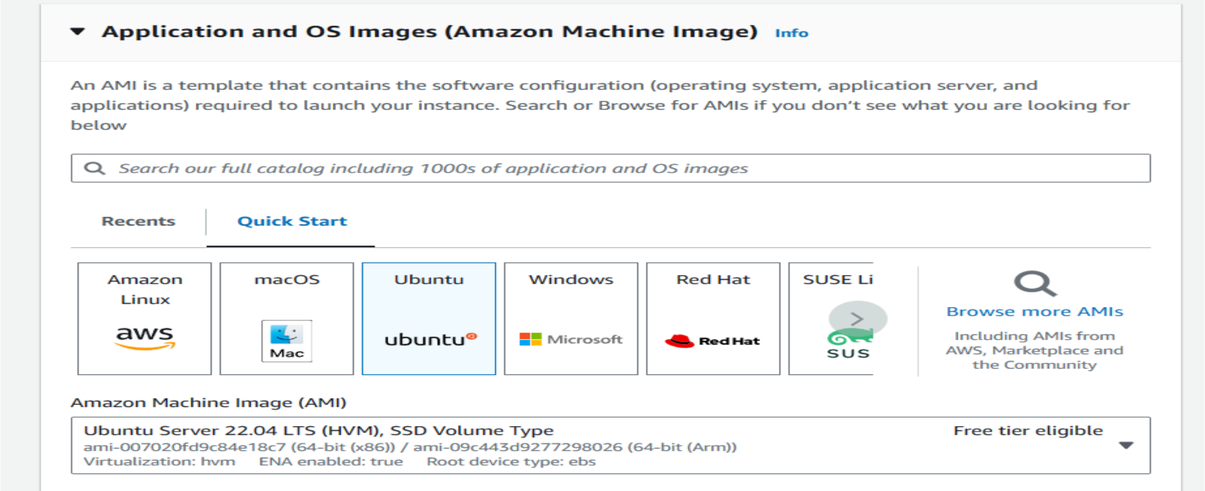
Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name
Baston-Host

Choose the AMI.



Application and OS Images (Amazon Machine Image) Info

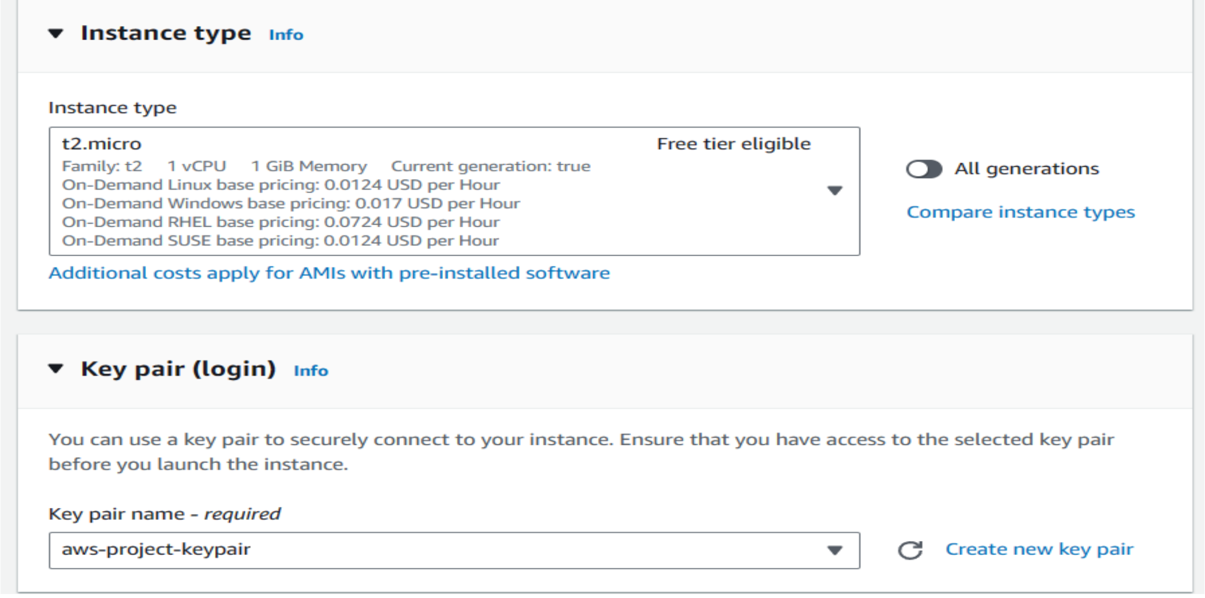
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type — Free tier eligible
ami-007020fd9c84e18c7 (64-bit (x86)) / ami-09c443d9277298026 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Select instance type and keypair.



Instance type Info

Instance type

t2.micro — Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Linux base pricing: 0.0124 USD per Hour
On-Demand Windows base pricing: 0.017 USD per Hour
On-Demand RHEL base pricing: 0.0724 USD per Hour
On-Demand SUSE base pricing: 0.0124 USD per Hour

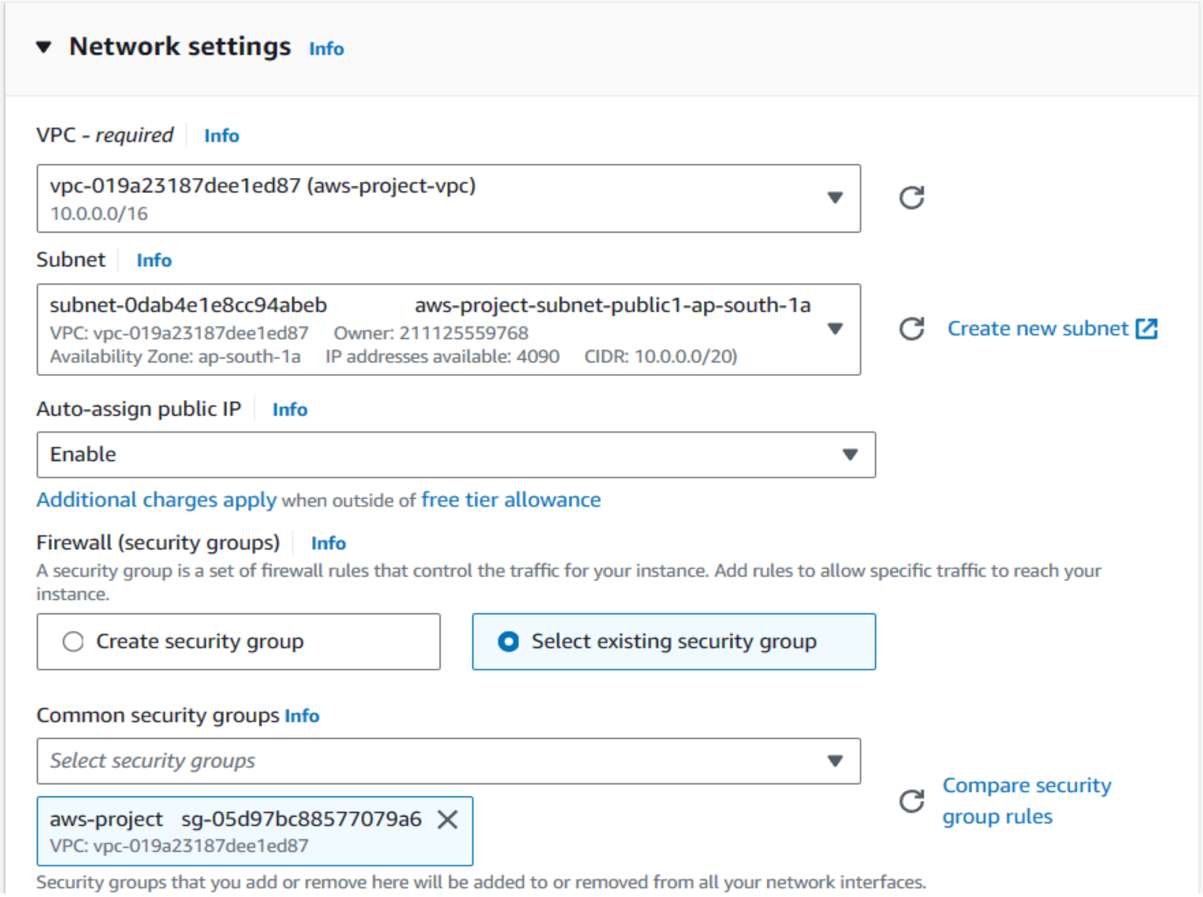
Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

aws-project-keypair

Choose the VPC and public subnet. Enable auto-assign public IP.Select the security group that is created for launch template.



▼ Network settings Info

VPC - *required* Info

vpc-019a23187dee1ed87 (aws-project-vpc)
10.0.0.0/16

Subnet Info

subnet-0dab4e1e8cc94abeb aws-project-subnet-public1-ap-south-1a
VPC: vpc-019a23187dee1ed87 Owner: 211125559768
Availability Zone: ap-south-1a IP addresses available: 4090 CIDR: 10.0.0.0/20)

Create new subnet

Auto-assign public IP Info

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

○ Create security group ● Select existing security group

Common security groups Info

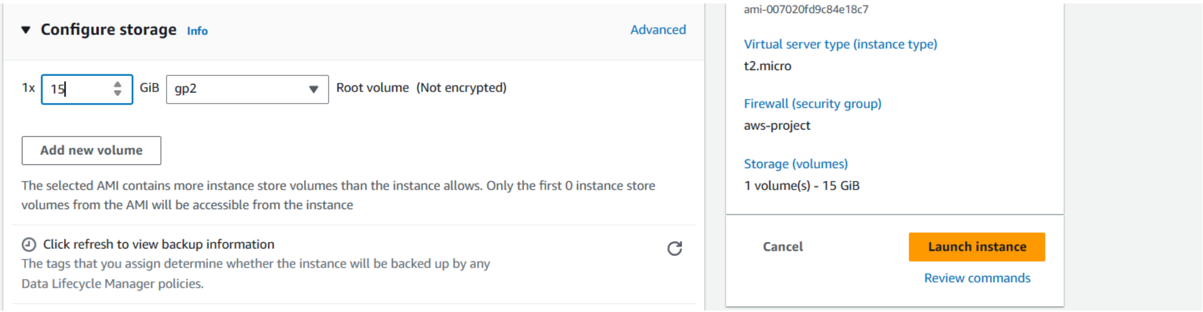
Select security groups

aws-project sg-05d97bc88577079a6
VPC: vpc-019a23187dee1ed87

Compare security group rules

Security groups that you add or remove here will be added to or removed from all your network interfaces.

