

Made By

Satish Rajurkar

Vaibhav Sarode

**016 Practical - Load Balancer to route traffic according to Headers of
Applicatins - 21 Feb 2022**

We have to create 4 instances in two different regions.

The screenshot shows the AWS Launch Instance Wizard Step 3: Configure Instance Details. The user has specified 2 instances. The network selected is `vpc-0d530da491583e01c (default)`, and the subnet selected is `subnet-04bb880c3944e9026 | Default in us-east-2a`. The subnet field is highlighted with a red box. Other settings include requesting spot instances, enabling auto-assigned public IP, and configuring DNS hostnames.

Number of instances launch into Auto Scaling Group [\[i\]](#)

You may want to consider launching these instances into an Auto Scaling Group to help you maintain application availability and for easy scaling in the future. [Learn how Auto Scaling can help your application stay healthy and cost effective.](#)

Purchasing option Request Spot instances

Network [\[i\]](#) `vpc-0d530da491583e01c (default)` [\[C\]](#) Create new VPC

Subnet [\[i\]](#) `subnet-04bb880c3944e9026 | Default in us-east-2a` [\[C\]](#) Create new subnet
4091 IP Addresses available

Auto-assign Public IP [\[i\]](#) Use subnet setting (Enable)

Hostname type [\[i\]](#) Use subnet setting (IP name)

DNS Hostname [\[i\]](#)
 Enable IP name IPv4 (A record) DNS requests
 Enable resource-based IPv4 (A record) DNS requests
 Enable resource-based IPv6 (AAAA record) DNS requests

Cancel Previous Review and Launch Next: Add Storage

Feedback English (US) © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 9:53 AM 2/24/2022 1

- Here we are creating 2 instances in zone a

Launch instance wizard | EC2 Manager Billing Management Console

us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard:

Search for services, features, blogs, docs, and more [Alt+S]

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: Create a new security group Select an existing security group

Security group name: ohio-security-group-http-80

Description: ohio-security-group-http-80

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Anywhere	0.0.0.0/0, ::/0 e.g. SSH for Admin Desktop
HTTP	TCP	80	Anywhere	0.0.0.0/0, ::/0 e.g. SSH for Admin Desktop

Add Rule

Warning
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Cancel Previous Review and Launch

Feedback English (US) © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences

ENG IN 9:55 AM 2/24/2022 1

- Adding **HTTP 80** into inbound rule
- Launch the instances.

The screenshot shows the AWS Launch Instance Wizard at Step 3: Configure Instance Details. The user has selected to launch 2 instances. A callout box highlights the subnet selection, which is set to a specific subnet in the us-east-2b region. The 'Review and Launch' button is prominently displayed at the bottom.

Number of instances launch into Auto Scaling Group

You may want to consider launching these instances into an Auto Scaling Group to help you maintain application availability and for easy scaling in the future. [Learn how Auto Scaling can help your application stay healthy and cost effective.](#)

Purchasing option Request Spot instances

Network [Create new VPC](#)

Subnet [Create new subnet](#)
4091 IP Addresses available

Auto-assign Public IP

Hostname type

DNS Hostname Enable IP name IPv4 (A record) DNS requests
 Enable resource-based IPv4 (A record) DNS requests
 Enable resource-based IPv6 (AAAA record) DNS requests

Cancel Previous Review and Launch Next: Add Storage

Feedback English (US) © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 10:00 AM 2/24/2022 1

- Here again we are creating 2 instances in zone b

Launch instance wizard | EC2 Manager | Billing Management Console | (2) WhatsApp | us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard:

Search for services, features, blogs, docs, and more [Alt+S]

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: Create a new security group Select an existing security group

Security Group ID	Name	Description	Actions
sg-005a9761f1028d93f	default	default VPC security group	Copy to new
sg-0151153ca6adc0c8d	ohio-instance	ohio-instance	Copy to new
sg-03ca72a52fa7e0e71	ohio-security-group-http-80	ohio-security-group-http-80	Copy to new

Inbound rules for sg-03ca72a52fa7e0e71 (Selected security groups: sg-03ca72a52fa7e0e71)

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	0.0.0.0/0	
HTTP	TCP	80	::/0	
SSH	TCP	22	0.0.0.0/0	

Cancel Previous Review and Launch

Feedback English (US) © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 10:00 AM 2/24/2022 1

- Here we are selecting our existing security group which is created at the time of launching previous instances
- Launch the instances

The screenshot shows the AWS EC2 Instances page. The left sidebar lists various EC2-related services like EC2 Dashboard, Global View, Events, Tags, Limits, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, and Elastic Block Store. The main content area displays a table titled 'Instances (4)'. The table has columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IP. The data is as follows:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
payment-A	i-08c357f46e8f92409	Running	t2.micro	2/2 checks passed	No alarms	us-east-2a	ec2-3-14
billing-A	i-04275796ed33936b1	Running	t2.micro	2/2 checks passed	No alarms	us-east-2a	ec2-18-1
payment-B	i-012e58119dd28f18b	Running	t2.micro	2/2 checks passed	No alarms	us-east-2b	ec2-18-1
billing-B	i-0fdfbd5661141b447	Running	t2.micro	2/2 checks passed	No alarms	us-east-2b	ec2-3-14

A modal window titled 'Select an instance' is open at the bottom, showing the same list of instances.

- Here we have created all instances.

- Now next step is to install **httpd** server into each instance. So use following commands.
- sudo su –
- yum install httpd –y
- yum httpd start
- yum httpd status

After installing **httpd** server create index.html file into each instances

Now we will login into each instance and create index.html homepage file and particular directory with file as well.

<h1>This is homepage 1</h1>

Cd /var/www/html

<h1>This is homepage 2</h1>

Cd /var/www/html

<h1>This is homepage 3</h1>

Cd /var/www/html

<h1>This is homepage 4</h1>

Cd /var/www/html

<h1>This is Payment Page A</h1>

Cd /var/www/html/payments

<h1>This is Payment Page B</h1>

Cd /var/www/html/payments

<h1>This is Billing Page A</h1>

Cd /var/www/html/billings

<h1>This is Billing Page B</h1>

Cd /var/www/html/billings

So lets create index.html files

```
root@ip-172-31-2-162:~# cd desktop/keys
root@ip-172-31-2-162:~# ssh -i "ohio-instance.pem" ec2-user@ec2-3-145-164-197.us-east-2.compute.amazonaws.com
Warning: Permanently added 'ec2-3-145-164-197.us-east-2.compute.amazonaws.com' (RSA) to the list of known hosts.
X11 forwarding request failed on channel 0

[ec2-user@ip-172-31-2-162 ~]$ sudo su -
[root@ip-172-31-2-162 ~]# cd /
[root@ip-172-31-2-162 /]# yum install httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
Resolving Dependencies
--> Running transaction check
--> Package httpd.x86_64 0:2.4.52-1.amzn2 will be installed
--> Processing Dependency: httpd-tools = 2.4.52-1.amzn2 for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem = 2.4.52-1.amzn2 for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.52-1.amzn2.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.52-1.amzn2.x86_64
--> Running transaction check
--> Package apr.x86_64 0:1.7.0-9.amzn2 will be installed

| 3.7 kB 00:00:00

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net
```

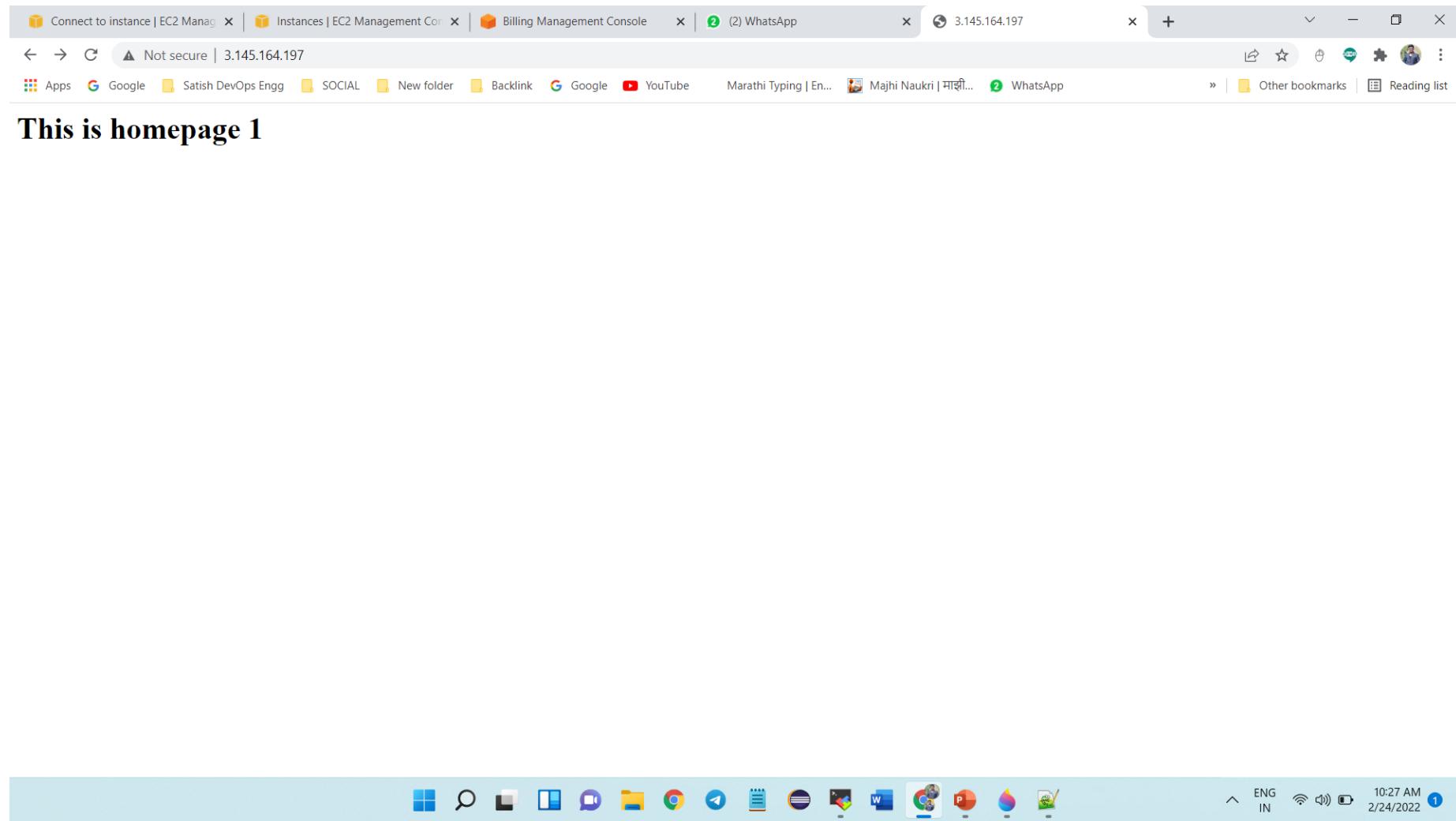
- For instance – 1 in zone a

```
[root@ip-172-31-2-162 ~]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-2-162 ~]#
[root@ip-172-31-2-162 ~]# cd /var/www/html
[root@ip-172-31-2-162 html]# cat >> index.html
<h1>This is homepage 1</h1>
[root@ip-172-31-2-162 html]#
[root@ip-172-31-2-162 html]# mkdir payments
[root@ip-172-31-2-162 html]#
[root@ip-172-31-2-162 html]# cd payments
[root@ip-172-31-2-162 payments]#
[root@ip-172-31-2-162 payments]# cat >> index.html
<h1>This is Payment Page A</h1>
[root@ip-172-31-2-162 payments]#
[root@ip-172-31-2-162 payments]#
[root@ip-172-31-2-162 payments]#
```

