

5.Launching the website using load balancer

VPC

The screenshot shows the AWS VPC dashboard for a specific VPC. The left sidebar contains navigation links for VPC, Subnets, Route tables, Internet gateways, Egress-only internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, NAT gateways, and Peering connections. The main content area displays the details for VPC ID vpc-0b32a7e5eb6371cce, which is in an 'Available' state. Key details include: Block Public Access is Off, DNS hostnames are Disabled, DNS resolution is Enabled, Tenancy is default, DHCP option set is dopt-038b22d07c74ffe27, Main route table is rtb-0d97de91159616dd5, Main network ACL is acl-016f0ac511cf29723, Default VPC is No, IPv4 CIDR is 10.0.0.0/16, IPv6 CIDR is -, Network Address Usage metrics are Disabled, Route 53 Resolver DNS Firewall rule groups are Failed to load rule groups, and the Owner ID is 471112860190. Below the details, there are tabs for Resource map, CIDRs, Flow logs, Tags, and Integrations.

Property	Value
VPC ID	vpc-0b32a7e5eb6371cce
State	Available
Block Public Access	Off
DNS hostnames	Disabled
DNS resolution	Enabled
Tenancy	default
DHCP option set	dopt-038b22d07c74ffe27
Main route table	rtb-0d97de91159616dd5
Main network ACL	acl-016f0ac511cf29723
Default VPC	No
IPv4 CIDR	10.0.0.0/16
IPv6 CIDR	-
Network Address Usage metrics	Disabled
Route 53 Resolver DNS Firewall rule groups	Failed to load rule groups
Owner ID	471112860190

Subnet

The screenshot shows the AWS VPC dashboard for route tables. The left sidebar is the same as the previous screenshot. The main content area displays a list of route tables. The table has columns for Name, Route table ID, Explicit subnet associations, Edge associations, and Metrics. Two route tables are selected: sai-private-RT and sai-public-RT, both associated with 2 subnets. Below the table, there is a summary of the selected route tables: rtb-08839bba789d44d07, rtb-03b56c019eb11f0c.

Name	Route table ID	Explicit subnet associations	Edge associations	Metrics
sai-private-RT	rtb-08839bba789d44d07	2 subnets	-	Nk
-	rtb-0d97de91159616dd5	-	-	Ye
sai-public-RT	rtb-03b56c019eb11f0c	2 subnets	-	Nk
-	rtb-0a6ffc28ece5ab0c0b	-	-	Ye
ullas-private-routetable	rtb-0b81579d947a2ac6e	-	-	Nk

Internet Gateway

The screenshot shows the AWS Management Console interface for an Internet Gateway. The left sidebar lists navigation options under 'Virtual private cloud' and 'Security'. The main content area displays the details for the Internet Gateway 'igw-08b082ac439f468b7' / 'sai-igw'. The 'Details' section shows the Internet gateway ID, State (Attached), VPC ID, and Owner. The 'Tags' section shows a single tag with the key 'Name' and value 'sai-igw'.

Internet Gateway Details:

- Internet gateway ID: `igw-08b082ac439f468b7`
- State: **Attached**
- VPC ID: `vpc-0b32a7a5efb6371c8a` (Link to VPC)
- Owner: `471112860190`

Tags:

Key	Value
Name	sai-igw

Instances

The screenshot shows the AWS Management Console interface for EC2 instances. The left sidebar lists navigation options under 'Instances' and 'Images'. The main content area displays a list of instances with columns for Name, Instance ID, Instance state, Instance type, Status check, and Alarm status. Three instances are selected, and the 'Monitoring' tab is active.

Instances List:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
sai-private1-se...	i-007fa7d0b8425860	Running	t2.micro	2/2 checks passed	View alarms +
jeewan-jump-s...	i-0e380d7ba157bc71e	Running	t2.micro	2/2 checks passed	View alarms +
ullas-jump-ser...	i-005aad44ad9413f6a	Running	t3.micro	3/3 checks passed	View alarms +
sai-private-ser...	i-02d306e63a0270508	Running	t3.micro	3/3 checks passed	View alarms +
jeewan-private...	i-0e2b8eadd71502d75	Running	t3.micro	3/3 checks passed	View alarms +
sai-jump-server	i-0295879dadf232fc0	Running	t2.micro	2/2 checks passed	View alarms +