Enter the size of the volume and Choose Launch instance.



▼ Configure storage Info Advanced

1x 15 GiB gp2 Root volume (Not encrypted)

Add new volume

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

Click refresh to view backup information
The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

ami-007020fd9c84e18c7

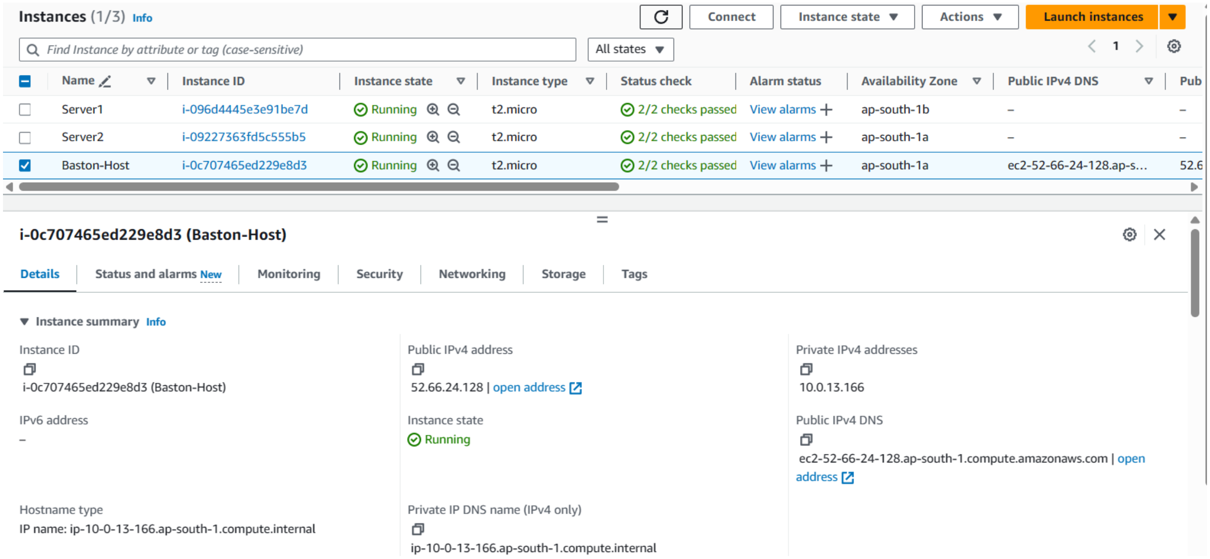
Virtual server type (instance type)
t2.micro

Firewall (security group)
aws-project

Storage (volumes)
1 volume(s) - 15 GiB

Cancel Launch instance
Review commands

The Bastion-Host is created and the public IP has been assigned to the instance.



Instances (1/3) Info

☐	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Pub
☐	Server1	i-096d4445e3e91be7d	✓ Running	t2.micro	✓ 2/2 checks passed	View alarms +	ap-south-1b	–	–
☐	Server2	i-09227363fd5c555b5	✓ Running	t2.micro	✓ 2/2 checks passed	View alarms +	ap-south-1a	–	–
☑	Baston-Host	i-0c707465ed229e8d3	✓ Running	t2.micro	✓ 2/2 checks passed	View alarms +	ap-south-1a	ec2-52-66-24-128.ap-s...	52.6

i-0c707465ed229e8d3 (Baston-Host)

▼ Instance summary Info

Instance ID
i-0c707465ed229e8d3 (Baston-Host)

Public IPv4 address
52.66.24.128 | open address

Private IPv4 addresses
10.0.13.166

IPv6 address
–

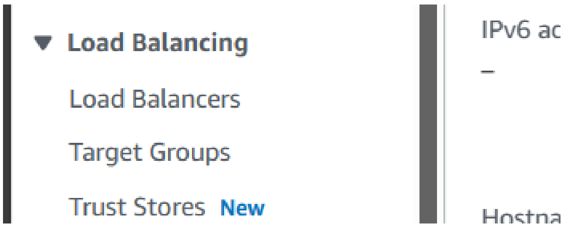
Instance state
✓ Running

Public IPv4 DNS
ec2-52-66-24-128.ap-south-1.compute.amazonaws.com | open address

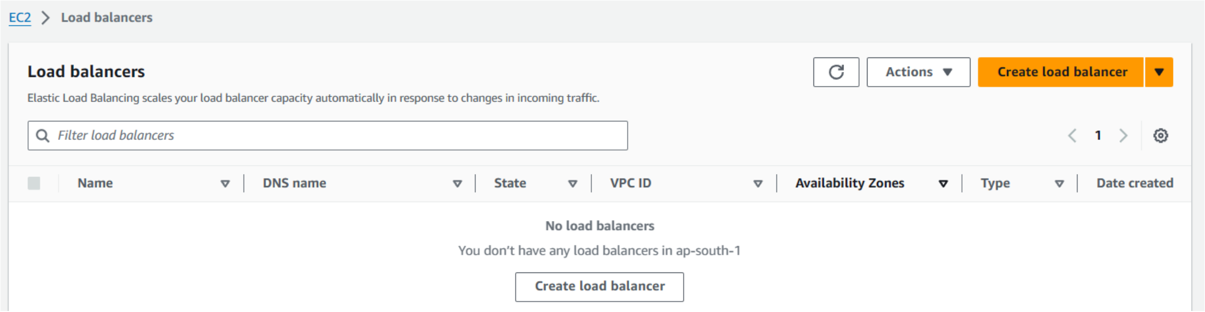
Hostname type
IP name: ip-10-0-13-166.ap-south-1.compute.internal

Private IP DNS name (IPv4 only)
ip-10-0-13-166.ap-south-1.compute.internal

In the EC2 dashboard itself, Choose the Loadbalancers.



Choose Create Load Balancer.



Load balancers

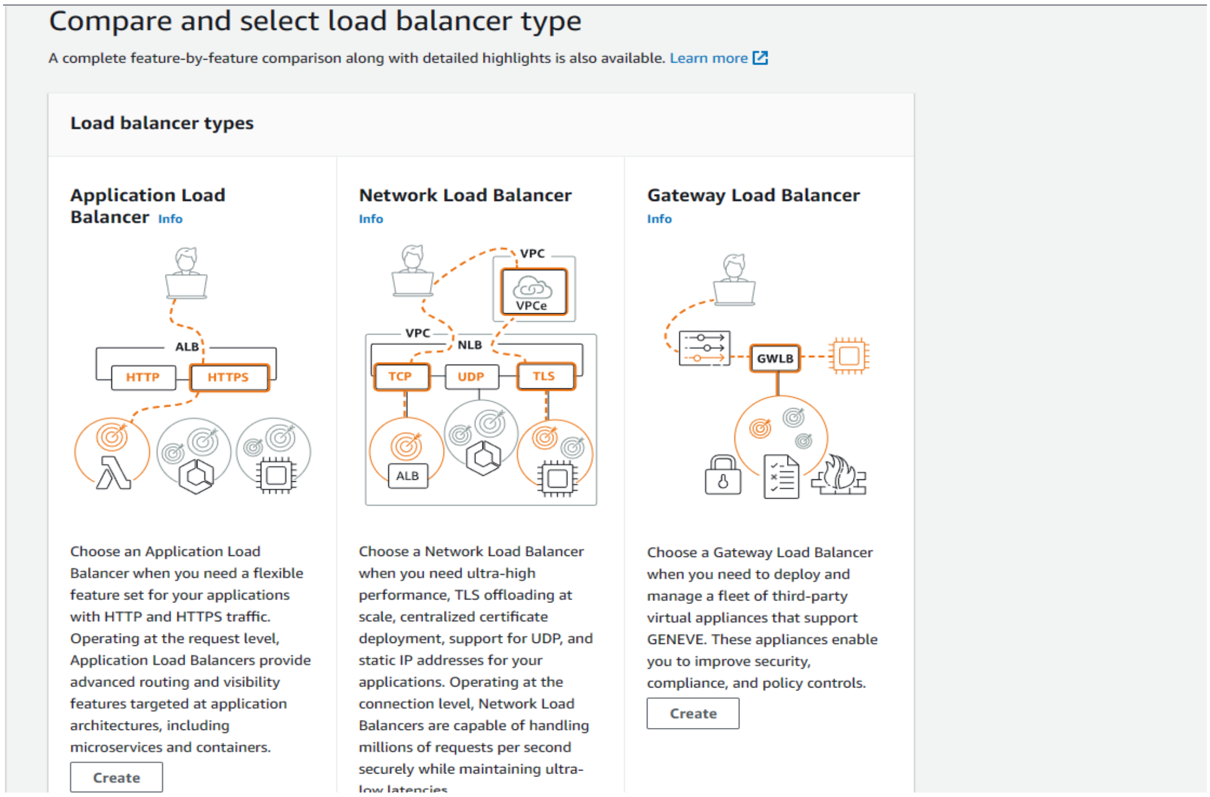
Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

☐	Name	DNS name	State	VPC ID	Availability Zones	Type	Date created

No load balancers

You don't have any load balancers in ap-south-1

The Load Balancers helps to distribute the traffic between the servers and helps them to run efficiently. Create Application Load Balancer.



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

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.