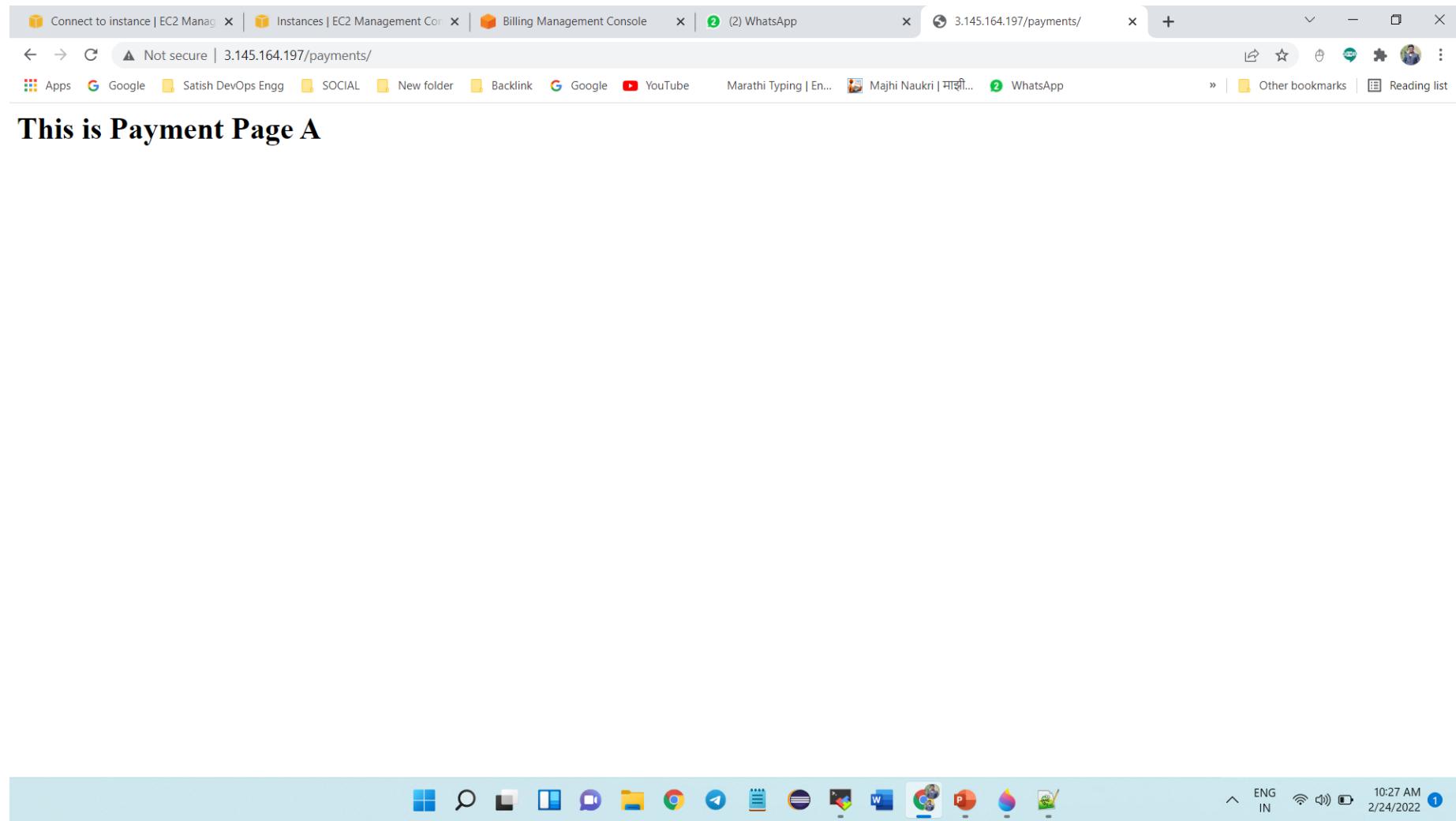
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

ENG IN 10:22 AM 2/24/2022

- Create **index.html** file using above commands



- Check by using public ip



- Check by using public ip with header

- For instance – 2 in zone a

root@ip-172-31-5-7:/

Terminal Sessions View Xserver Tools Games Settings Macros Help

Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help

X server Exit

24/02/2022 10:27:07 /home/mobaxterm/desktop/keys ssh -i "ohio-instance.pem" ec2-user@ec2-18-118-1-1
22.us-east-2.compute.amazonaws.com
Warning: Permanently added 'ec2-18-118-1-122.us-east-2.compute.amazonaws.com' (RSA) to the list of known hosts.
X11 forwarding request failed on channel 0

Amazon Linux 2 AMI

<https://aws.amazon.com/amazon-linux-2/>
8 package(s) needed for security, out of 14 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-5-7 ~]\$ sudo su -
[root@ip-172-31-5-7 ~]#
[root@ip-172-31-5-7 ~]# cd /
[root@ip-172-31-5-7 /]#
[root@ip-172-31-5-7 /]# yum install httpd -y

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

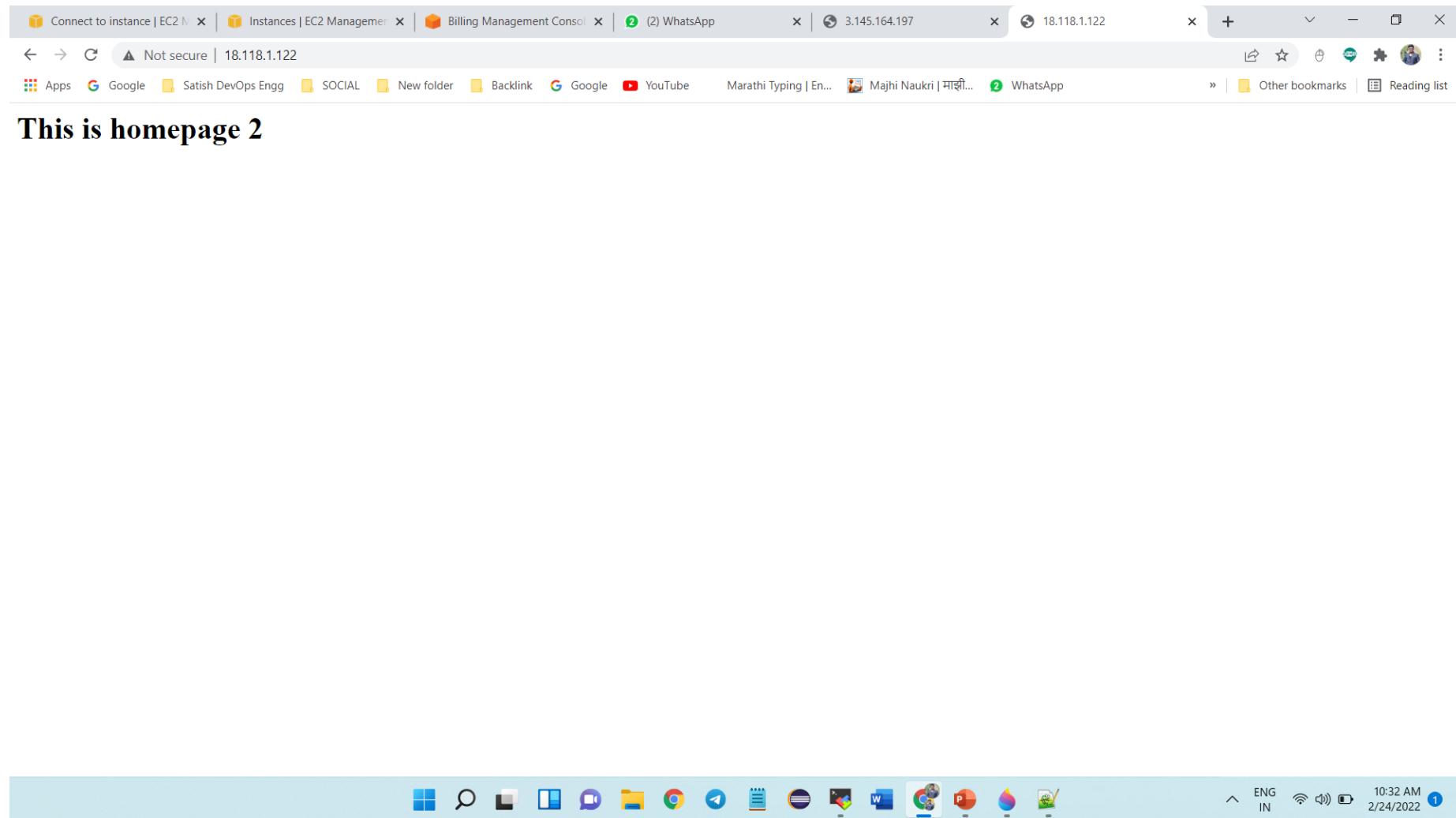
ENG IN 10:29 AM 2/24/2022 1

The screenshot shows a terminal window in MobaXterm with the following session path: root@ip-172-31-5-7:/var/www/html/billings. The terminal window has several tabs open, with the current tab showing a command-line session. The session starts with the user becoming root and then performing the following actions:

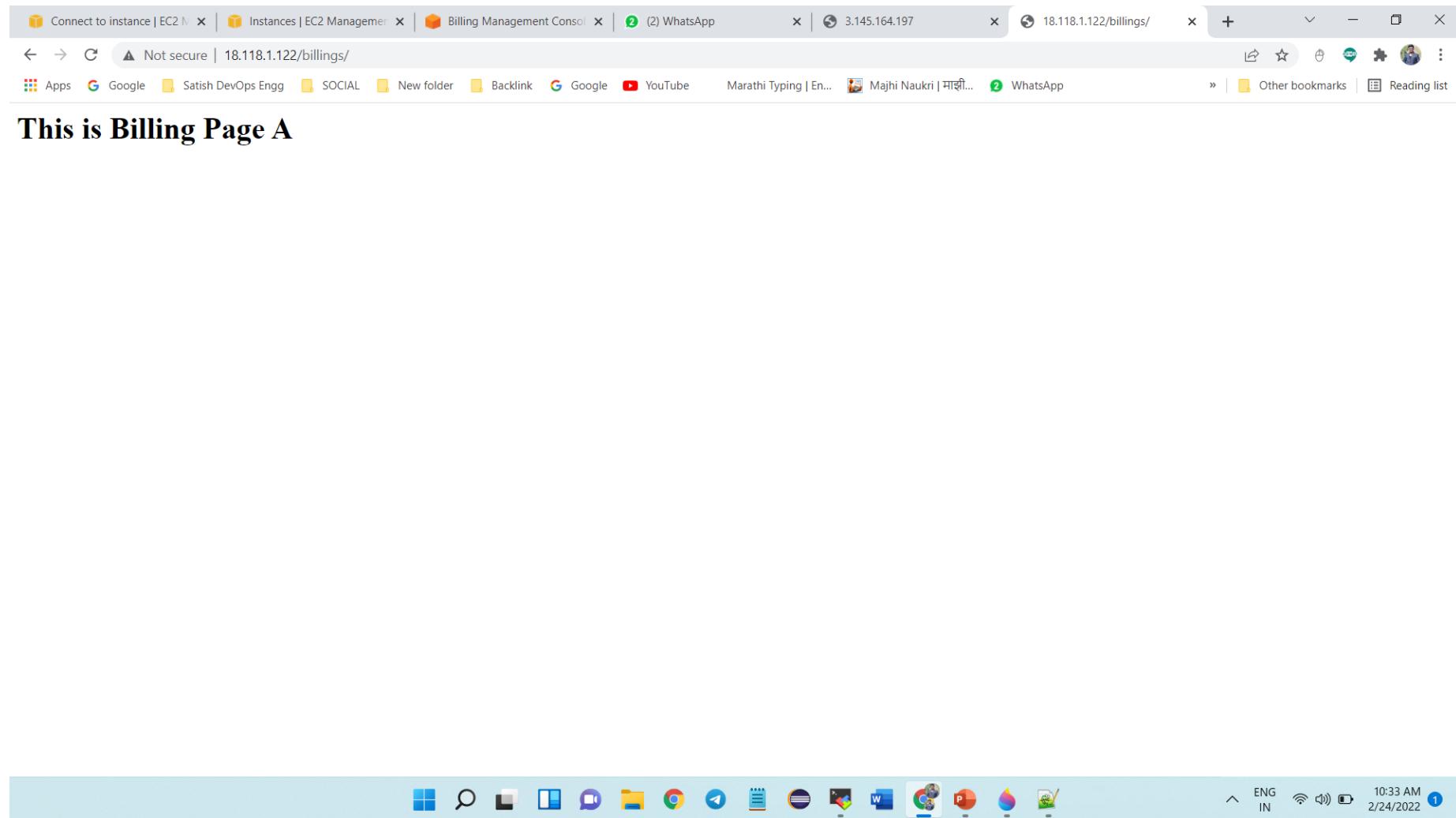
- Starting the httpd service: `service httpd start`
- Changing directory to /var/www/html: `cd /var/www/html`
- Creating a new file index.html and adding the content: `<h1>This is homepage 2</h1>`
- Creating a new directory billings: `mkdir billings`
- Changing directory to billings: `cd billings`
- Creating a new file index.html and adding the content: `<h1>This is Billing Page A</h1>`

