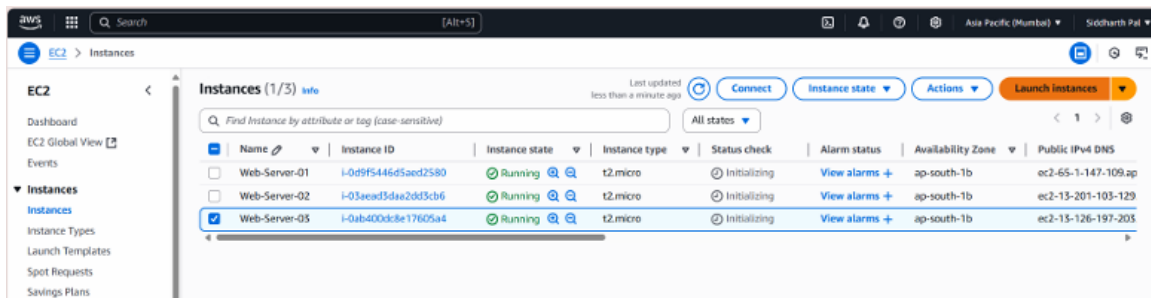


AWS Lab – Hosting Web Pages on Multiple EC2 Instances + Load Balancing

By Ayush Singh || 22BCSE63

Step 1: Launch Web Server 1 (web-server-01)

- Click Launch Instance
- Instance Name: web-server-01
- AMI: Amazon Linux 2 (Free Tier Eligible)
- Instance Type: t2.micro
- Key Pair: Choose existing or create new
- Network Settings: Allow HTTP and HTTPS traffic (SSH is checked by default)
- Leave all other settings default
- Click Launch Instance



Step 2: Connect to web-server-01 via SSH

- Use MobaXterm or terminal:
- `ssh -i <your-key.pem> ec2-user@<Public-IP-of-web-server-01>`

Step 3: Install and Start Apache on web-server-01

- `sudo su`
- `cd`
- `yum install httpd -y`
- `cd /var/www/html`
- `echo "welcome to the webserver demo" > index.html`
- `systemctl start httpd`
- `systemctl enable httpd`

```

Installing      : httpd-core-2.4.62-1.amzn2023.x86_64      8/12
Installing      : mod_httpd-2.0.27-1.amzn2023.0.3.x86_64  9/12
Installing      : mod_lua-2.4.62-1.amzn2023.x86_64       10/12
Installing      : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch 11/12
Installing      : httpd-2.4.62-1.amzn2023.x86_64         12/12
Running scriptlet: httpd-2.4.62-1.amzn2023.x86_64       12/12
Verifying       : apr-1.7.5-1.amzn2023.0.4.x86_64        1/15
Verifying       : apr-util-1.6.3-1.amzn2023.0.1.x86_64   2/15
Verifying       : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64 3/15
Verifying       : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch 4/15
Verifying       : httpd-2.4.62-1.amzn2023.x86_64        5/15
Verifying       : httpd-core-2.4.62-1.amzn2023.x86_64    6/15
Verifying       : httpd-filesystem-2.4.62-1.amzn2023.noarch 7/15
Verifying       : httpd-tools-2.4.62-1.amzn2023.x86_64   8/15
Verifying       : libbrotli-1.0.9-4.amzn2023.0.2.x86_64  9/15
Verifying       : mailcap-2.1.49-3.amzn2023.0.3.noarch   10/15
Verifying       : mod_httpd-2.0.27-1.amzn2023.0.3.x86_64 11/15
Verifying       : mod_lua-2.4.62-1.amzn2023.x86_64      12/15

Installed:
apr-1.7.5-1.amzn2023.0.4.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  httpd-filesystem-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_httpd-2.0.27-1.amzn2023.0.3.x86_64      mod_lua-2.4.62-1.amzn2023.x86_64

complete!
root@ip-172-31-10-70 ~# cd /var/www/html
root@ip-172-31-10-70 html# echo "Hello from server 1!" > index.html
root@ip-172-31-10-70 html# systemctl start httpd
root@ip-172-31-10-70 html# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
root@ip-172-31-10-70 html#

```

Step 4: Access Web Page from Browser

- Open browser and visit: <http://<Public-IP-of-web-server-01>>

Welcome to the webserver demo

Step 5: Launch Two More Web Servers (web-server-02 and web-server-03)

- Click Launch Instance
- Number of Instances: 2
- Instance Names: web-server-02, web-server-03
- AMI: Amazon Linux 2
- Instance Type: t2.micro
- Network Settings: Allow HTTP and HTTPS traffic
- Launch the instances
- Rename instances via "Actions > Manage Tags" on the EC2 dashboard if needed