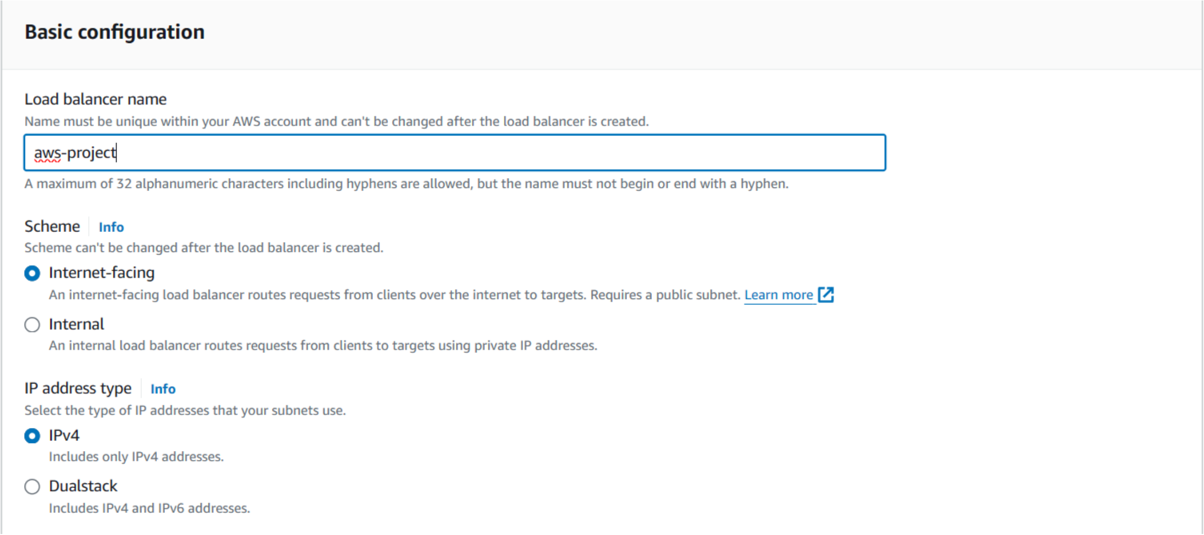
Network Load Balancer Info

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

Gateway Load Balancer Info

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.

Name the ALB and make it internet-facing and IPv4.



Basic configuration

Load balancer name

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

aws-project

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

An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. Learn more

○ Internal

An internal load balancer routes requests from clients to targets using private IP addresses.

IP address type Info

Select the type of IP addresses that your subnets use.

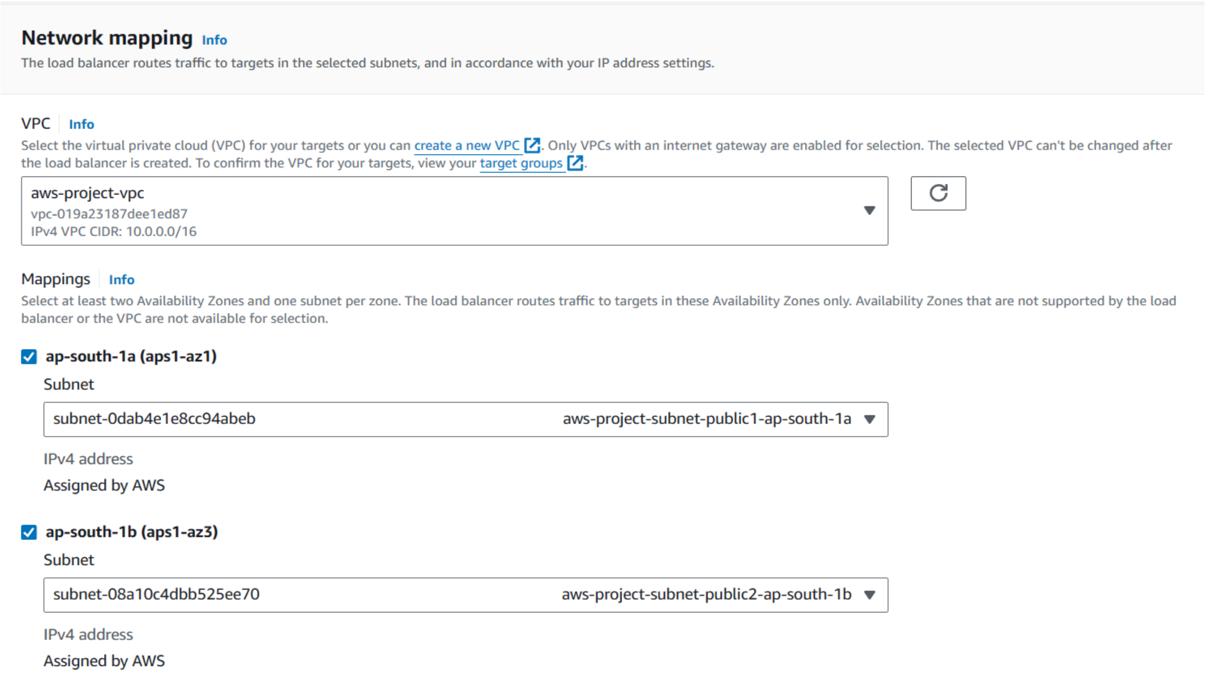
● IPv4

Includes only IPv4 addresses.

○ Dualstack

Includes IPv4 and IPv6 addresses.

Choose the VPC that has been created and map the public subnets from both of the availability zones.



Network mapping Info

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

VPC Info

Select the virtual private cloud (VPC) for your targets or you can create a new VPC. Only VPCs with an internet gateway are enabled for selection. The selected VPC can't be changed after the load balancer is created. To confirm the VPC for your targets, view your target groups.

aws-project-vpc
vpc-019a23187dee1ed87
IPv4 VPC CIDR: 10.0.0.0/16

Mappings Info

Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

☑ **ap-south-1a (aps1-az1)**

Subnet

subnet-0dab4e1e8cc94abeb aws-project-subnet-public1-ap-south-1a

IPv4 address

Assigned by AWS

☑ **ap-south-1b (aps1-az3)**

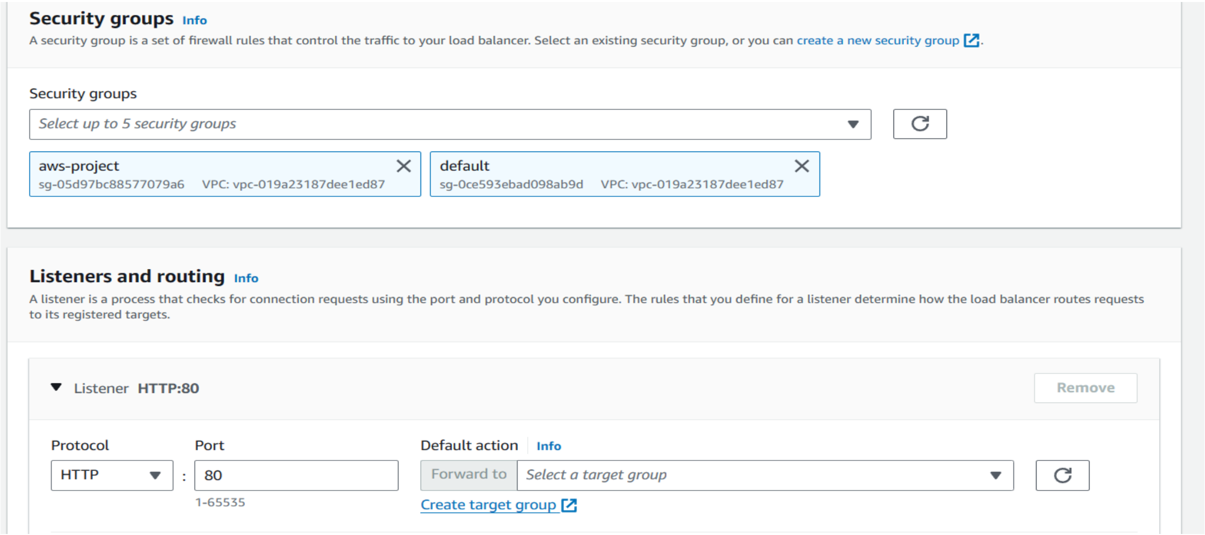
Subnet

subnet-08a10c4dbb525ee70 aws-project-subnet-public2-ap-south-1b

IPv4 address

Assigned by AWS

Choose the security group that is created and make to use the same SG. Create a target group. Target groups are used to route the traffic to different servers and monitor the health of the instances.



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

aws-project ✕
sg-05d97bc88577079a6 VPC: vpc-019a23187dee1ed87

default ✕
sg-0ce593ebad098ab9d VPC: vpc-019a23187dee1ed87

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: HTTP

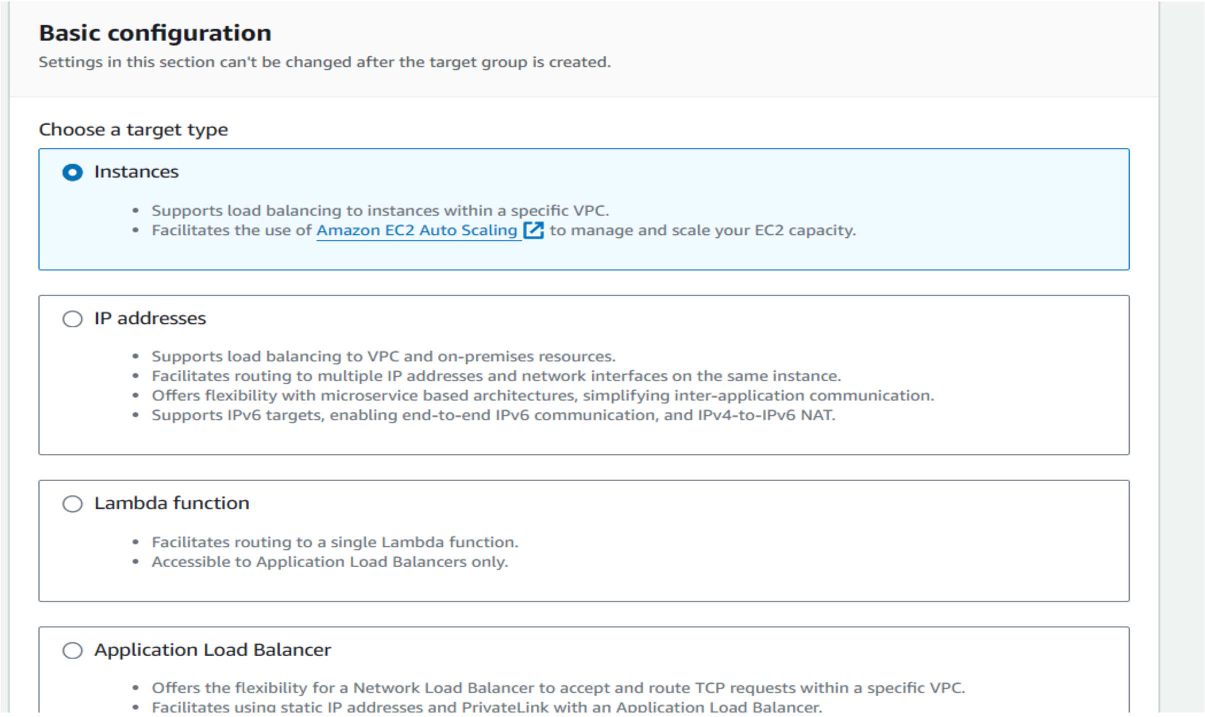
Port: 80
1-65535

Default action Info

Forward to *Select a target group*

Create target group

Choose instances.