The terminal window includes a menu bar with options like Terminal, Sessions, View, Xserver, Tools, Games, Settings, Macros, Help, and a toolbar with icons for Session, Servers, Tools, Games, Sessions, View, Split, MultiExec, Tunneling, Packages, Settings, and Help. The status bar at the bottom shows the message "UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>". It also displays system information such as ENG IN, battery level, and the date and time (10:31 AM, 2/24/2022).

- Create **index.html** file using above commands



- Check by using public ip



- Check by using public ip with header

- For instance – 3 in zone b

```
root@ip-172-31-17-225:/home/mobaxterm/desktop/keys> ssh -i "ohio-instance.pem" ec2-user@ec2-18-118-17-29.us-east-2.compute.amazonaws.com
Warning: Permanently added 'ec2-18-118-17-29.us-east-2.compute.amazonaws.com' (RSA) to the list of known hosts.
X11 forwarding request failed on channel 0

Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
8 package(s) needed for security, out of 14 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-17-225 ~]$
[ec2-user@ip-172-31-17-225 ~]$ sudo su -
[root@ip-172-31-17-225 ~]#
[root@ip-172-31-17-225 ~]# cd /
[root@ip-172-31-17-225 /]#
[root@ip-172-31-17-225 /]#
[root@ip-172-31-17-225 /]# yum install httpd -y
```

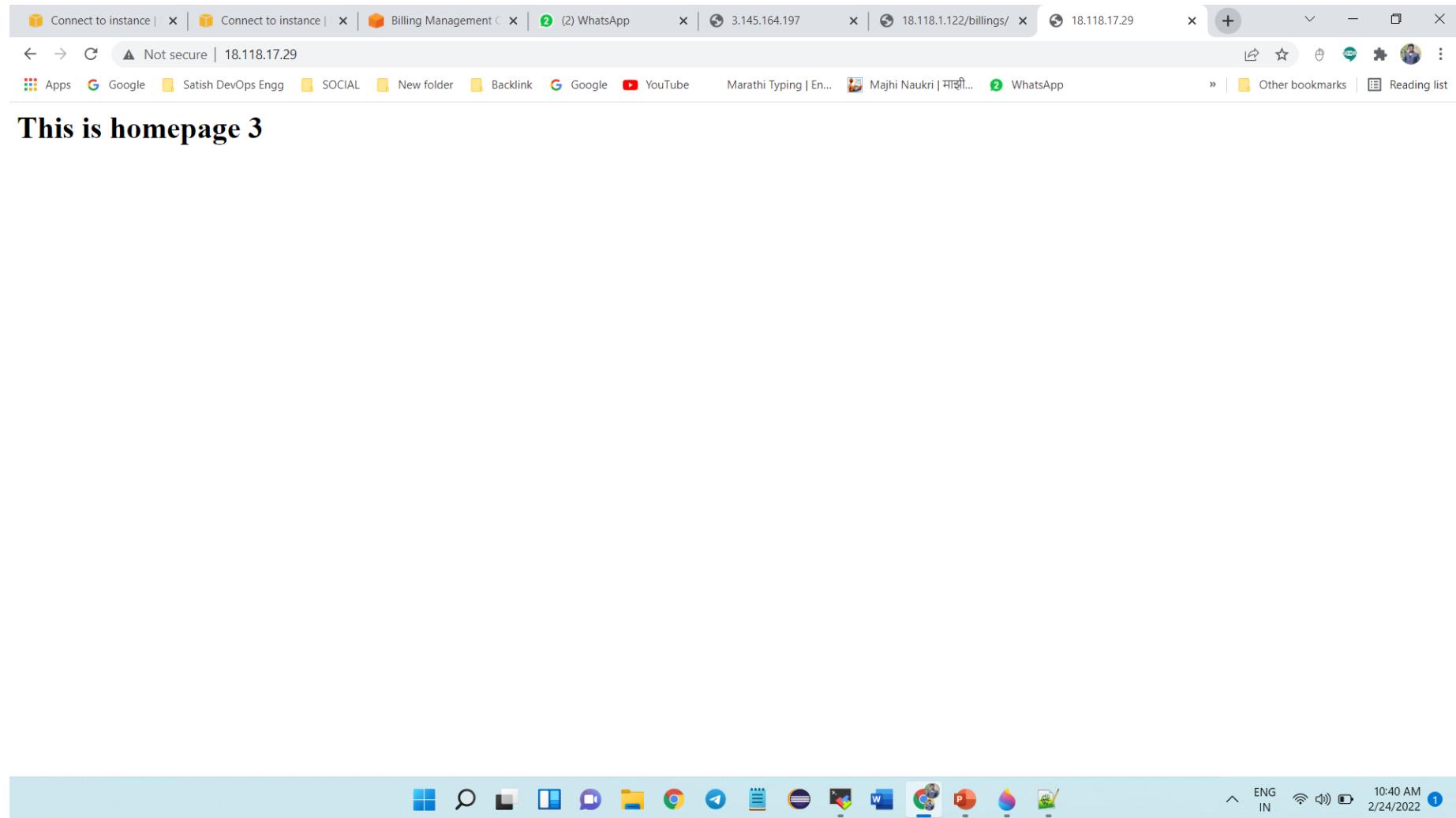
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

The screenshot shows a terminal session in MobaXterm with the following commands and output:

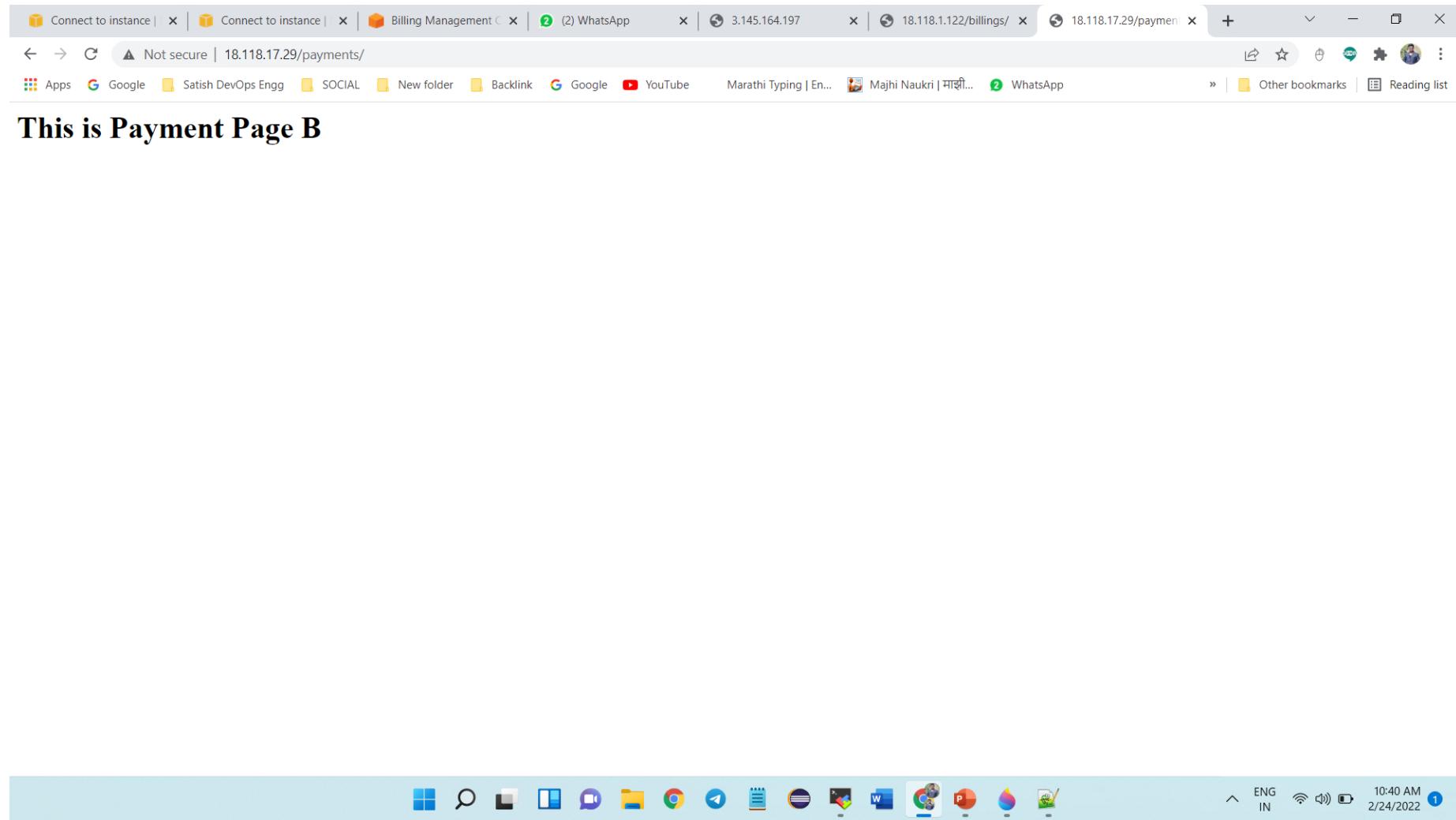
```
[root@ip-172-31-17-225 ~]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-17-225 ~]# cd /var/www/html
[root@ip-172-31-17-225 html]# cat >> index.html
<h1>This is homepage 3</h1>
[root@ip-172-31-17-225 html]# mkdir payments
[root@ip-172-31-17-225 html]# cd payments
[root@ip-172-31-17-225 payments]# cat >> index.html
<h1>This is Payment Page B</h1>
[root@ip-172-31-17-225 payments]#
```

The terminal interface includes a menu bar with Terminal, Sessions, View, Xserver, Tools, Games, Settings, Macros, Help, and a toolbar with Session, Servers, Tools, Games, Sessions, View, Split, MultiExec, Tunneling, Packages, Settings, and Help. The status bar at the bottom shows "UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>" and system information like ENG IN, 10:39 AM, 2/24/2022.