3 instances selected

Monitoring

Load Balancer

The screenshot displays the AWS Management Console for the 'neha-LB01' load balancer. The left sidebar shows navigation options under 'Elastic Block Store', 'Network & Security', and 'Load Balancing'. The main content area is titled 'neha-LB01' and includes an 'Actions' button. The 'Details' section provides the following information:

- Load balancer type:** Application
- Status:** Active
- VPC:** vpc-0b32a7e5eb6371cce
- Load balancer IP address type:** IPv4
- Scheme:** Internet-facing
- Hosted zone:** Z2P70J7HTTTPLU
- Availability Zones:** subnet-00fe557575dc4b073 (sa-east-1a), subnet-0f137fd15664ec3e (sa-east-1b)
- Date created:** December 12, 2024, 10:28 (UTC+05:30)
- Load balancer ARN:** arn:aws:elasticloadbalancing:sa-east-1:471112860190:loadbalancer/app/neha-LB01/314cdeecf8ee00ee
- DNS name:** neha-LB01-1858029303.sa-east-1.elb.amazonaws.com (A Record)

Target Groups

The screenshot displays the AWS Management Console for the 'Target groups' section. The left sidebar shows navigation options under 'Instances', 'Images', and 'Elastic Block Store'. The main content area is titled 'Target groups (1/5)' and includes a 'Create target group' button. A table lists the target groups:

Name	ARN	Port	Protocol	Target type
viksha-tg01	arn:aws:elasticloadbalancing:sa-east-1:471112860190:targetgroup/viksha-tg01/314cdeecf8ee00ee	80	HTTP	Instance
sakshith5-Tg1	arn:aws:elasticloadbalancing:sa-east-1:471112860190:targetgroup/sakshith5-Tg1/314cdeecf8ee00ee	80	HTTP	Instance

Below the table, the details for the 'Target group: SN-TG01' are shown:

- IP address type:** IPv4
- Load balancer:** neha-LB01
- 2 Total targets**
- Health:** 2 Healthy, 0 Unhealthy
- 0 Anomalous**
- 0 Unused**
- 0 Initial**
- 0 Draining**

Command prompt settings

```
Microsoft Windows [Version 10.0.19H45.671]
(c) Microsoft Corporation. All rights reserved.

C:\Users\neha.DESKTOP-HP7JT2I>cd Downloads

C:\Users\neha.DESKTOP-HP7JT2I\Downloads>ssh -i "sn-keypair.pem" ec2-user@54.219.176.48
The authenticity of host '54.219.176.48 (54.219.176.48)' can't be established.
ECDSA key fingerprint is SHA256:SIXkXpDgVweFiBjU5kat9cniGGOYblYvAUAJtQwL7qZ.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '54.219.176.48' (ECDSA) to the list of known hosts.
```

A terminal window displaying the output of the command `wget https://aws.amazon.com/linux/amazon-linux-2023`. The browser's graphical user interface is visible in the background, showing the Amazon Linux 2023 download page. A dashed arrow points from the URL in the terminal to the corresponding link on the webpage.

```
#
####
#####
###
#/#> https://aws.amazon.com/linux/amazon-linux-2023
#/#>
```

```
[ec2-user@ip-10-0-1-69 ~]$ sudo su
[root@ip-10-0-1-69 ec2-user]# cd
[root@ip-10-0-1-69 ~]# vi sn-private-server1.pem
[root@ip-10-0-1-69 ~]# chmod 400 "sn-private-server1.pem"
[root@ip-10-0-1-69 ~]# ssh -i "sn-private-server1.pem" ec2-user@10.0.3.19
The authenticity of host '10.0.3.19 (10.0.3.19)' can't be established.
ED25519 key fingerprint is SHA256:fuzi3ubobd7ewJokRlTYbBDWKLZW/NLRPFy4IRikTHSH.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.3.19' (ED25519) to the list of known hosts:
```