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.

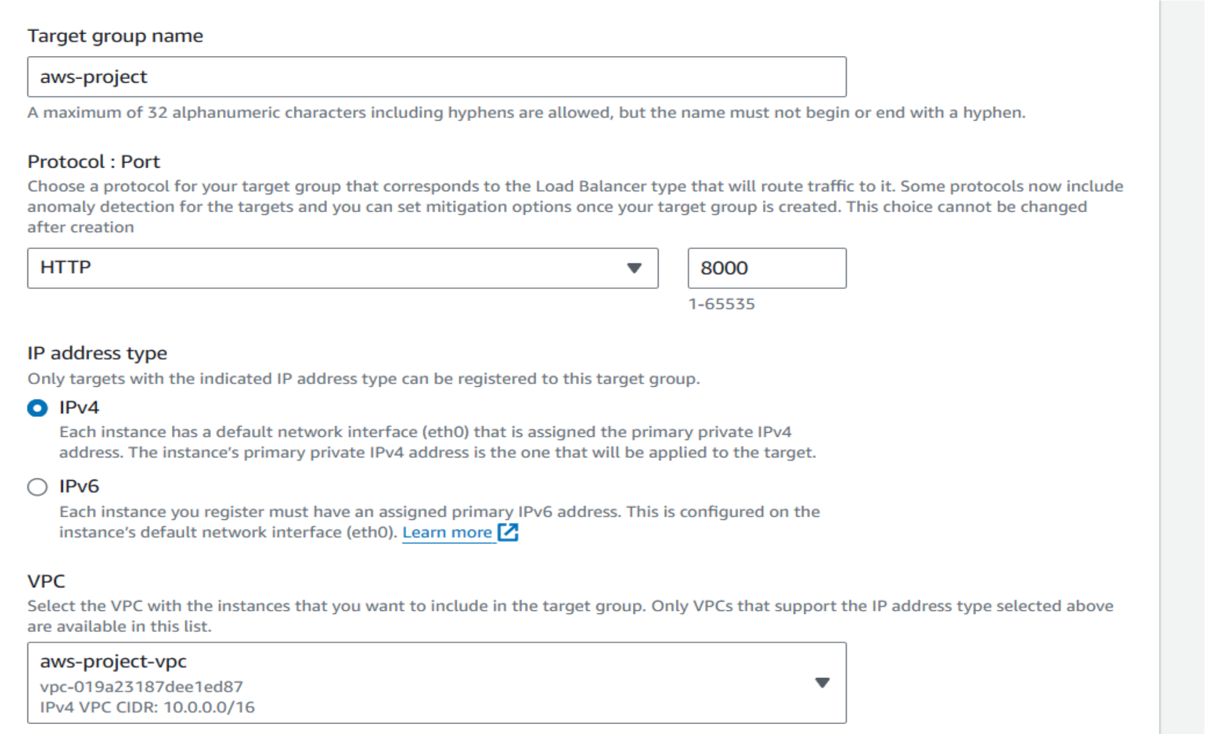
○ Lambda function

- 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.

Name it and select the procotol and give the port. Select the VPC.



Target group name

aws-project

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

Protocol : Port

Choose a protocol for your target group that corresponds to the Load Balancer type that will route traffic to it. Some protocols now include anomaly detection for the targets and you can set mitigation options once your target group is created. This choice cannot be changed after creation

HTTP ▼ 8000

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

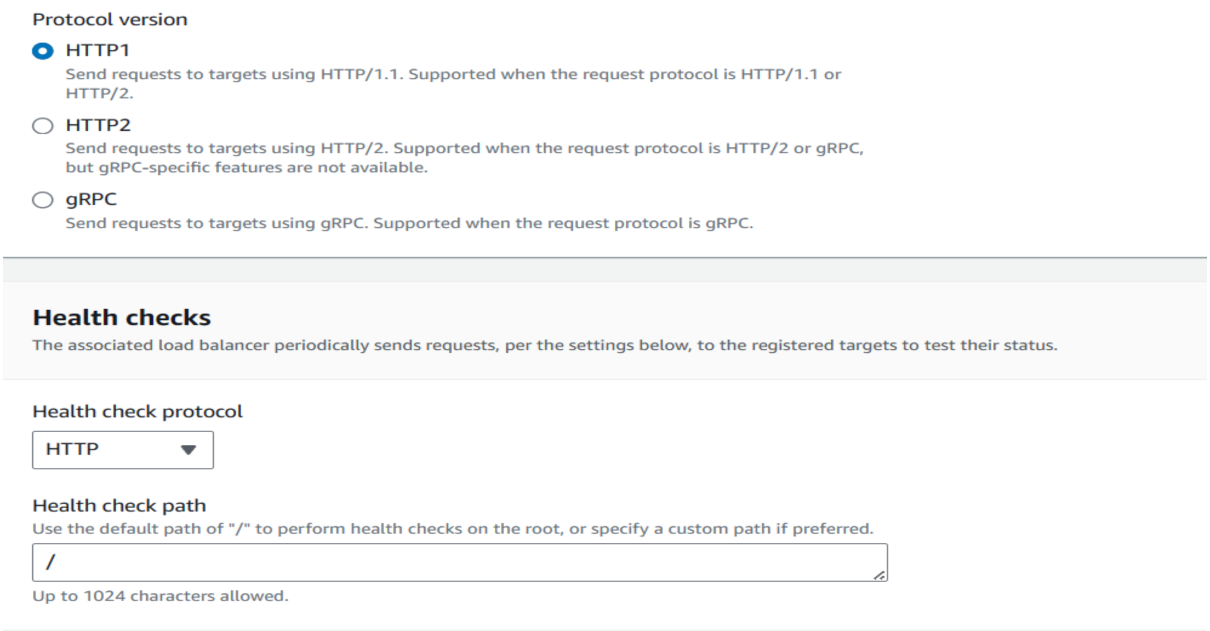
Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). Learn more

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.

aws-project-vpc
vpc-019a23187dee1ed87
IPv4 VPC CIDR: 10.0.0.0/16 ▼

Leave the health checks as it is.



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.

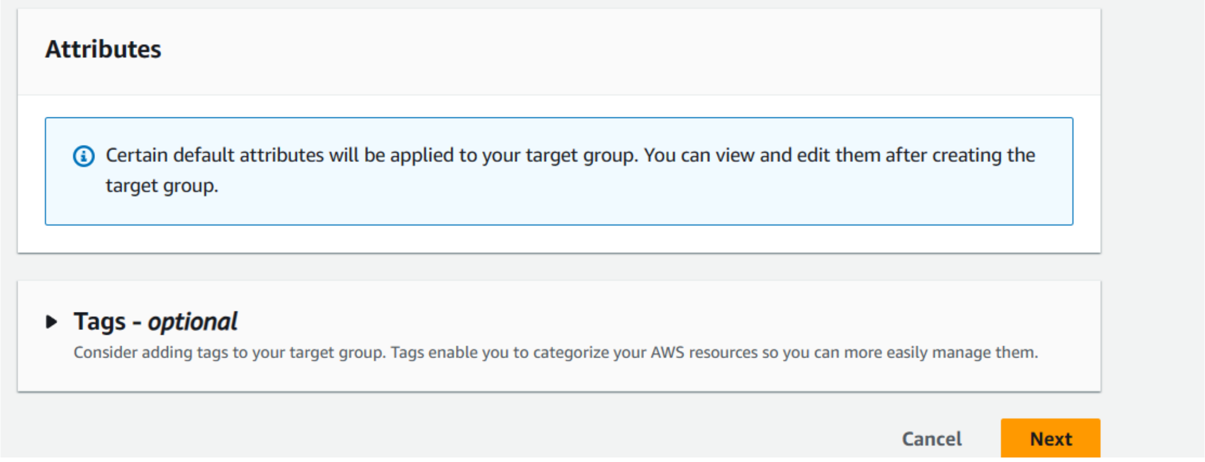
○ gRPC
Send requests to targets using gRPC. Supported when the request protocol is gRPC.

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.

Choose Next.

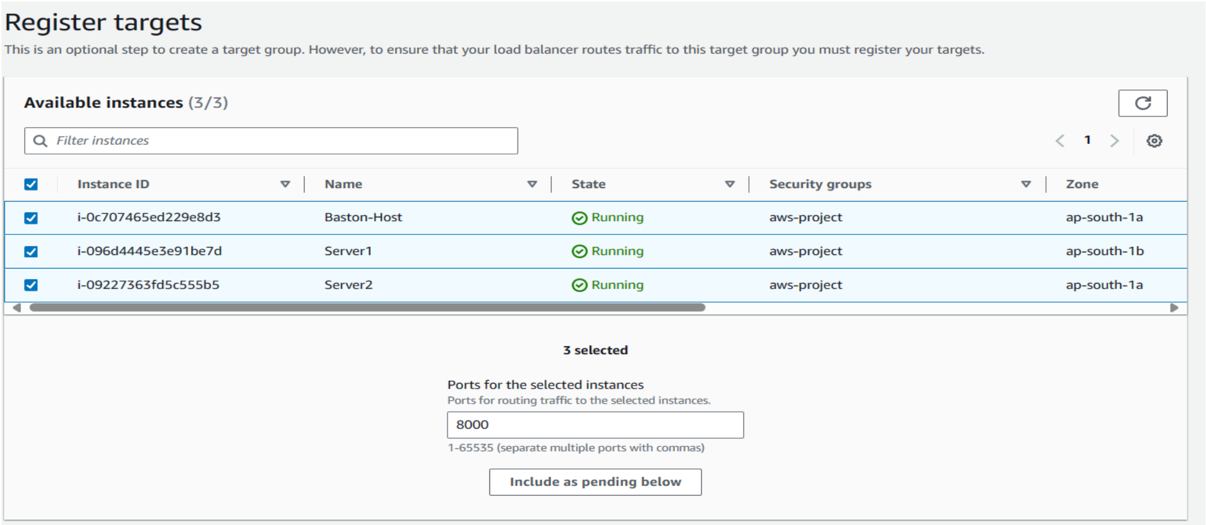


Attributes

ⓘ 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.

Check the instances and choose Incluse as pending below.