- Create **index.html** file using above commands

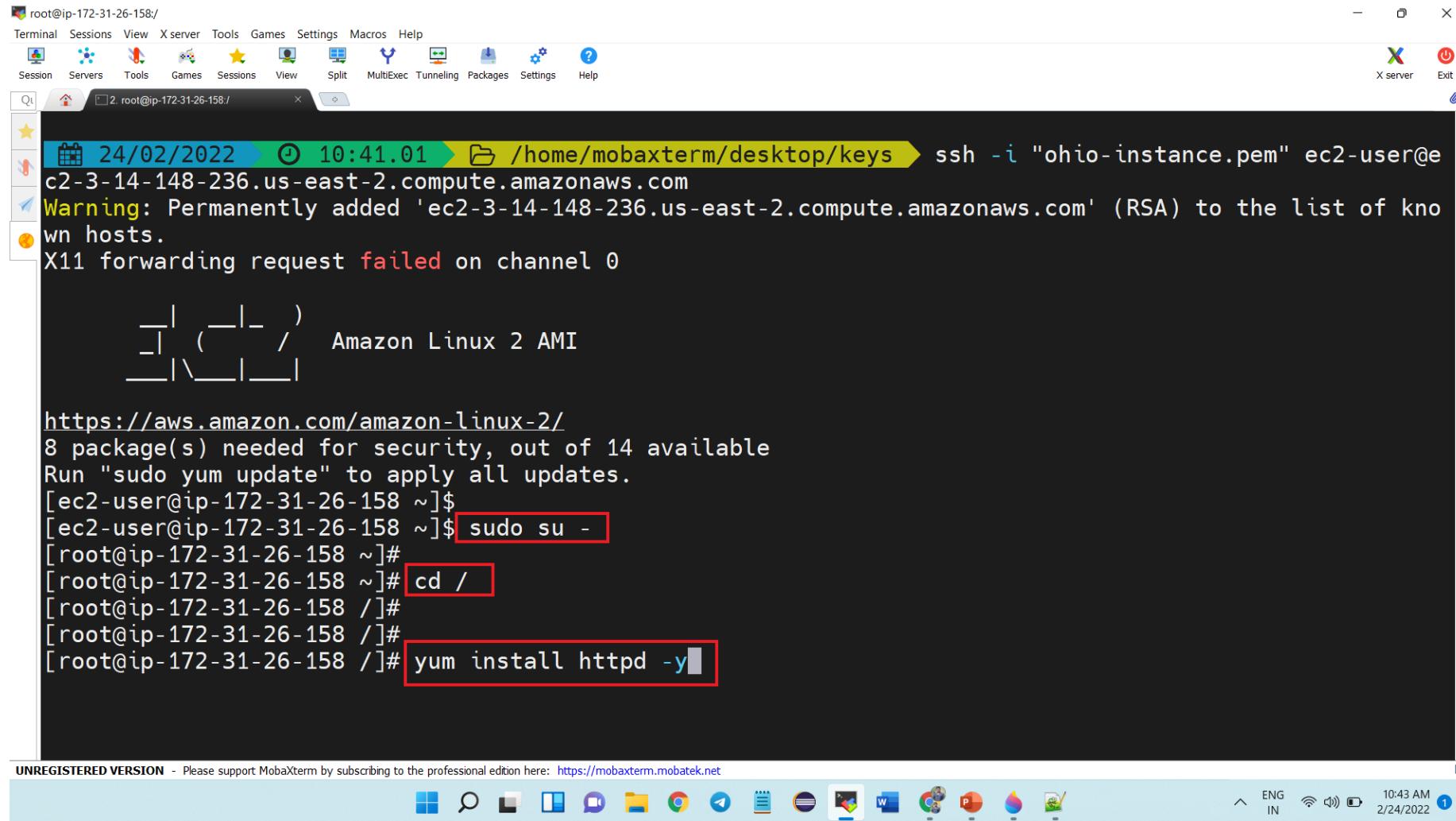


- Check by using public ip



- Check by using public ip with header

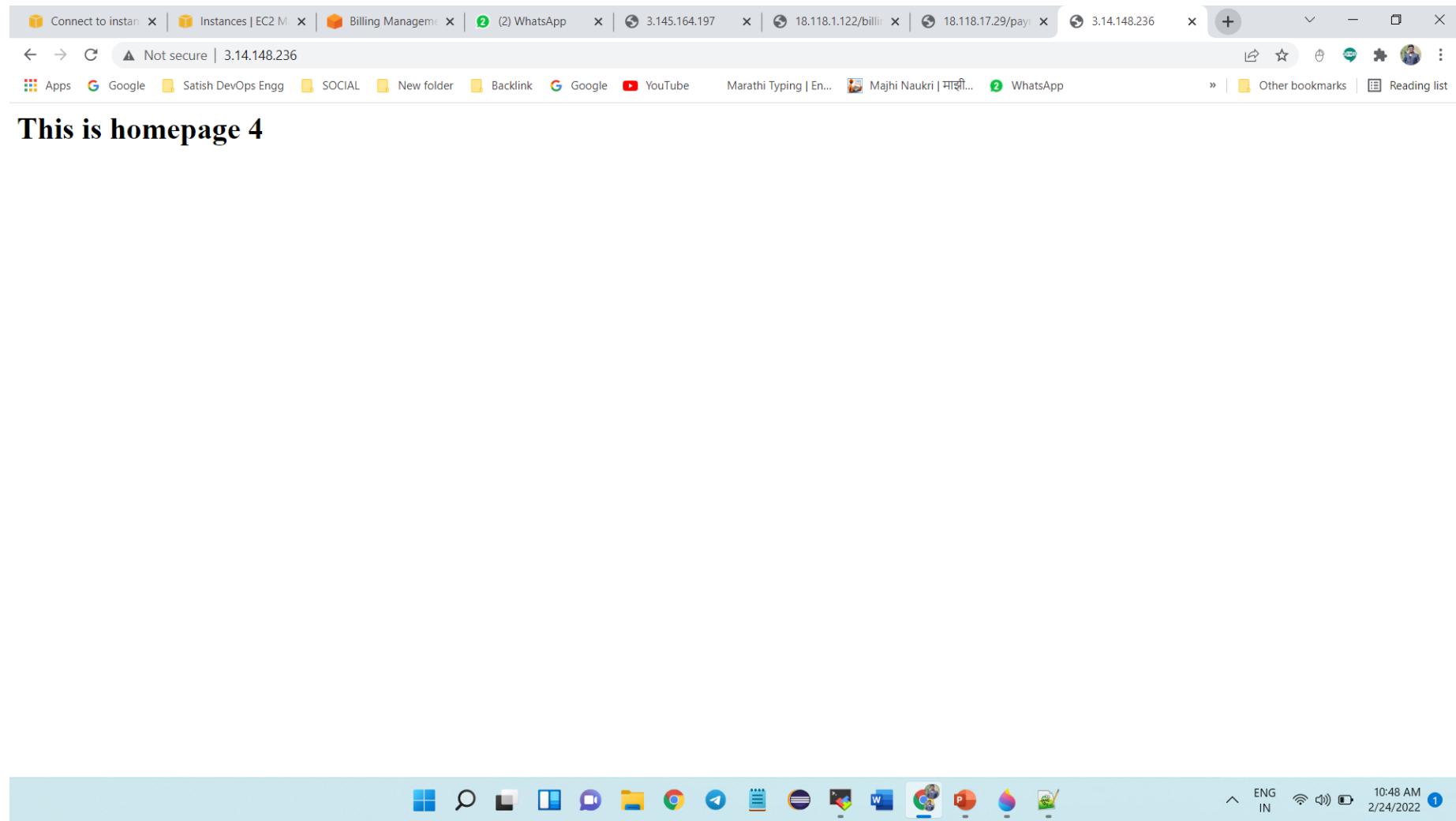
- For instance – 4 in zone b



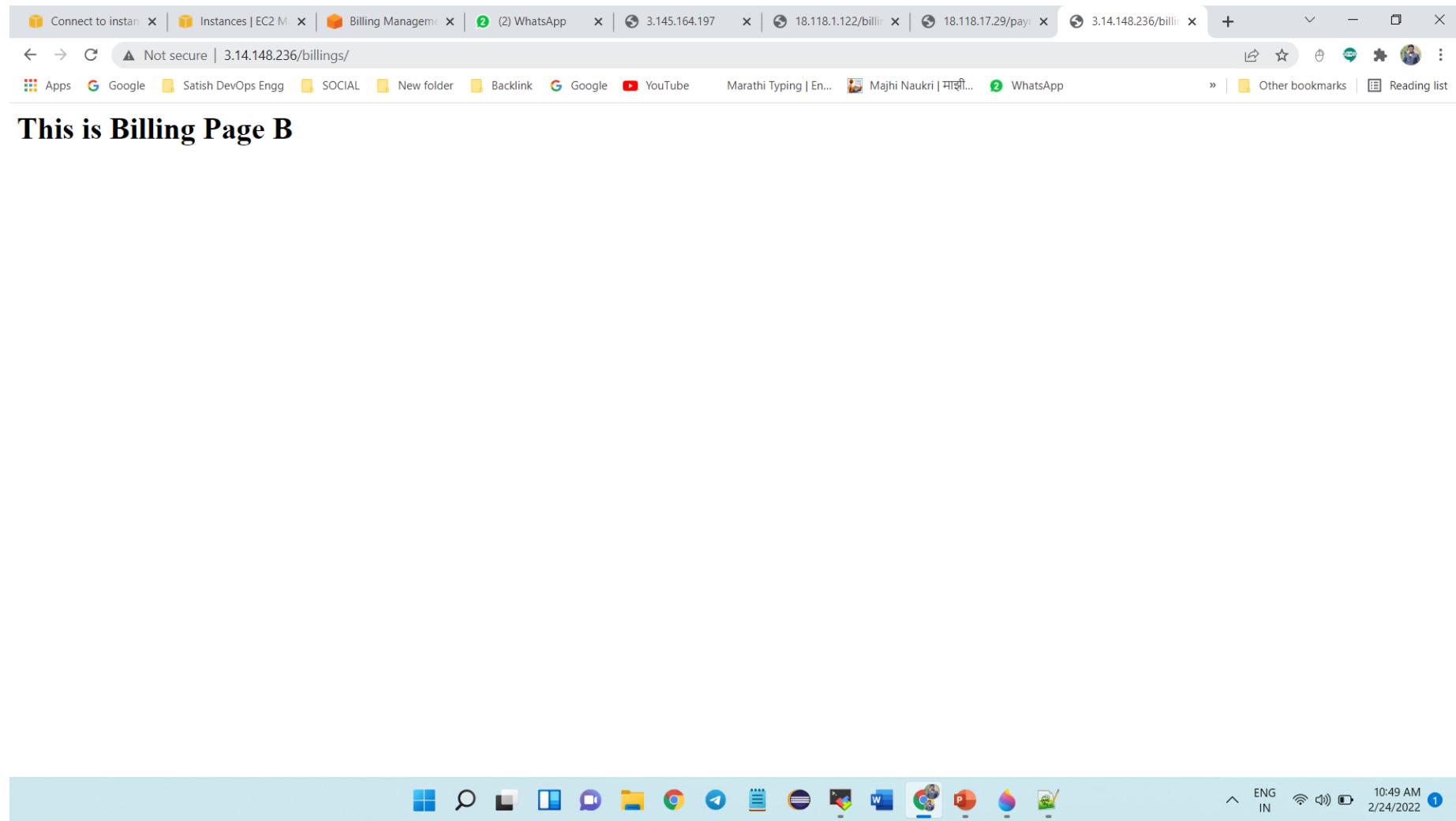
The screenshot shows a MobaXterm window with a terminal session running as root on a Linux system. The session title is "root@ip-172-31-26-158:/var/www/html/billings". The terminal window has a toolbar at the top with icons for Session, Servers, Tools, Games, Sessions, View, Split, MultiExec, Tunneling, Packages, Settings, and Help. The status bar at the bottom indicates an unregistered version, a taskbar with various application icons, and system status including language (ENG IN), battery level (10:45 AM 2/24/2022).

```
[root@ip-172-31-26-158 ~]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-26-158 ~]#
[root@ip-172-31-26-158 ~]# cd /var/www/html
[root@ip-172-31-26-158 html]#
[root@ip-172-31-26-158 html]# cat >> index.html
<h1>This is homepage 4</h1>
[root@ip-172-31-26-158 html]#
[root@ip-172-31-26-158 html]# mkdir billings
[root@ip-172-31-26-158 html]#
[root@ip-172-31-26-158 html]# cd billings
[root@ip-172-31-26-158 billings]#
[root@ip-172-31-26-158 billings]# cat >> index.html
<h1>This is Billing Page B</h1>
[root@ip-172-31-26-158 billings]#
```

- Create **index.html** file using above commands



- Check by using public ip



- Check by using public ip with header

- All the files are working fine in all the instances.
- Now we will create target group accordingly.