A second terminal window showing the continuation of the SSH connection setup. It displays the same warning message about adding the new host to the known hosts list. In the background, another instance of the Amazon Linux 2023 download page is shown, with a dashed arrow pointing from the terminal's URL to the webpage.

```
#
####
#####
###
#/#> https://aws.amazon.com/linux/amazon-linux-2023
#/#>
```

```
[ec2-user@ip-10-0-3-19 ~]$ sudo su
```

```

apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
httpd-2.4.62-1.amzn2023.x86_64
httpd-filesystem-2.4.62-1.amzn2023.noarch
libbrotli-1.0.9-4.amzn2023.0.2.x86_64
mod_http-2.0.27-1.amzn2023.0.3.x86_64

generic-logos-httpsd-18.0.8-12.amzn2023.0.3.noarch
httpd-core-2.4.62-1.amzn2023.x86_64
httpd-tools-2.4.62-1.amzn2023.x86_64
mailcap-2.1.49-3.amzn2023.0.3.noarch
mod_lua-2.4.62-1.amzn2023.x86_64

Complete!
[root@ip-10-0-3-19 ~]# cd /var/www/html
[root@ip-10-0-3-19 html]# vi index.html
[root@ip-10-0-3-19 html]# [root@ip-10-0-3-19 html]# systemctl start httpd
[root@ip-10-0-3-19 html]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-10-0-3-19 html]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Active: active (running) since Thu 2024-12-12 18:17:51 UTC; 26s ago
     Docs: man:httpd.service(8)
  Main PID: 26180 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"
    Tasks: 177 (limit: 1111)
   Memory: 15.0M
      CPU: 68ms
   CGroup: /system.slice/httpd.service
           └─26180 /usr/sbin/httpd -DFOREGROUND
             └─26181 /usr/sbin/httpd -DFOREGROUND
               └─26182 /usr/sbin/httpd -DFOREGROUND
                 └─26183 /usr/sbin/httpd -DFOREGROUND
                   └─26184 /usr/sbin/httpd -DFOREGROUND

Dec 12 18:17:51 ip-10-0-3-19.us-west-1.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Dec 12 18:17:51 ip-10-0-3-19.us-west-1.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Dec 12 18:17:51 ip-10-0-3-19.us-west-1.compute.internal httpd[26180]: Server configured, listening on: port 80
[root@ip-10-0-3-19 html]# exit
exit
[ec2-user@ip-10-0-3-19 ~]$ exit
logout
Connection to 10.0.3.19 closed.
[root@ip-10-0-1-69 ~]# ssh -i "sn-private-server1.pem" ec2-user@10.0.4.59
The authenticity of host '10.0.4.59 (10.0.4.59)' can't be established.
ED25519 key fingerprint is SHA256:81KA7jwew/7iIRb4vy6dxCSkn/S7A4cr90eIGX100Cj5.
This key is not known by any other names

```

```

Installed:
apr-1.7.2-2.amzn2023.0.2.x86_64          apr-util-1.6.3-1.amzn2023.0.1.x86_64
apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64  generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
httpd-2.4.62-1.amzn2023.x86_64             httpd-core-2.4.62-1.amzn2023.x86_64
httpdfilesystem-2.4.62-1.amzn2023.noarch    httpd-tools-2.4.62-1.amzn2023.x86_64
libbrotli-1.0.9-4.amzn2023.0.2.x86_64      mailcap-2.1.49-3.amzn2023.0.3.noarch
mod_http2-2.0.27-1.amzn2023.0.3.x86_64     mod_lua-2.4.62-1.amzn2023.x86_64

Complete!
[root@ip-10-0-4-59 ~]# cd /usr/www/html
[root@ip-10-0-4-59 html]# vi index.html
[root@ip-10-0-4-59 html]# [root@ip-10-0-4-59 html]# systemctl start httpd
[root@ip-10-0-4-59 html]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-10-0-4-59 html]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Active: active (running) since Thu 2024-12-12 18:21:16 UTC; 34s ago
     Docs: man:httpd.service(8)
   Main PID: 26125 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec:  0 B/sec"
    Tasks: 177 (limit: 1111)
   Memory: 13.0M
      CPU: 71ms
   CGroup: /system.slice/httpd.service
           └─26125 /usr/sbin/httpd -DFOREGROUND
             └─26126 /usr/sbin/httpd -DFOREGROUND
               └─26127 /usr/sbin/httpd -DFOREGROUND
                 └─26128 /usr/sbin/httpd -DFOREGROUND
                   └─26129 /usr/sbin/httpd -DFOREGROUND

Dec 12 18:21:16 ip-10-0-4-59.us-west-1.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Dec 12 18:21:16 ip-10-0-4-59.us-west-1.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Dec 12 18:21:16 ip-10-0-4-59.us-west-1.compute.internal httpd[26125]: Server configured, listening on: port 80
[root@ip-10-0-4-59 html]# client_loop: send disconnect: Connection reset

C:\Users\neha.DESKTOP-HP73721\Downloads>

```

Final Result





This is server 2This is server 2