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.

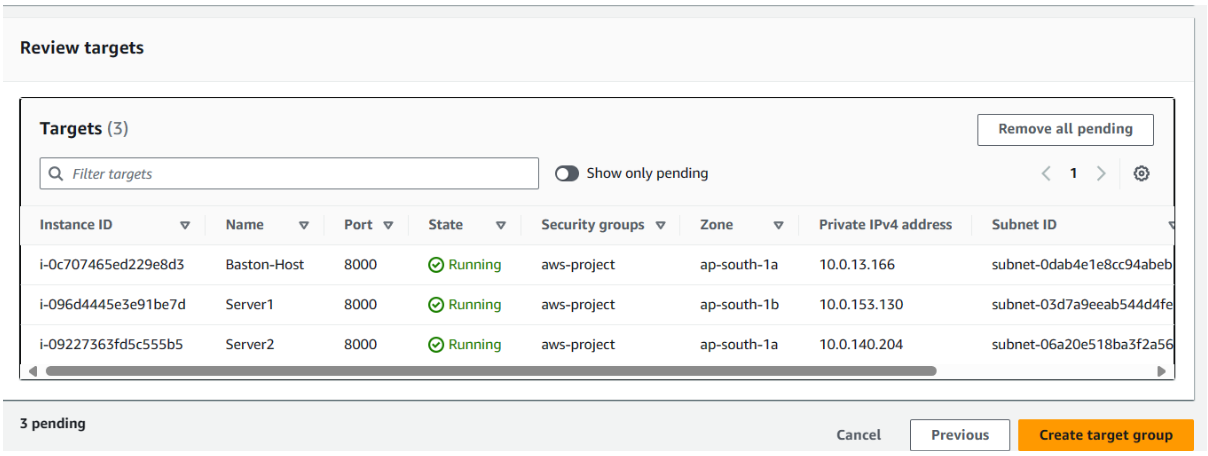
Available instances (3/3)

☑	Instance ID	Name	State	Security groups	Zone
☑	i-0c707465ed229e8d3	Baston-Host	✓ Running	aws-project	ap-south-1a
☑	i-096d4445e3e91be7d	Server1	✓ Running	aws-project	ap-south-1b
☑	i-09227363fd5c555b5	Server2	✓ Running	aws-project	ap-south-1a

3 selected

Ports for the selected instances
Ports for routing traffic to the selected instances.
8000
1-65535 (separate multiple ports with commas)

Review targets and Choose Create Target Group.



Review targets

Targets (3)

Remove all pending

Filter targets

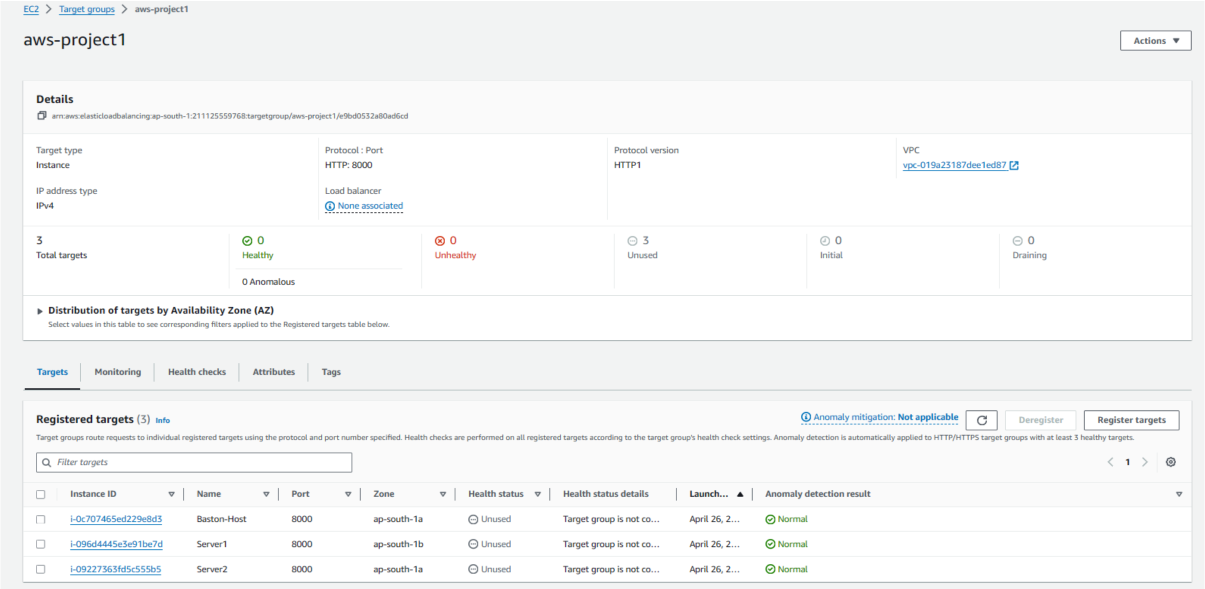
Show only pending

Instance ID	Name	Port	State	Security groups	Zone	Private IPv4 address	Subnet ID
i-0c707465ed229e8d3	Baston-Host	8000	✓ Running	aws-project	ap-south-1a	10.0.13.166	subnet-0dab4e1e8cc94abeb
i-096d4445e3e91be7d	Server1	8000	✓ Running	aws-project	ap-south-1b	10.0.153.130	subnet-03d7a9eeab544d4fe
i-09227363fd5c555b5	Server2	8000	✓ Running	aws-project	ap-south-1a	10.0.140.204	subnet-06a20e518ba3f2a56

3 pending

Cancel | Previous | Create target group

Overview of the target group.



EC2 > Target groups > aws-project1

aws-project1

Actions ▼

Details

arn:aws:elasticloadbalancing:ap-south-1:211125559768:targetgroup/aws-project1/e9bd0532a80ad6cd

Target type: Instance

IP address type: IPv4

Protocol : Port: HTTP: 8000

Load balancer: None associated

Protocol version: HTTP1

VPC: vpc-019a23187dee1ed87

3 Total targets | 0 Healthy | 0 Unhealthy | 3 Unused | 0 Initial | 0 Draining

0 Anomalous

Distribution of targets by Availability Zone (AZ)

Select values in this table to see corresponding filters applied to the Registered targets table below.

Targets | Monitoring | Health checks | Attributes | Tags

Registered targets (3) Info

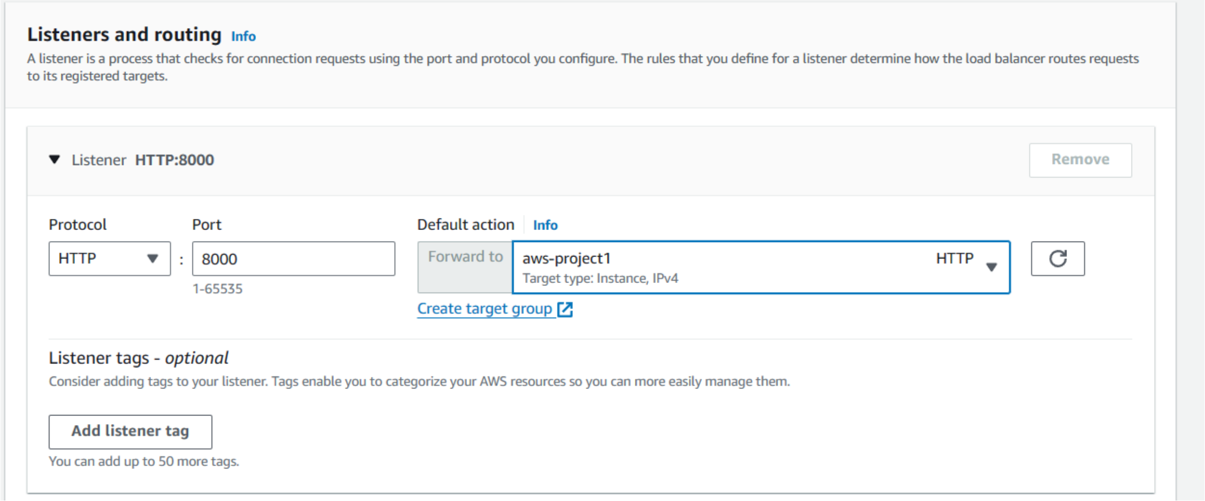
Anomaly mitigation: Not applicable | Deregister | Register targets

Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.

Filter targets

□	Instance ID	Name	Port	Zone	Health status	Health status details	Launch...	Anomaly detection result
□	i-0c707465ed229e8d3	Baston-Host	8000	ap-south-1a	Unused	Target group is not co...	April 26, 2...	✓ Normal
□	i-096d4445e3e91be7d	Server1	8000	ap-south-1b	Unused	Target group is not co...	April 26, 2...	✓ Normal
□	i-09227363fd5c555b5	Server2	8000	ap-south-1a	Unused	Target group is not co...	April 26, 2...	✓ Normal

Select the target group.