The screenshot shows the AWS EC2 Target groups page. The left sidebar lists various services under EC2, with 'Target Groups' highlighted. The main area displays a table titled 'Target groups' with columns for Name, ARN, Port, Protocol, Target type, and Load balancer. A message at the bottom says 'No target groups to display.' An orange box highlights the 'Create target group' button in the top right corner of the table header.

- Go to target groups
- Click on **Create Target Group**

The screenshot shows the AWS EC2 Target groups 'Create target group' wizard. The current step is 'Step 1: Specify group details'. The page title is 'Specify group details'. A sub-section titled 'Basic configuration' contains the instruction: 'Your load balancer routes requests to the targets in a target group and performs health checks on the targets.' Below this, the 'Choose a target type' section is displayed, showing four options: 'Instances' (selected), 'IP addresses', 'Lambda function', and 'Application Load Balancer'. The 'Instances' option is highlighted with a blue border and includes a bulleted list: 'Supports load balancing to instances within a specific VPC.' The other three options are shown in separate boxes.

Feedback English (US) © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 10:53 AM 2/24/2022

- Select instances

The screenshot shows the AWS CloudFront console interface for creating a new target group. The target group name is set to "homepage-target-group". The protocol is selected as "HTTP" and the port is "80". A VPC is chosen, and the protocol version is set to "HTTP1". The "Health checks" section is partially visible below.

Target group name
homepage-target-group

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

Protocol Port
HTTP : 80

VPC
Select the VPC with the instances that you want to include in the target group.
-
vpc-0d530da491583e01c
IPv4: 172.31.0.0/16

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.

Feedback English (US) © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 10:54 AM 2/24/2022

- Add name of Target Group

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 ping the root, or specify a custom path if preferred.
/

Up to 1024 characters allowed.

► **Advanced health check settings**

► **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**

Feedback English (US) © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 10:55 AM 2/24/2022 1

- Scroll Down and click on next

EC2 > Target groups > Create target group

Step 1
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 (4/4)

Instance ID	Name	State	Security groups	Zone	Subnet ID
i-08c357f46e8f92409	payment-A	running	ohio-security-group-http-80	us-east-2a	subnet-04bb880c3944e9026
i-04275796ed33936b1	billing-A	running	ohio-security-group-http-80	us-east-2a	subnet-04bb880c3944e9026
i-012e58119dd28f18b	payment-B	running	ohio-security-group-http-80	us-east-2b	subnet-0d94ff13f84c09eb1
i-0fdfbd5661141b447	billing-B	running	ohio-security-group-http-80	us-east-2b	subnet-0d94ff13f84c09eb1

4 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

Feedback English (US) ▾ © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 10:57 AM 2/24/2022

- Select instances for homepage
- Click on **Include as pending below**

The screenshot shows the AWS Lambda console with the URL us-east-2.console.aws.amazon.com/lambda/home?region=us-east-2#CreateTargetGroup. The browser tab is titled "Billing Manager". The search bar at the top contains the placeholder "Search for services, features, blogs, docs, and more". Below the search bar is a text input field with the placeholder "1-65535 (separate multiple ports with commas)". A button labeled "Include as pending below" is present. A message indicates "4 selections are now pending below. Include more or register targets when ready." The main section is titled "Review targets" and contains a table titled "Targets (4)". The table columns are: Remove, Health status, Instance ID, Name, Port, State, Security groups, Zone, and Subnet ID. The data in the table is as follows:

Remove	Health status	Instance ID	Name	Port	State	Security groups	Zone	Subnet ID
X	Pending	i-08c357f46e8f92409	payment-A	80	running	ohio-security-group-http-80	us-east-2a	subnet-04bb880c3944e9026
X	Pending	i-04275796ed33936b1	billing-A	80	running	ohio-security-group-http-80	us-east-2a	subnet-04bb880c3944e9026
X	Pending	i-012e58119dd28f18b	payment-B	80	running	ohio-security-group-http-80	us-east-2b	subnet-0d94ff13f84c09eb1
X	Pending	i-0fdfbd5661141b447	billing-B	80	running	ohio-security-group-http-80	us-east-2b	subnet-0d94ff13f84c09eb1

At the bottom of the page, there are buttons for "Cancel", "Previous", and "Create target group". The "Create target group" button is highlighted with a red box. The footer includes links for "Feedback", "English (US)", "© 2022, Amazon Internet Services Private Ltd. or its affiliates.", "Privacy", "Terms", and "Cookie preferences". It also shows system status icons for battery, signal, and network.

- Click on Create Target Group

The screenshot shows the AWS EC2 Target groups page. The URL in the browser is `us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#TargetGroups:`. The page title is "Target groups". A success message at the top says "Successfully created target group: homepage-target-group". The main table displays one target group:

Name	ARN	Port	Protocol	Target type	Load balancer
homepage-target-group	arn:aws:elasticloadbalancing:... (blue link)	80	HTTP	Instance	(Info) None associated

A red box highlights the "homepage-target-group" name in the table. Below the table, a message says "Select a target group above." The left sidebar shows navigation options for EC2 instances, images, and elastic block store.

- Our 1st target group for homepage is created successfully

- Now we will create target group for payments page

The screenshot shows the AWS EC2 Target groups page. On the left, there's a sidebar with various EC2-related options like EC2 Dashboard, Global View, Instances, Images, and Elastic Block Store. The main area displays a table titled 'Target groups (1) Info' with one entry: 'homepage-target-group'. The table columns include Name, ARN, Port, Protocol, Target type, and Load balancer. A search bar at the top of the table allows filtering by target group name. To the right of the table, there's a large orange button labeled 'Create target group' with a bounding box of approximately [250, 710, 280, 790]. The status bar at the bottom indicates the date as 2/24/2022 and the time as 11:01 AM.

Name	ARN	Port	Protocol	Target type	Load balancer
homepage-target-group	arn:aws:elasticloadbalancing:... arn:aws:elasticloadbalancing:...	80	HTTP	Instance	None associated

Select a target group above.

Feedback English (US) © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 11:01 AM 2/24/2022

- Click on Create Target Group

The screenshot shows the AWS EC2 Target groups 'Create target group' wizard. The current step is 'Step 1: Specify group details'. The page title is 'Specify group details'. A sub-instruction says 'Your load balancer routes requests to the targets in a target group and performs health checks on the targets.' Below this, the 'Basic configuration' section states 'Settings in this section cannot be changed after the target group is created.' The 'Choose a target type' section contains four options: 'Instances' (selected), 'IP addresses', 'Lambda function', and 'Application Load Balancer'. The 'Instances' option is highlighted with a blue border and includes a bulleted list: 'Supports load balancing to instances within a specific VPC.' The other three options have a grey border and a smaller font size.

Feedback English (US) © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 11:01 AM 2/24/2022

- Select instances

The screenshot shows the AWS CloudFront console interface for creating a new target group. The target group name is set to "payments-target-group". The protocol is selected as "HTTP" and the port is "80". A VPC is chosen, and the protocol version is set to "HTTP1". The "Health checks" section is partially visible at the bottom.

Target group name
payments-target-group

Protocol Port
HTTP : 80

VPC
Select the VPC with the instances that you want to include in the target group.
-
vpc-0d530da491583e01c
IPv4: 172.31.0.0/16

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.

Feedback English (US) © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 11:02 AM 2/24/2022

- Add name of Target Group

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 ping the root, or specify a custom path if preferred.
/

Up to 1024 characters allowed.

► Advanced health check settings

► 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**

Feedback English (US) © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 11:02 AM 2/24/2022 1

- Scroll Down and click on next

EC2 > Target groups > Create target group

Step 1
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 (2/4)

	Instance ID	Name	State	Security groups	Zone	Subnet ID
<input checked="" type="checkbox"/>	i-08c357f46e8f92409	payment-A	running	ohio-security-group-http-80	us-east-2a	subnet-04bb880c3944e9026
<input type="checkbox"/>	i-04275796ed33936b1	billing-A	running	ohio-security-group-http-80	us-east-2a	subnet-04bb880c3944e9026
<input checked="" type="checkbox"/>	i-012e58119dd28f18b	payment-B	running	ohio-security-group-http-80	us-east-2b	subnet-0d94ff13f84c09eb1
<input type="checkbox"/>	i-0fdfbd5661141b447	billing-B	running	ohio-security-group-http-80	us-east-2b	subnet-0d94ff13f84c09eb1

2 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

Feedback English (US) ▾ © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 11:02 AM 2/24/2022 1

- Select instances for payments
- Click on **Include as pending below**

The screenshot shows the AWS Lambda console interface. At the top, there are several tabs and browser bookmarks. The main area displays the configuration for a new Lambda function named "Billing Manager". The "Code" tab is selected, showing the code editor with the following code:

```
function handler(event, context) {  const response = {    statusCode: 200,    body: JSON.stringify({      message: "Hello from Lambda"    })  };  context.succeed(response);}
```

The "Test" tab is also visible, showing a sample event and a successful execution result.

- Click on Create Target Group

The screenshot shows the AWS EC2 Target Groups page. The navigation bar at the top includes links for Connect to instance, Target groups, Billing Management, WhatsApp, and several other browser tabs. The main search bar says "Search for services, features, blogs, docs, and more". A sidebar on the left lists EC2 services like EC2 Dashboard, Global View, Events, Tags, Limits, Instances (with sub-links for Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, and Capacity Reservations), Images (AMIs, AMI Catalog), and Elastic Block Store.

A prominent green success message box at the top center states "Successfully created target group: payments-target-group".

The main content area displays a table titled "Target groups (2) Info". The table has columns for Name, ARN, Port, Protocol, Target type, and Load balancer. It lists two entries:

Name	ARN	Port	Protocol	Target type	Load balancer
homepage-target-group	arn:aws:elasticloadbalancing:... arn:aws:elasticloadbalancin...	80	HTTP	Instance	None associated
payments-target-group	arn:aws:elasticloadbalancin...	80	HTTP	Instance	None associated

Below the table, a message says "Select a target group above." The bottom of the page includes standard footer links for Feedback, English (US), Privacy, Terms, and Cookie preferences, along with system status icons for ENG IN, 11:03 AM, and 2/24/2022.

- Our 2nd target group for payments is created successfully

- Now we will create target group for billings page

The screenshot shows the AWS EC2 Target groups page. On the left, there's a sidebar with various EC2-related options like Instances, Images, and Elastic Block Store. The main area displays a table titled "Target groups (2) Info" with two entries:

Name	ARN	Port	Protocol	Target type	Load balancer
homepage-target-group	arn:aws:elasticloadbalancing:... arn:aws:elasticloadbalancin...	80	HTTP	Instance	None associated
payments-target-group	arn:aws:elasticloadbalancing:... arn:aws:elasticloadbalancin...	80	HTTP	Instance	None associated

Below the table, a message says "Select a target group above." At the top right of the main area, there's a red box highlighting the "Create target group" button. The browser's address bar shows the URL "us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#TargetGroups:".