Step 6: Configure web-server-02

- Connect via SSH: `ssh -i <your-key.pem> ec2-user@<Public-IP-of-web-server-02>`
- `sudo su`
- `cd`
- `yum install httpd -y`
- `cd /var/www/html`
- `echo "Hello from server 2!" > index.html`
- `systemctl start httpd`
- `systemctl enable httpd`

```

Amazon Linux 2023

https://aws.amazon.com/linux/amazon-linux-2023

et login: Fri Jun 20 18:33:59 2025 from 13.223.177.4
ec2-user@ip-172-31-10-70 ~]$ sudo su
root@ip-172-31-10-70 ec2-user# cd
root@ip-172-31-10-70 ~# yum install httpd -y
et metadata expiration check: 0:02:14 ago on Fri Jun 20 18:34:28 2025.
Package httpd-2.4.62-1.amzn2023.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
root@ip-172-31-10-70 ~# yum install httpd -y
et metadata expiration check: 0:02:45 ago on Fri Jun 20 18:34:28 2025.
Package httpd-2.4.62-1.amzn2023.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
root@ip-172-31-10-70 ~# echo "Hello from server 2!" > index.html
root@ip-172-31-10-70 ~# systemctl start httpd
root@ip-172-31-10-70 ~# systemctl enable httpd
root@ip-172-31-10-70 ~#

```

Step 7: Configure web-server-03

- Connect via SSH: `ssh -i <your-key.pem> ec2-user@<Public-IP-of-web-server-03>`
- `sudo su`
- `cd`
- `yum install httpd -y`
- `cd /var/www/html`
- `echo "Hello from server 3!" > index.html`
- `systemctl start httpd`
- `systemctl enable httpd`

```

Installing      : httpd-core-2.4.62-1.amzn2023.x86_64          8/12
Installing      : mod_http2-2.0.27-1.amzn2023.0.3.x86_64     9/12
Installing      : mod_lua-2.4.62-1.amzn2023.x86_64          10/12
Installing      : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch 11/12
Installing      : httpd-2.4.62-1.amzn2023.x86_64            12/12
Running scriptlet: httpd-2.4.62-1.amzn2023.x86_64            12/12
Verifying       : apr-1.7.5-2.amzn2023.0.4.x86_64           1/12
Verifying       : apr-util-1.6.3-1.amzn2023.0.1.x86_64       2/12
Verifying       : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64 3/12
Verifying       : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch 4/12
Verifying       : httpd-2.4.62-1.amzn2023.x86_64            5/12
Verifying       : httpd-core-2.4.62-1.amzn2023.x86_64        6/12
Verifying       : httpd-filesystem-2.4.62-1.amzn2023.noarch   7/12
Verifying       : httpd-tools-2.4.62-1.amzn2023.x86_64       8/12
Verifying       : libbrotli-1.0.9-4.amzn2023.0.2.x86_64      9/12
Verifying       : mailcap-2.1.49-3.amzn2023.0.3.noarch       10/12
Verifying       : mod_http2-2.0.27-1.amzn2023.0.3.x86_64    11/12
Verifying       : mod_lua-2.4.62-1.amzn2023.x86_64          12/12

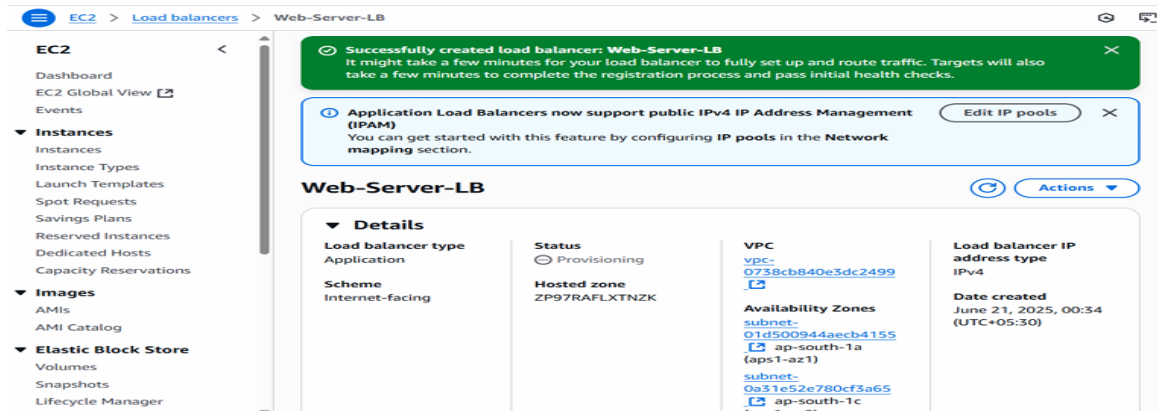
Installed:
apr-1.7.5-1.amzn2023.0.4.x86_64      apr-util-1.6.3-1.amzn2023.0.1.x86_64  apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64  generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
httpd-2.4.62-1.amzn2023.x86_64      httpd-core-2.4.62-1.amzn2023.x86_64  httpd-filesystem-2.4.62-1.amzn2023.noarch      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-172-31-6-65 ~]# cd /var/www/html
[root@ip-172-31-6-65 html]# echo "Hello from server 3!" > index.html
[root@ip-172-31-6-65 html]# systemctl start httpd
[root@ip-172-31-6-65 html]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-6-65 html]#