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:8000**

Remove

Protocol: HTTP

Port: 8000 (1-65535)

Default action Info: Forward to aws-project1 — HTTP — Target type: Instance, IPv4

Create target group

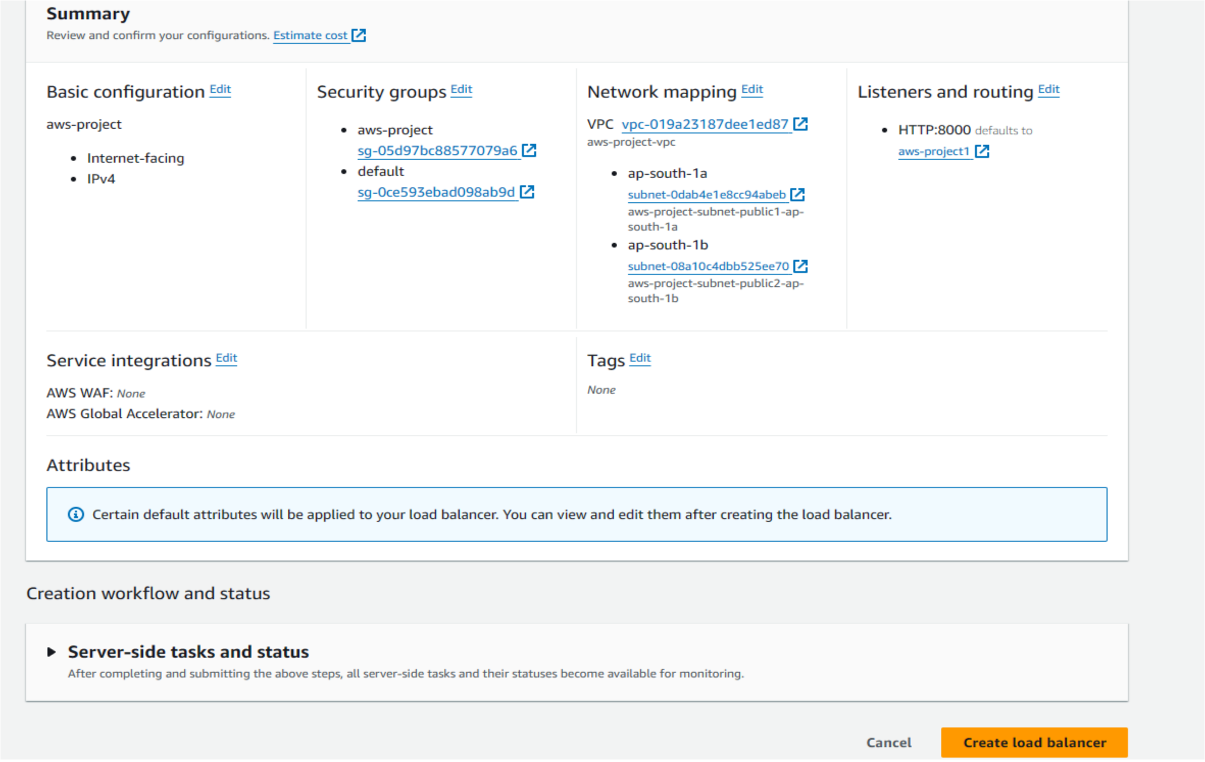
Listener tags - *optional*

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.

Review configurations and Create Load Balancer.



Summary

Review and confirm your configurations. Estimate cost

Basic configuration Edit

aws-project

- Internet-facing
- IPv4

Security groups Edit

- aws-project
  sg-05d97bc88577079a6
- default
  sg-0ce593ebad098ab9d

Network mapping Edit

VPC vpc-019a23187dee1ed87
aws-project-vpc

- ap-south-1a
  subnet-0dab4e1e8cc94abeb
  aws-project-subnet-public1-ap-south-1a
- ap-south-1b
  subnet-08a10c4dbb525ee70
  aws-project-subnet-public2-ap-south-1b

Listeners and routing Edit

- HTTP:8000 defaults to
  aws-project1

Service integrations Edit

AWS WAF: *None*
AWS Global Accelerator: *None*

Tags Edit

None

Attributes

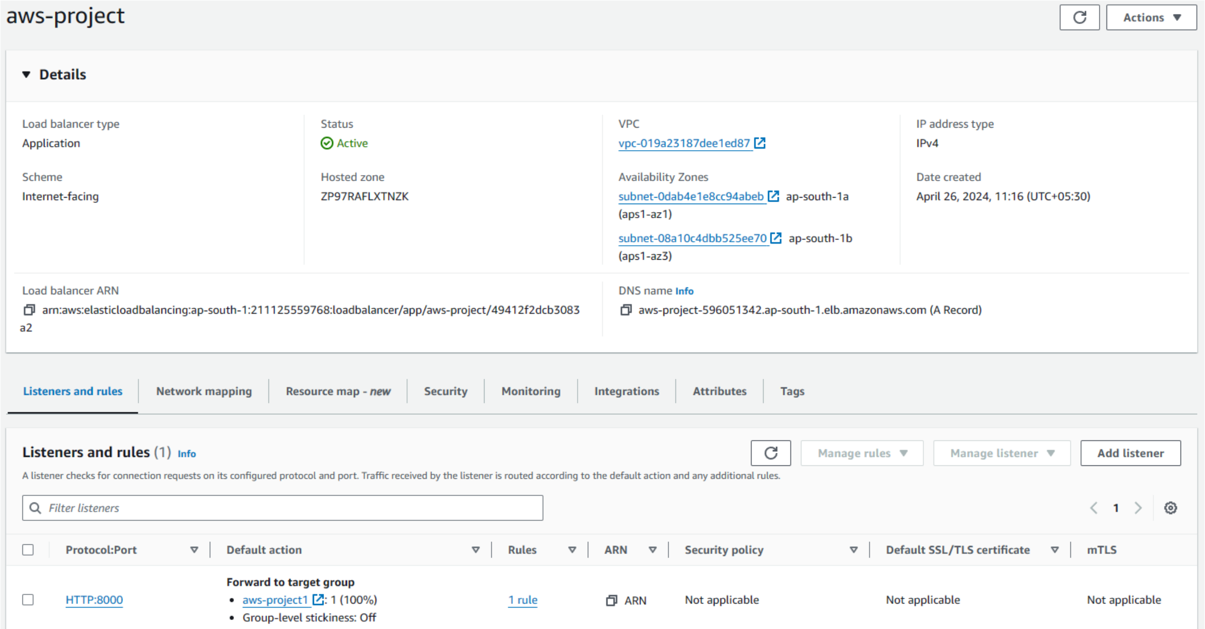
Certain default attributes will be applied to your load balancer. You can view and edit them after creating the load balancer.

Creation workflow and status

Server-side tasks and status

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

Overview of load balancer and it is active.



aws-project

Details

Load balancer type Application	Status Active	VPC vpc-019a23187dee1ed87	IP address type IPv4
Scheme Internet-facing	Hosted zone ZP97RAFLXTNZK	Availability Zones subnet-0dab4e1e8cc94abeb ap-south-1a (aps1-az1) subnet-08a10c4dbb525ee70 ap-south-1b (aps1-az3)	Date created April 26, 2024, 11:16 (UTC+05:30)

Load balancer ARN
arn:aws:elasticloadbalancing:ap-south-1:211125559768:loadbalancer/app/aws-project/49412f2dcb3083a2

DNS name Info
aws-project-596051342.ap-south-1.elb.amazonaws.com (A Record)

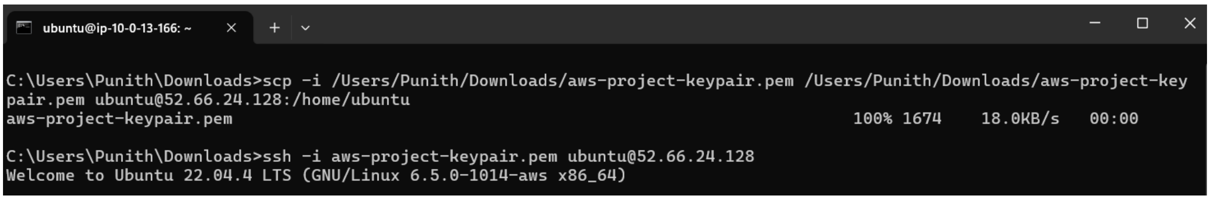
Listeners and rules | Network mapping | Resource map - *new* | Security | Monitoring | Integrations | Attributes | Tags

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.

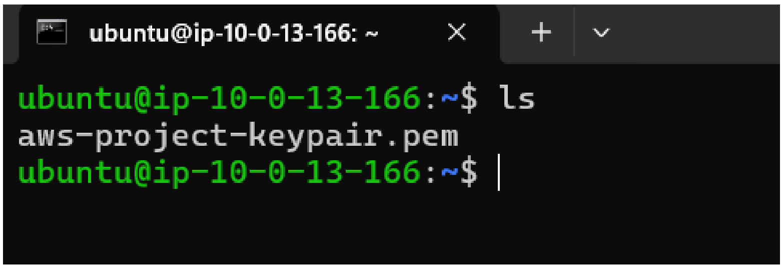
Protocol:Port	Default action	Rules	ARN	Security policy	Default SSL/TLS certificate	mTLS
HTTP:8000	**Forward to target group** • aws-project1: 1 (100%) • Group-level stickiness: Off	1 rule	ARN	Not applicable	Not applicable	Not applicable

Now, Open Command Prompt and Make sure to navigate to path where the keypair is present. Enter the following command to copy the keypair to bastion-host. After copying, ssh to the bastion-host with keypair and IP address.



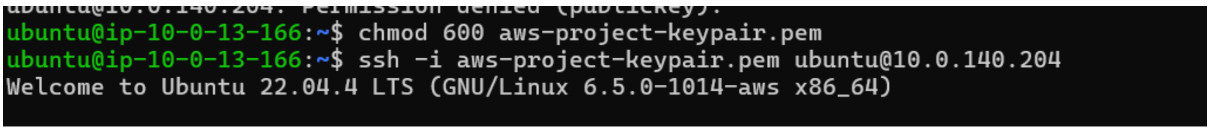
```
C:\Users\Punith\Downloads>scp -i /Users/Punith/Downloads/aws-project-keypair.pem /Users/Punith/Downloads/aws-project-key
pair.pem ubuntu@52.66.24.128:/home/ubuntu
aws-project-keypair.pem                                                                100% 1674     18.0KB/s   00:00

C:\Users\Punith\Downloads>ssh -i aws-project-keypair.pem ubuntu@52.66.24.128
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.5.0-1014-aws x86_64)
```

The keypair is copied. So, now we can access the private instances from inside the bastion-host.



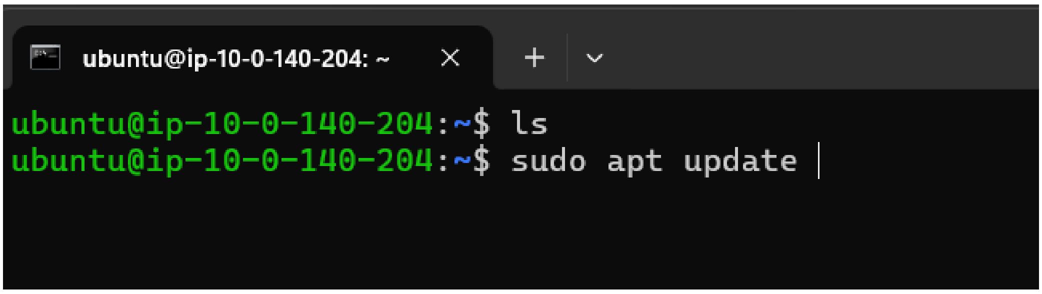
```
ubuntu@ip-10-0-13-166:~$ ls
aws-project-keypair.pem
ubuntu@ip-10-0-13-166:~$
```

Change permissions and ssh to one of the private instance with it's private IP address.