- Click on Create Target Group

The screenshot shows the AWS EC2 Target groups 'Create target group' wizard. The current step is 'Step 1: Specify group details'. The page title is 'Specify group details'. A sub-instruction says 'Your load balancer routes requests to the targets in a target group and performs health checks on the targets.' Below this, the 'Basic configuration' section is titled 'Basic configuration' with the note 'Settings in this section cannot be changed after the target group is created.' The 'Choose a target type' section contains four options: 'Instances' (selected), 'IP addresses', 'Lambda function', and 'Application Load Balancer'. The 'Instances' option is highlighted with a blue border and includes the description: 'Supports load balancing to instances within a specific VPC.' The other three options are described as follows:

- IP addresses**: Supports load balancing to VPC and on-premises resources, facilitates routing to multiple IP addresses and network interfaces on the same instance, and offers flexibility with microservice based architectures, simplifying inter-application communication.
- Lambda function**: Facilitates routing to a single Lambda function and is accessible to Application Load Balancers only.
- Application Load Balancer**: This option is not currently available.

At the bottom of the page, there are links for 'Feedback', 'English (US)', '© 2022, Amazon Internet Services Private Ltd. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'. The status bar at the bottom right shows 'ENG IN', '11:05 AM', '2/24/2022', and a notification icon.

- Select instances

The screenshot shows the AWS Lambda console with the URL us-east-2.console.aws.amazon.com/lambda/v2/home?region=us-east-2#CreateTargetGroup. The page is titled "Create Target Group".

Target group name: billings-target-group (highlighted with a red box)

Protocol: HTTP (selected) : Port 80

VPC: Select the VPC with the instances that you want to include in the target group.

- vpc-0d530da491583e01c
IPv4: 172.31.0.0/16

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: (dropdown menu)

Footer:

- Feedback English (US) ▾
- © 2022, Amazon Internet Services Private Ltd. or its affiliates.
- Privacy Terms Cookie preferences
- ENG IN 11:05 AM 2/24/2022

- Add name of Target Group

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 ping the root, or specify a custom path if preferred.
/

Up to 1024 characters allowed.

► Advanced health check settings

► 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

Feedback English (US) © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 11:05 AM 2/24/2022 1

- Scroll Down and click on next

The screenshot shows the AWS EC2 Target groups 'Create target group' interface. The 'Register targets' step is active. A table lists four instances:

Instance ID	Name	State	Security groups	Zone	Subnet ID
i-08c357f46e8f92409	payment-A	running	ohio-security-group-http-80	us-east-2a	subnet-04bb880c3944e9026
i-04275796ed33936b1	billing-A	running	ohio-security-group-http-80	us-east-2a	subnet-04bb880c3944e9026
i-012e58119dd28f18b	payment-B	running	ohio-security-group-http-80	us-east-2b	subnet-0d94ff13f84c09eb1
i-0fdfbd5661141b447	billing-B	running	ohio-security-group-http-80	us-east-2b	subnet-0d94ff13f84c09eb1

Two instances, 'billing-A' and 'billing-B', are selected and highlighted with red boxes. Below the table, a message says '2 selected'. A port range '80' is specified for routing traffic. The 'Include as pending below' button is highlighted with a red box.

- Select instances for billings
- Click on **Include as pending below**

The screenshot shows the AWS Lambda console interface. At the top, there are several tabs and links: "Connect to instance", "Target groups | EC2", "Billing Manager", "(3) WhatsApp", "3.145.164.197", "18.118.1.122/billing", "18.118.17.29/pay", "3.14.148.236/billing", "+", "us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#CreateTargetGroup:", "Apps", "Google", "Satisf DevOps Engg", "SOCIAL", "New folder", "Backlink", "Google", "YouTube", "Marathi Typing | En...", "Majhi Naukri | माझी...", "WhatsApp", "Other bookmarks", "Reading list", "Ohio", "Satisf DevOps Engineer". The main navigation bar has "aws" and "Services" selected. A search bar at the top right contains the placeholder "Search for services, features, blogs, docs, and more [Alt+S]".

The main content area is titled "Ports for the selected instances" and "Ports for routing traffic to the selected instances." It shows a single port entry: "80". Below it, a note says "1-65535 (separate multiple ports with commas)". There is a button "Include as pending below". A message at the bottom states "2 selections are now pending below. Include more or register targets when ready."

Below this, a section titled "Review targets" displays a table of "Targets (2)". The table columns are: Remove, Health status, Instance ID, Name, Port, State, Security groups, Zone, and Subnet ID. Two entries are listed:

Remove	Health status	Instance ID	Name	Port	State	Security groups	Zone	Subnet ID
X	Pending	i-0dfbd5661141b447	billing-B	80	running	ohio-security-group-http-80	us-east-2b	subnet-0d94ff13f84c09eb1
X	Pending	i-04275796ed33936b1	billing-A	80	running	ohio-security-group-http-80	us-east-2a	subnet-04bb880c3944e9026

At the bottom of the "Review targets" section, there are buttons for "Cancel", "Previous", and "Create target group".

The footer of the browser window includes: "Feedback", "English (US)", "© 2022, Amazon Internet Services Private Ltd. or its affiliates.", "Privacy", "Terms", "Cookie preferences", "ENG IN", "11:06 AM", "2/24/2022", and a notification icon.

- Click on Create Target Group

The screenshot shows the AWS EC2 Target groups page. A success message at the top states "Successfully created target group: billings-target-group". The main table displays three target groups:

Name	ARN	Port	Protocol	Target type	Load balancer
billings-target-group	arn:aws:elasticloadbalancing:... arn:aws:elasticloadbalancin...	80	HTTP	Instance	None associated
homepage-target-group	arn:aws:elasticloadbalancin...	80	HTTP	Instance	None associated
payments-target-group	arn:aws:elasticloadbalancin...	80	HTTP	Instance	None associated

A red box highlights the "billings-target-group" row. A tooltip "Select a target group above." is visible below the table.

- Our 3rd target group for billings is created successfully

- Now we will create load balancer and add rule accordingly

The screenshot shows the AWS EC2 Management console with the Load Balancers section selected. The 'Create Load Balancer' button is highlighted with a red box. The 'Load Balancers' link under the Load Balancing section is also highlighted with a red box.

Feedback English (US) ▾ © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences

ENG IN 11:08 AM 2/24/2022 1

- Go to Load Balancers
- Click on Click Load Balancer

Connect to instance | Load balancers | Billing Manager | (1) WhatsApp | 3.145.164.197 | 18.118.1.122/billing | 18.118.17.29/payment | 3.14.148.236/billing | + | - | X

← → ⌂ 🔒 us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#SelectCreateELBWizard:

Apps Google SOCIAL New folder Backlink Google YouTube Marathi Typing | En... Majhi Naukri | माझी... WhatsApp Other bookmarks Reading list

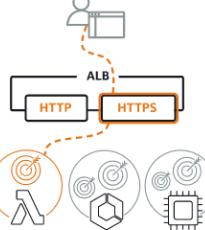
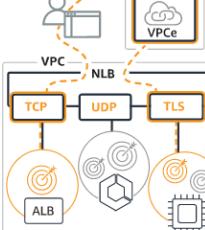
aws Services Search for services, features, blogs, docs, and more [Alt+S]

EC2 > Load balancers > Select load balancer type

Select load balancer type

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

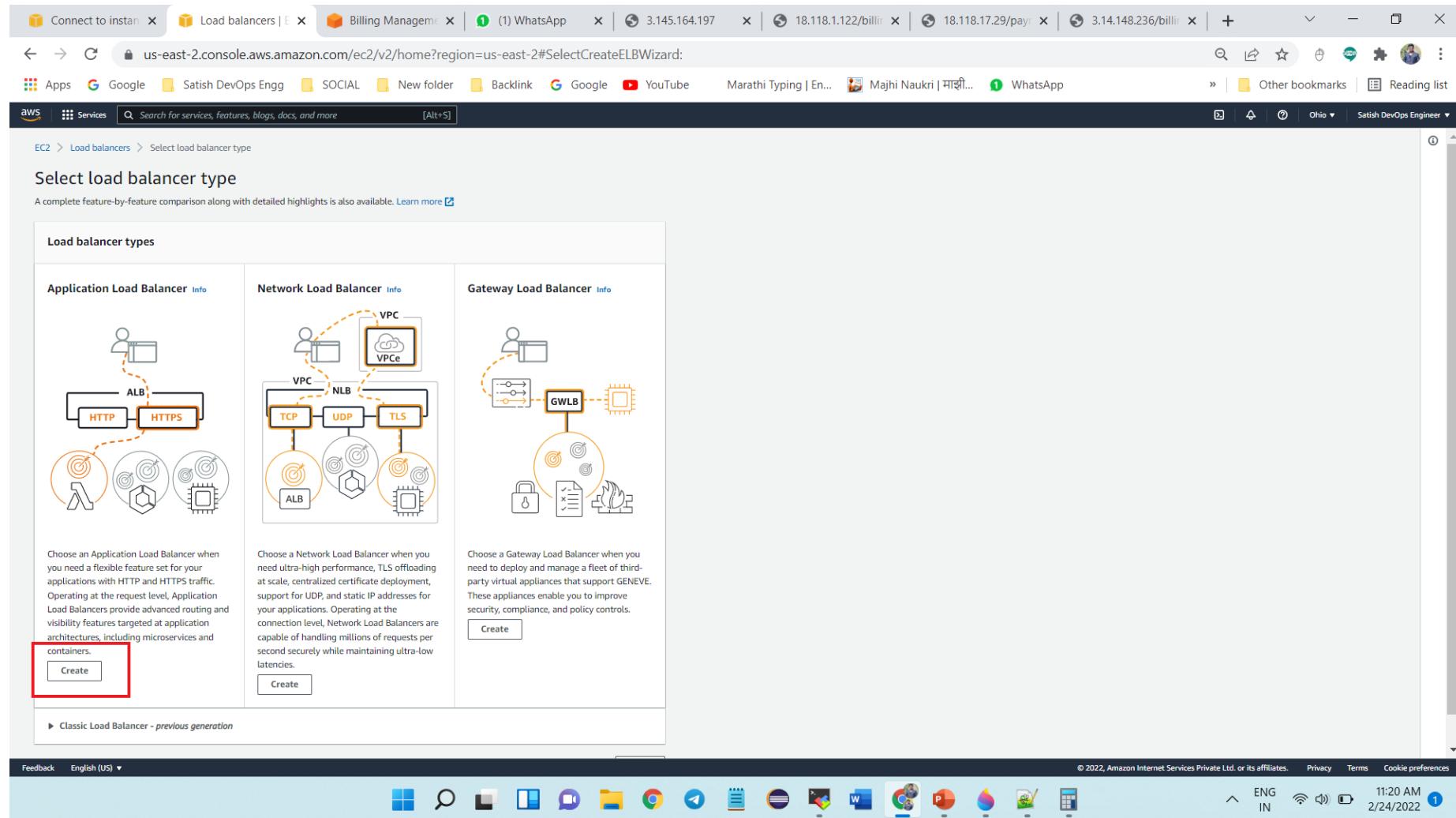
Load balancer types

Application Load Balancer Info	Network Load Balancer Info	Gateway 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. Create	 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	 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

Feedback English (US) ▾ © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences

ENG IN 11:20 AM 2/24/2022



- Select Application Load Balancer
- Click on Create

The screenshot shows the 'Create Application Load Balancer' wizard on the AWS Management Console. The 'Basic configuration' section is highlighted. A red box surrounds the 'Load balancer name' field, which contains the value 'load-balancer'. Another red box surrounds the 'Scheme' section, where 'Internet-facing' is selected. A third red box surrounds the 'IP address type' section, where 'IPv4' is selected.