```

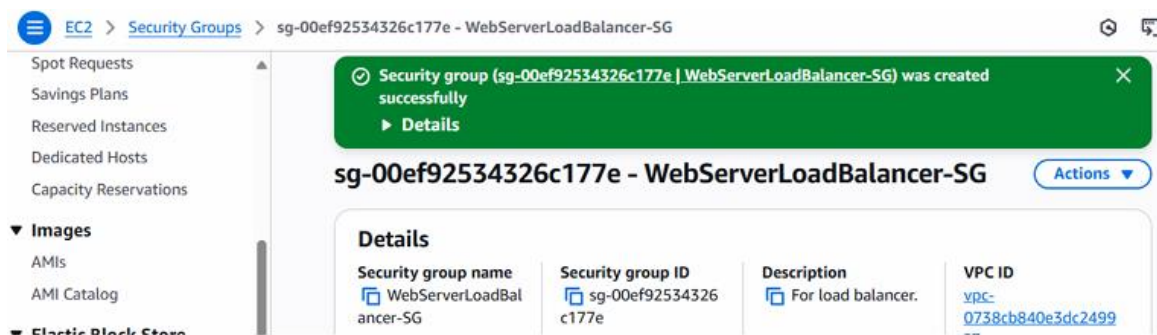
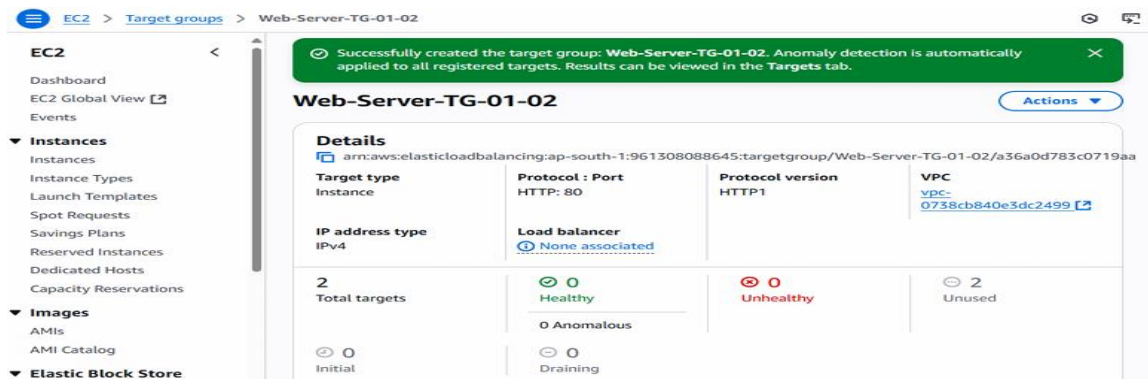
Step 8: Set Up Application Load Balancer (ALB)

- Create Application Load Balancer
- Name: Web-Server-LB, Scheme: Internet-facing, IP type: IPv4
- Choose VPC used by web servers
- Select 2–3 subnets from different Availability Zones



Step 9: Create a Target Group

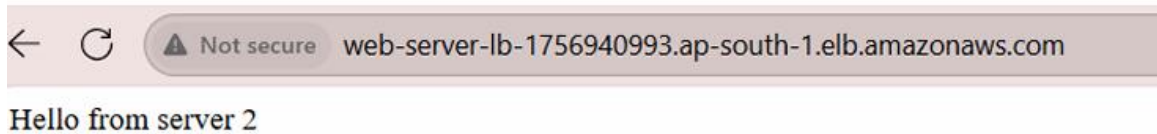
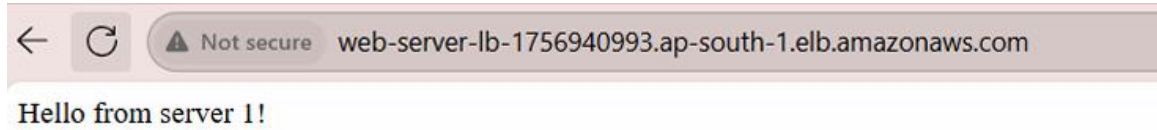
- Target Type: Instances
- Name: Web-Server-TG-01-02
- VPC: Same as web servers
- Select instances: web-server-01, web-server-02
- Click Include as Pending, then Create Target Group



Step 10: Access Website via Load Balancer DNS

- Go to EC2 Dashboard > Load Balancers

- • Copy Load Balancer DNS name
- • Paste in browser: <http://<Load-Balancer-DNS>>



Step 11: Setup SSL Certificate via ACM

- • Request Public Certificate in ACM
- • Enter domain: *.yourdomain.com and yourdomain.com
- • Validate via Route 53 by adding CNAME records

Step 12: Add HTTPS Listener to Load Balancer

- • Go to Load Balancer settings → Add Listener
- • Protocol: HTTPS, Port: 443
- • Forward to: Web-Server-TG-01-02
- • Use SSL Certificate from ACM

Step 13: (Optional) Redirect HTTP to HTTPS

- • Edit HTTP Listener rules (port 80)
- • Change action to Redirect to HTTPS (port 443) with status code HTTP_301