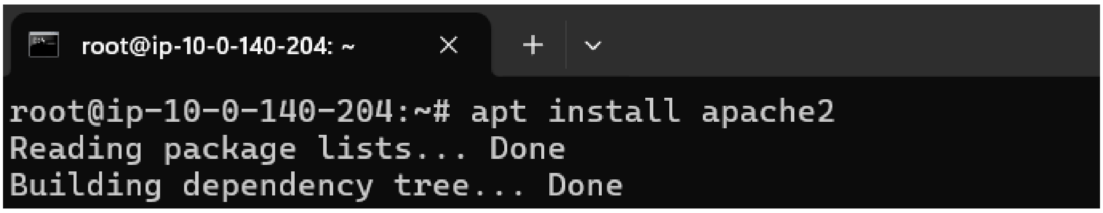
```
ubuntu@10.0.140.204: Permission denied (publickey).
ubuntu@ip-10-0-13-166:~$ chmod 600 aws-project-keypair.pem
ubuntu@ip-10-0-13-166:~$ ssh -i aws-project-keypair.pem ubuntu@10.0.140.204
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.5.0-1014-aws x86_64)
```

SSH login was successful and update the packages.



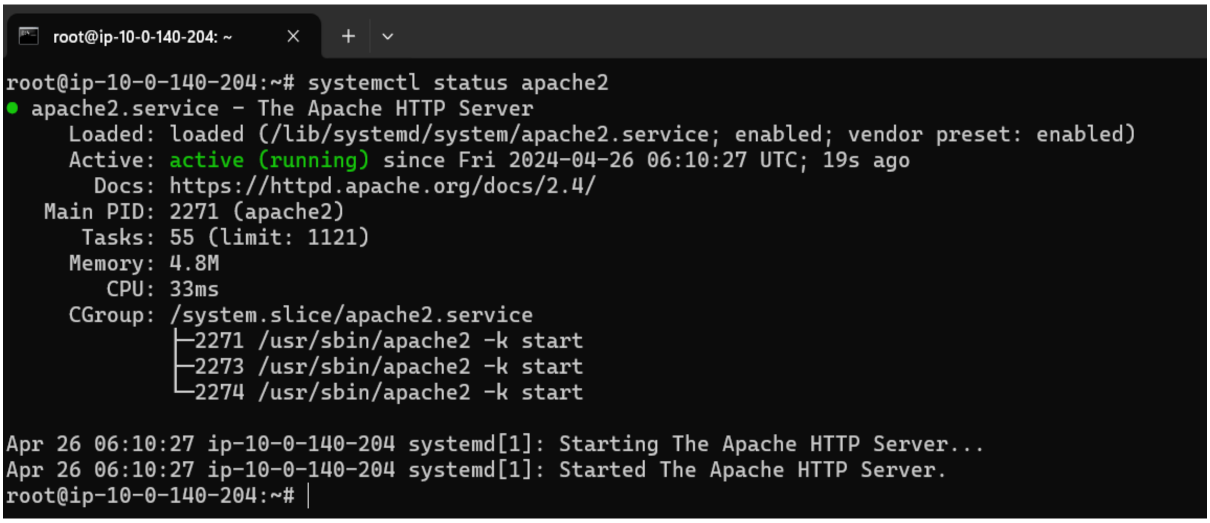
```
ubuntu@ip-10-0-140-204:~$ ls
ubuntu@ip-10-0-140-204:~$ sudo apt update
```

Install apache2 server.



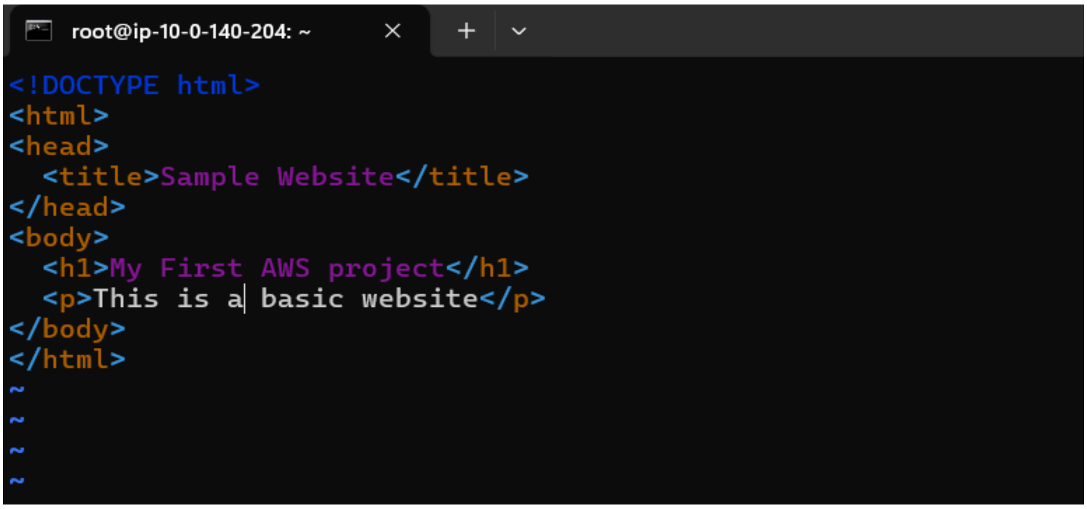
```
root@ip-10-0-140-204:~# apt install apache2
Reading package lists... Done
Building dependency tree... Done
```

Check the status of the server.



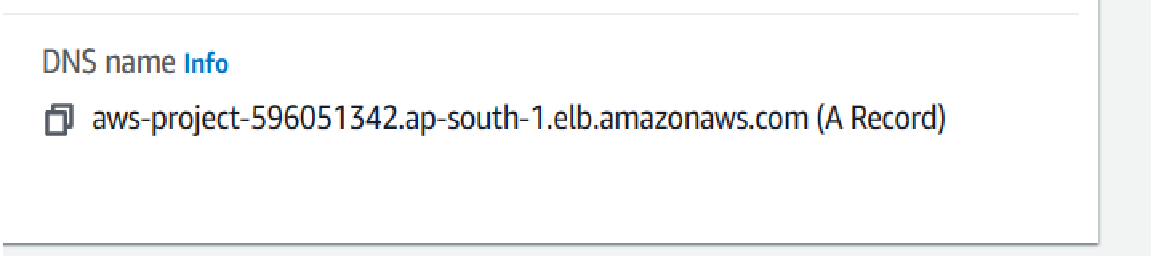
```
root@ip-10-0-140-204:~# systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2024-04-26 06:10:27 UTC; 19s ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2271 (apache2)
      Tasks: 55 (limit: 1121)
     Memory: 4.8M
        CPU: 33ms
     CGroup: /system.slice/apache2.service
             ├─2271 /usr/sbin/apache2 -k start
             ├─2273 /usr/sbin/apache2 -k start
             └─2274 /usr/sbin/apache2 -k start

Apr 26 06:10:27 ip-10-0-140-204 systemd[1]: Starting The Apache HTTP Server...
Apr 26 06:10:27 ip-10-0-140-204 systemd[1]: Started The Apache HTTP Server.
root@ip-10-0-140-204:~# 
```

Enter the following code into the index.html.



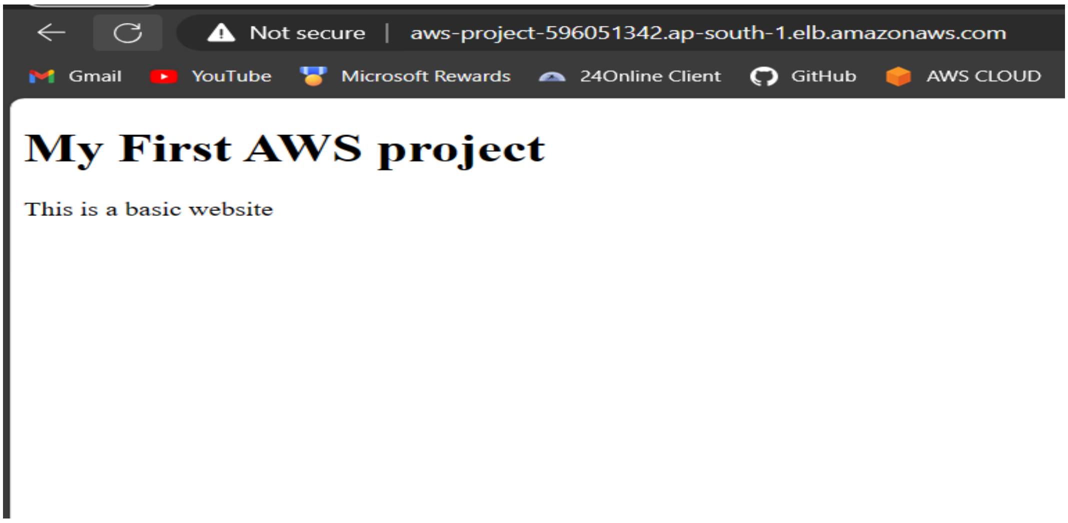
```
<!DOCTYPE html>
<html>
<head>
  <title>Sample Website</title>
</head>
<body>
  <h1>My First AWS project</h1>
  <p>This is a basic website</p>
</body>
</html>
~
~
~
~
```

Copy the DNS name of the loadbalancer and try accessing it.


DNS name Info

aws-project-596051342.ap-south-1.elb.amazonaws.com (A Record)

Successfully accessed loadbalancer and hosted.