Load balancer name
Name must be unique within your AWS account and cannot be changed after the load balancer is created.
load-balancer

Scheme [Info](#)
Scheme cannot 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
Recommended for internal load balancers.

- Give name to Load Balancer

The screenshot shows the AWS Lambda console with the URL us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#CreateALBWizard. The page is titled "Create Application Load Balancer" and is on step 2, "Configure the load balancer". The "Mappings" section is displayed, showing three Availability Zones: us-east-2a, us-east-2b, and us-east-2c. Each zone has a checkbox next to its name, which is checked for all three. The "Subnet" dropdown for each zone shows the selected subnet ID. The "IPv4 settings" and "Assigned by AWS" sections are also visible. The entire "Mappings" section is highlighted with a red box.

- Select all Zones

The screenshot shows the AWS CloudFormation Create ALB Wizard interface. In the top navigation bar, there are several tabs and links, including "Connect to instance", "Load balancers", "Billing Manager", "(1) WhatsApp", "3.145.164.197", "18.118.1.122/billing", "18.118.17.29/payment", "3.14.148.236/billing", and "Other bookmarks". The main content area is titled "Services" and has a search bar "Search for services, features, blogs, docs, and more".

The configuration pane shows the following settings:

- Subnet:** subnet-0a84a11858506eb19
- IPv4 settings:** Assigned by AWS

Security groups: Info

A security group is a set of firewall rules that control the traffic to your load balancer.

Security groups:

- default (VPC: vpc-0d530da491583e01c)
- ohio-instance (VPC: vpc-0d530da491583e01c)
- ohio-security-group-https-80 (VPC: vpc-0d530da491583e01c)

The "ohio-security-group-https-80" entry is highlighted with a red box.

Listener HTTP:80:

Protocol	Port	Default action
HTTP	: 80	Select a target group

Feedback: English (US)

© 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences

ENG IN 11:22 AM 2/24/2022

- Select security groups

The screenshot shows the AWS Load Balancer Wizard interface. At the top, there are several browser tabs open, including 'Connect to instance', 'Load balancers | E...', 'Billing Management', '(1) WhatsApp', '3.145.164.197', '18.118.1.122/bill...', '18.118.17.29/pay...', '3.14.148.236/bill...', and others. The main content area is titled 'Create new security group' and shows two entries: 'default sg-005a9761f1028d93f' and 'ohio-security-group-https-80 sg-03ca72a52fa7e0e71'. The second entry is highlighted with a red box. Below this, the 'Listeners and routing' section is shown, with a 'Listener HTTP:80' configuration. The 'Protocol' is set to 'HTTP' and 'Port' is '80'. The 'Default action' dropdown is set to 'Select a target group' and has a 'Create target' button. A search bar shows 'billings-target-group'. Below it, 'homepage-target-group' is selected, highlighted with a red box. Other options like 'payments-target-group' are also listed. At the bottom, there's an 'Add-on services - optional' section with a note about integrating additional AWS services.

- Select Target Group

us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#CreateALBWizard:

Services Search for services, features, blogs, docs, and more [Alt+S]

Listener HTTP:80

Protocol Port

HTTP : 80

Default action [Info](#)

Forward to **homepage-target-group** Target type: Instance, IPv4

Create target group [\[i\]](#)

Add listener

Add-on services - optional

Additional AWS services can be integrated with this load balancer at launch. You can also add these and other services after your load balancer is created by reviewing the "Integrated Services" tab for the selected load balancer.

AWS Global Accelerator [Info](#)

Create an accelerator to get static IP addresses and improve the performance and availability of your applications. [Additional charges apply](#) [\[i\]](#)

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.

Feedback English (US) © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences

ENG IN 11:23 AM 2/24/2022 1

- Scroll Down

The screenshot shows the AWS CloudFormation console with the URL us-east-2.console.aws.amazon.com/cloudformation/v2/home?region=us-east-2#CreateStackWizard. The page is titled "Create stack wizard - Step 1: Set stack name and region". The "Region" dropdown is set to "us-east-1". The "Next Step" button at the bottom is highlighted in orange.

- Click on **Create Load Balancer**

Connect to instance | Load balancers | Billing Management | (1) WhatsApp | 3.145.164.197 | 18.118.1.122/billing | 18.118.17.29/payment | 3.14.148.236/billing | + | [us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#CreateLBWizardSuccess:loadBalancerArn=arn:aws:elasticloadbalancing:us-east-2:876283541003:loadbalancer/app/load-balancer](#)

Apps Google Satish DevOps Engg SOCIAL New folder Backlink Google YouTube Marathi Typing | En... Majhi Naukri | माझी... WhatsApp » | Other bookmarks Reading list

 Services [Alt+S]    Ohio Satish DevOps Engineer

Successfully created load balancer: **load-balancer**
Note: 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.

EC2 > Load balancers > Create Application Load Balancer

Create Application Load Balancer

 **Suggested next steps**

- Review, customize, or enable attributes for your load balancer and listeners using the **Description** and **Listeners** tabs within [load-balancer](#).
- Discover other services that you can integrate with your load balancer. Visit the **Integrated services** tab within [load-balancer](#).

[View load balancer](#)

- Click on view load balancer

The screenshot shows the AWS EC2 Management console with the 'Load Balancers' service selected. The main pane displays a table of existing load balancers, with one entry ('load-balancer') highlighted and its details view open. In the detailed view, the 'Listeners' tab is active, and a specific listener entry for 'HTTP : 80' is shown, with the 'View/edit rules' button highlighted.

Load balancer: load-balancer

Name	DNS name	State	VPC ID	Availability Zones	Type	Created At
load-balancer	load-balancer-1023095652.u...	Provisioning	vpc-0d530da491583e01c	us-east-2a, us-east-2c, ...	application	February

Listeners

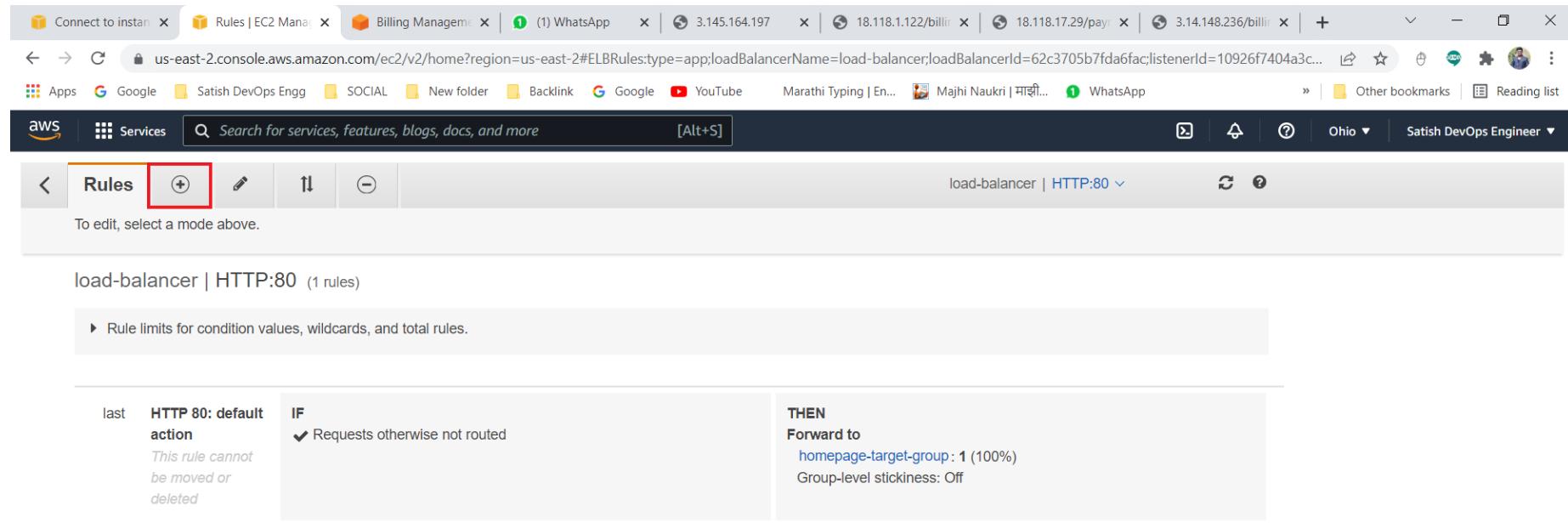
Listeners listen for connection requests using their protocol and port. You can add, remove, or update listeners and listener rules.

To view and edit listener attributes, select the listener and choose Edit.

Add listener Edit Delete

Listener ID	Security policy	SSL Certificate	Rules
HTTP : 80	N/A	N/A	Default: forwarding to homepage-target-group arn...10926f7404a3c71d... View/edit rules

- Now select Load Balancer
- Click on Listeners and then click on **View/edit Rules**



The screenshot shows the AWS Lambda console with the URL us-east-2.console.aws.amazon.com/lambda/v2/home?region=us-east-2#functions:search=load-balancer. The 'Rules' tab is selected, and the '+' icon is highlighted with a red box. The interface displays a single rule configuration:

last	HTTP 80: default action	IF	THEN
	<i>This rule cannot be moved or deleted</i>	✓ Requests otherwise not routed	Forward to homepage-target-group: 1 (100%) Group-level stickiness: Off

At the bottom, there is a navigation bar with links for Feedback, English (US), Privacy, Terms, and Cookie preferences, along with system status indicators for ENG IN, battery level, and the date/time (11:25 AM 2/24/2022).

- Click on + icon

The screenshot shows the AWS Lambda function configuration page. The top navigation bar includes tabs for 'Overview', 'Code', 'Triggers', 'Logs', 'Metrics', 'Actions', and 'Environment'. The 'Code' tab is currently selected. Below the tabs, there's a section for 'Lambda function code' with a dropdown menu for 'Runtime' (set to 'Node.js 14.x') and a large text area for the function code. A red box highlights the 'Edit' button in the top right corner of the code editor. The bottom of the page features a toolbar with various AWS services and a status bar showing the date and time.

• Click on Insert Rule

- Click on Insert Rule

The screenshot shows the AWS Lambda function configuration page. The top navigation bar includes tabs for 'Overview', 'Actions', 'Code', 'Environment', 'Logs', and 'Metrics'. The 'Code' tab is selected, showing the code editor with the following Python code:

```
def lambda_handler(event, context):  
    # Add your code here to process the event  
    # and return a response.  
    # You can pass context.invoked_by to  
    #ynamoDB = boto3.  
    #return {"statusCode": 200, "body": "Hello from Lambda!"}
```

The 'Handler' dropdown is set to `lambda_function.lambda_handler`. The 'Runtime' is set to `Python 3.7`. The 'Memory limit' is set to `128 MB`, and the 'Timeout' is set to `3.5 seconds`. The 'Environment variables' section contains the variable `stage` with the value `prod`. The 'Logs' section shows a log entry with the timestamp `2018-07-10T11:45:00+00:00` and the message `Function execution succeeded`.



- Select Path in Add condition
 - Add /payments/* into path

The screenshot shows the AWS Lambda@Edge Rule Editor interface. At the top, there are several tabs: 'Connect to instance', 'Rules | EC2 Manager', 'Billing Manager', '(1) WhatsApp', '3.145.164.197', '18.118.1.122/billing', '18.118.17.29/payment', '3.14.148.236/billing', and a '+' button. Below the tabs, the browser address bar shows 'us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#ELBRules:type=app;loadBalancerName=load-balancer;loadBalancerId=62c3705b7fda6fac;listenerId=10926f7404a3c...'. The main navigation bar has 'Services' selected. A search bar says 'Search for services, features, blogs, docs, and more [Alt+S]'. On the left, a sidebar shows 'load-balancer | HTTP:80 (2 rules)' and a note about rule limits. The main content area displays two rules:

RULE ID	IF (all match)	THEN
1	Path... is /payments/* or Value + Add condition	+ Add action Forward to... Redirect to... Return fixed response... Note: Additional actions are available for HTTPS listeners.
last	HTTP 80: default action This rule cannot be moved or	IF Requests otherwise not routed THEN Forward to homepage-target-group: 1 (100%) Group-level stickiness: Off

The 'Forward to...' option in the 'Add action' dropdown is highlighted with a red box. The bottom of the screen shows the Windows taskbar with various pinned icons and the system tray.

- Select **Forward to** in **Add Action**

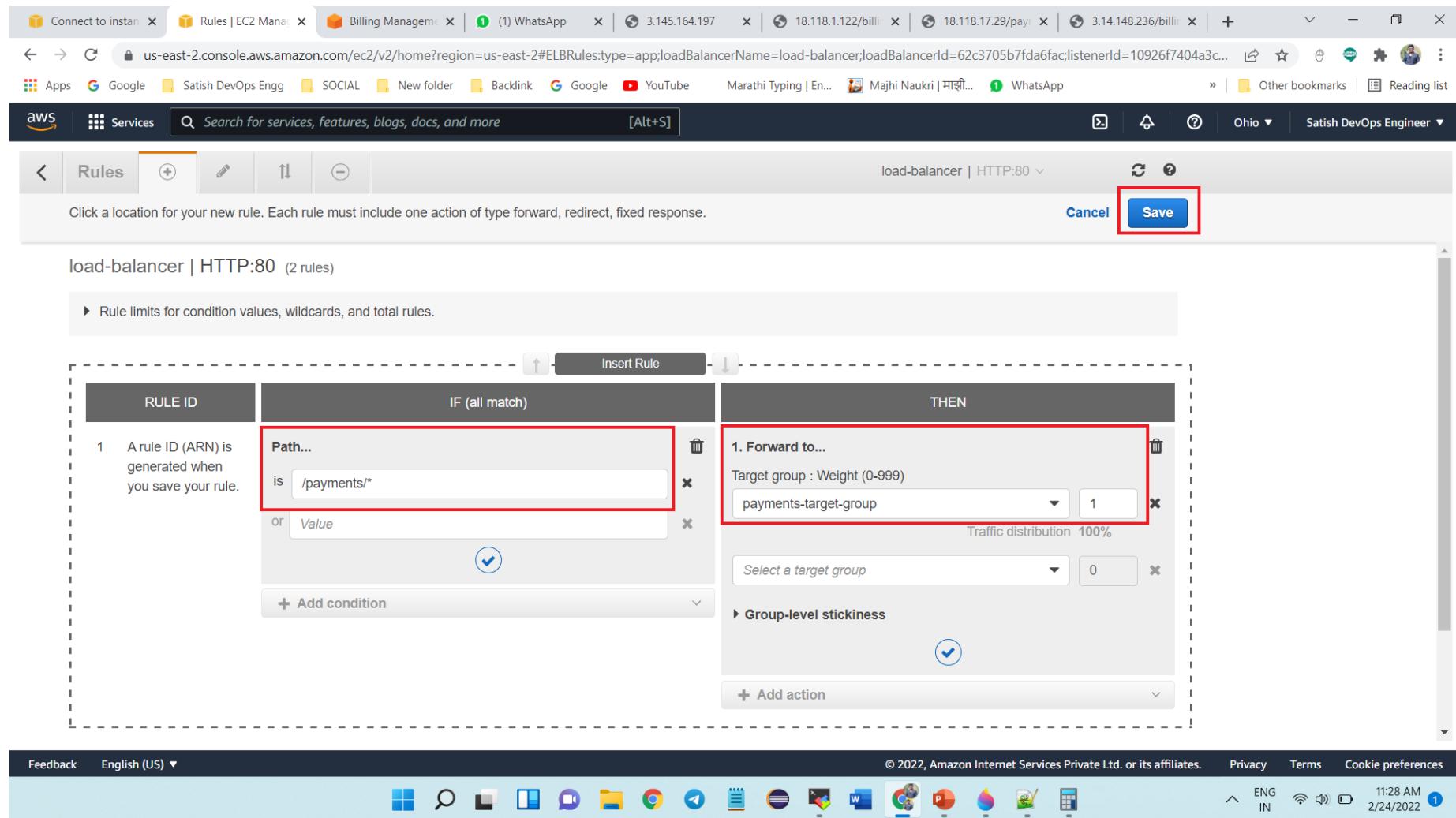
The screenshot shows the AWS CloudFront Rules configuration interface. The top navigation bar includes tabs for 'Connect to instance', 'Rules | EC2 Manager', 'Billing Manager', '(1) WhatsApp', '3.145.164.197', '18.118.1.122/billing', '18.118.17.29/payment', '3.14.148.236/billing', and a '+' button. Below the navigation is a search bar and a services menu.

The main area displays a rule named 'load-balancer | HTTP:80'. The rule table has three columns: 'RULE ID', 'IF (all match)', and 'THEN'. The first rule, ID 1, has an 'IF' condition set to 'Path...' where 'is /payments/*'. The 'THEN' section contains a single action: 'Forward to...' with a target group weight of 0-999. A dropdown menu lists 'billings-target-group', 'homepage-target-group', and 'payments-target-group', with 'payments-target-group' highlighted and selected.

Below the rule table, there is a summary row for the 'HTTP 80: default action' which states 'Requests otherwise not routed'.

The bottom of the screen shows standard browser controls, a taskbar with various application icons, and a system status bar indicating 'ENG IN', '11:27 AM', '2/24/2022', and a battery level of 1.

- Select target group for payments



- Now click on save

The screenshot shows the AWS Lambda function configuration page. The top navigation bar includes tabs for 'Overview', 'Code', 'Triggers', 'Logs', 'Metrics', 'Actions', and 'Configuration'. The main content area is titled 'Handler' and contains the following details:

- Handler:** index.handler
- File:** /var/task/index.js
- Runtime:** Node.js 14.x
- Memory:** 128 MB
- Timeout:** 3.5 seconds
- Environment variables:** None
- Log group:** /aws/lambda/test-function
- Log level:** INFO

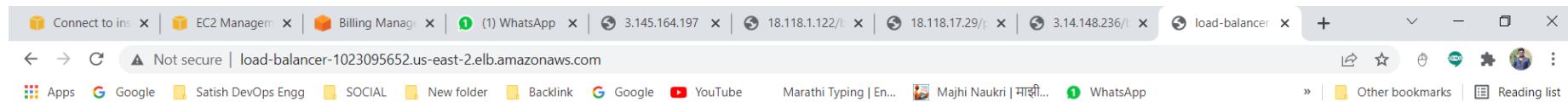
Below the Handler section, there is a note: "This function has no triggers defined. To start receiving events, add triggers or update the function's code." A large blue button at the bottom right says "Create trigger".

- Repeat the steps for billings as well and click on save

- Now we will check our HTML Application by using Load balancer DNS

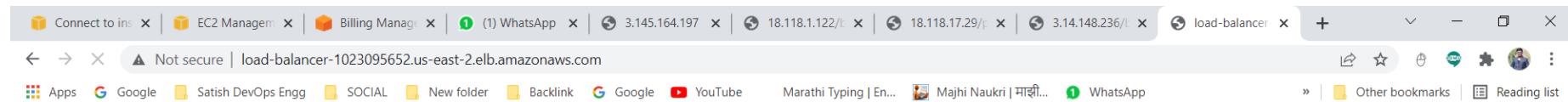
The screenshot shows the AWS Elastic Load Balancing (ELB) service in the AWS Management Console. The left sidebar lists various services like EC2 Dashboard, Instances, Images, and Elastic Block Store. The main content area displays a table of existing load balancers. One specific load balancer, named 'load-balancer', is selected and its configuration details are shown in the right panel. The 'Basic Configuration' section includes fields for Name (load-balancer), ARN (arn:aws:elasticloadbalancing:us-east-2:876283541003:loadbalancer/app/load-balancer/62c3705b7fda6fac), DNS name (load-balancer-1023095652.us-east-2.elb.amazonaws.com (A Record)), State (Active), Type (application), Scheme (internet-facing), and IP address type (ipv4). The 'DNS name' field is highlighted with a red box.

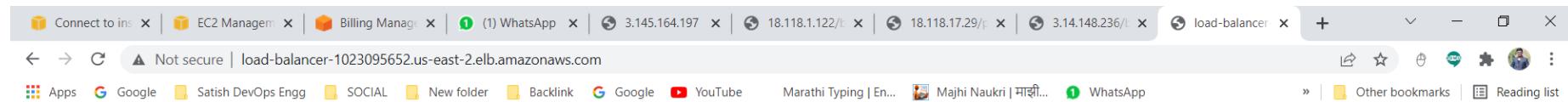
- Copy the DNS and Paste into your Browser

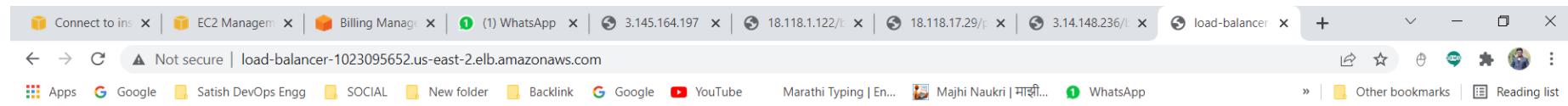


This is homepage 1





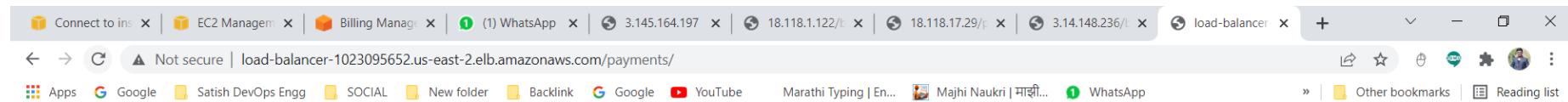




This is homepage 4

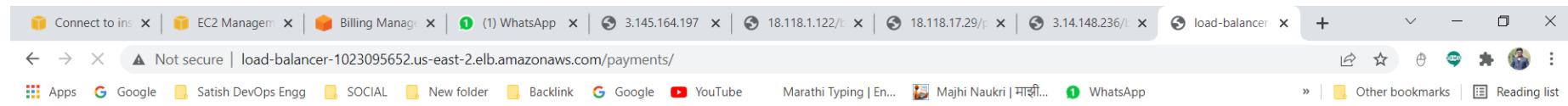


- Now check for payments



This is Payment Page A

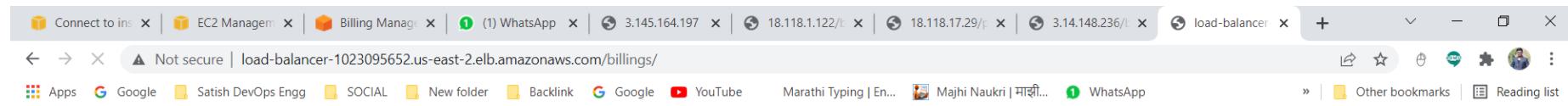


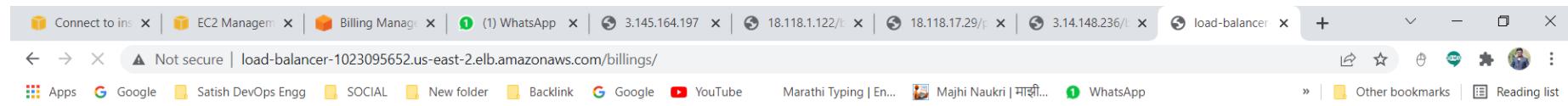


This is Payment Page B



- Now check for billings





- In this way we can use Application load Balancer to route Traffic according to application header.

Thanks for doing practical with us