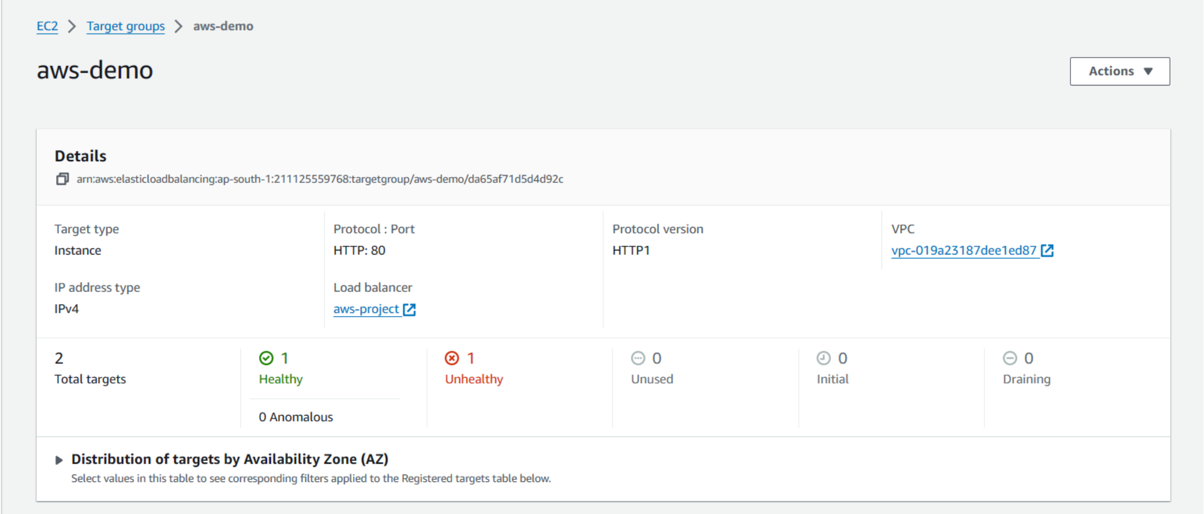


Not secure | aws-project-596051342.ap-south-1.elb.amazonaws.com
Gmail
YouTube
Microsoft Rewards
24Online Client
GitHub
AWS CLOUD

My First AWS project

This is a basic website

One of the instance is healthy. And try installing in another private instance also.



aws-demo

Details

arn:aws:elasticloadbalancing:ap-south-1:211125559768:targetgroup/aws-demo/da65af71d5d4d92c

Target type: Instance

Protocol : Port: HTTP: 80

Protocol version: HTTP1

VPC: vpc-019a23187dee1ed87

IP address type: IPv4

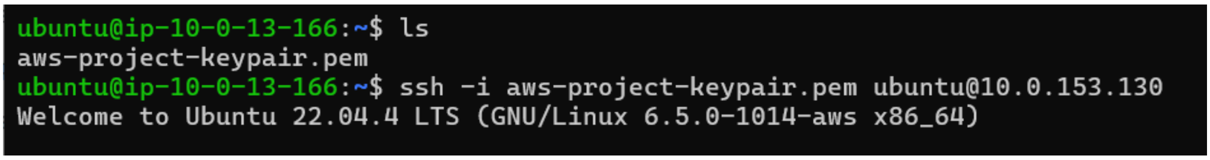
Load balancer: aws-project

2 Total targets; ✓ 1 Healthy; 0 Anomalous; ⊗ 1 Unhealthy; 0 Unused; 0 Initial; 0 Draining

▸ **Distribution of targets by Availability Zone (AZ)**

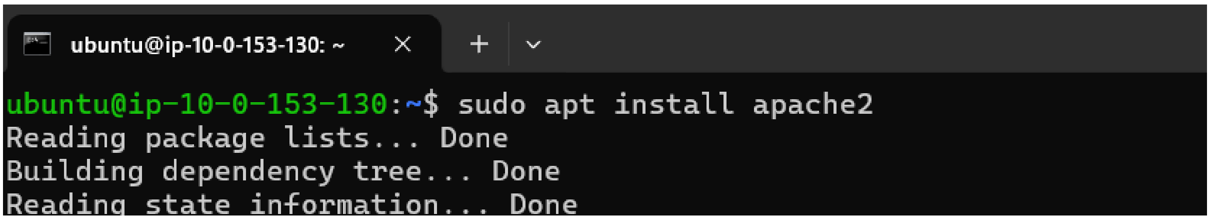
Select values in this table to see corresponding filters applied to the Registered targets table below.

Ssh to the Server2 instance with private IP address.



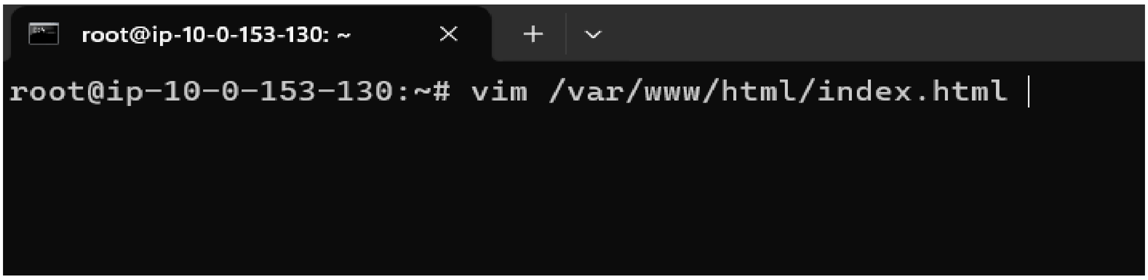
```
ubuntu@ip-10-0-13-166:~$ ls
aws-project-keypair.pem
ubuntu@ip-10-0-13-166:~$ ssh -i aws-project-keypair.pem ubuntu@10.0.153.130
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.5.0-1014-aws x86_64)
```

After ssh to Server2 instance, install apache2.



```
ubuntu@ip-10-0-153-130:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

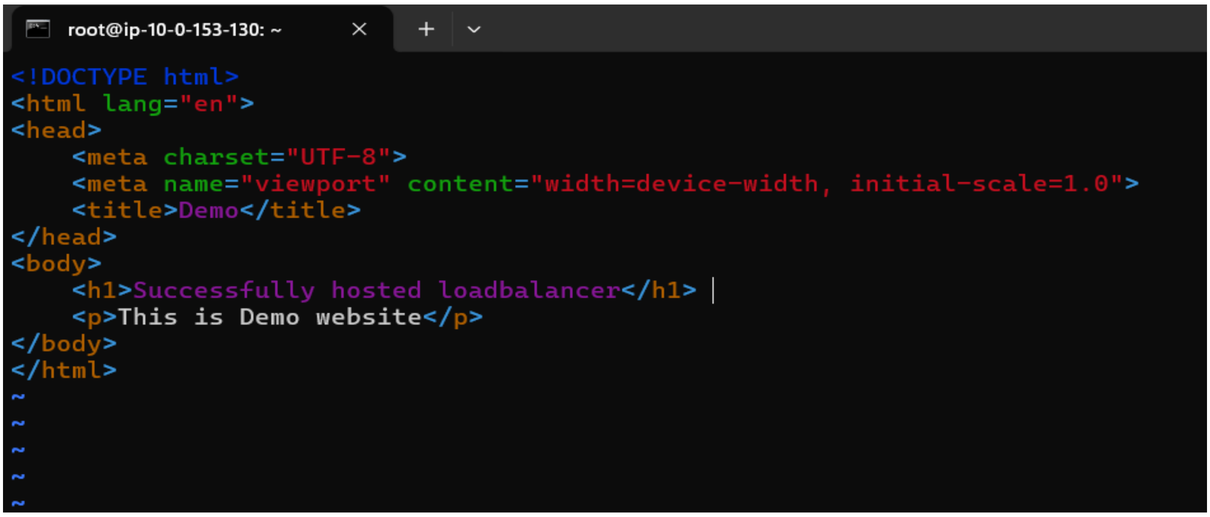
Open the index.html file.



root@ip-10-0-153-130: ~

```
root@ip-10-0-153-130:~# vim /var/www/html/index.html
```

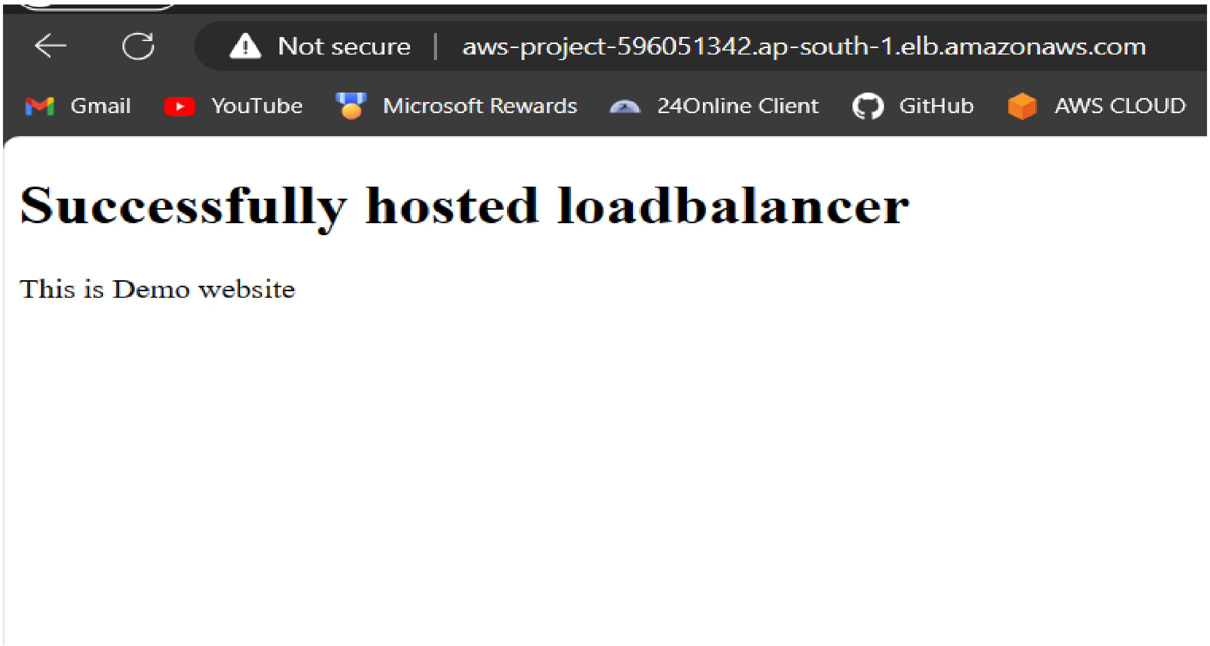
Enter the following code into the file.



root@ip-10-0-153-130: ~

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Demo</title>
</head>
<body>
    <h1>Successfully hosted loadbalancer</h1>
    <p>This is Demo website</p>
</body>
</html>
~
~
~
~
~
```

Whenever we refresh the page, the traffic gets distributed into both the private instances.



Successfully hosted loadbalancer

This is Demo website

MAKE SURE TO DELETE THE SETUP AFTER SUCCESSFUL EXECUTION.

HAPPY LEARNING!