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

Apps ★ Bookmarks Suggested Sites Prepaid Game Card... Online Shopping sit... CS:GO Stash Gmail YouTube (20+) Amin Sk | Fac... Eam - Freeskins.com WhatsApp Online Shopping Si... Maps Collections of HDQ...

aws   Services ▾   Resource Groups ▾ ★   AminSk007 ▾   Oregon ▾   Support ▾

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: launch-wizard-3

Description: launch-wizard-3 created 2020-08-23T22:14:49.593+05:30

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ		Description ⓘ	
All traffic ▾	All	0 - 65535	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

---

us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#Instances:sort=instanceId

Apps ★ Bookmarks Suggested Sites Prepaid Game Card... Online Shopping sit... CS:GO Stash Gmail YouTube (20+) Amin Sk | Fac... Eam - Freeskins.com WhatsApp Online Shopping Si... Maps Collections of HDQ...

aws   Services ▾   Resource Groups ▾ ★   AminSk007 ▾   Oregon ▾   Support ▾

Savings Plans
Reserved Instances
Dedicated Hosts New
Scheduled Instances
Capacity Reservations

▼ Images
  AMIs

▼ Elastic Block Store
  Volumes
  Snapshots
  Lifecycle Manager

▼ Network & Security
  Security Groups New
  Elastic IPs New
  Placement Groups New
  Key Pairs New
  Network Interfaces

▼ Load Balancing
  Load Balancers
  Target Groups New

▼ Auto Scaling
  Launch Configurations
  Auto Scaling Groups

Launch Instance ▾   Connect   Actions ▾

Q Filter by tags and attributes or search by keyword   1 to 2 of 2

	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP	IPv6 I	Key Name	Monitoring
	LetsUpgradeLinux	i-047df963999e3b54e	t2.micro	us-west-2c	🟢 running	✅ 2/2 checks ...	None	ec2-18-237-134-197.us...	18.237.134.197	-	LetsUpgradeSk	disabled
	LetsUpgradeLinux	i-0a86df5a8ac1a9379	t2.micro	us-west-2c	🟢 running	✅ 2/2 checks ...	None	ec2-54-185-124-105.us...	54.185.124.105	-	LetsUpgradeSk	disabled

Select an instance above

LetsUpgradeSk.pem ^   Show all   ✕

Select load balancer type

Elastic Load Balancing supports three types of load balancers: Application Load Balancers, Network Load Balancers (new), and Classic Load Balancers. Choose the load balancer type that meets your needs. Learn more about which load balancer is right for you

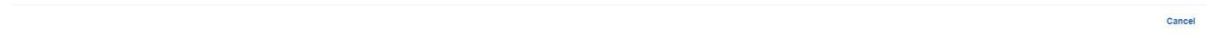| Application Load Balancer	Network Load Balancer	Classic Load Balancer
HTTP HTTPS	TCP TLS UDP	PREVIOUS GENERATION for HTTP, HTTPS, and TCP
Create	Create	Create
Choose an Application Load Balancer when you need a flexible feature set for your web applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers.	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 application. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies.	Choose a Classic Load Balancer when you have an existing application running in the EC2-Classic network.
Learn more >	Learn more >	Learn more >

Cancel

LetsUpgrade5k.pem   Show all   ×

---

1. Configure Load Balancer   2. Configure Security Settings   3. Configure Security Groups   4. Configure Routing   5. Register Targets   6. Review

Step 1: Configure Load Balancer

Basic Configuration

To configure your load balancer, provide a name, select a scheme, specify one or more listeners, and select a network. The default configuration is an Internet-facing load balancer in the selected network with a listener that receives HTTP traffic on port 80.

Name: LetsUpgradeelb
Scheme: ● internet-facing ○ internal
IP address type: ipv4

Listeners

A listener is a process that checks for connection requests, using the protocol and port that you configured.

Load Balancer Protocol	Load Balancer Port	
HTTP	80	✕

Add listener

Availability Zones

Specify the Availability Zones to enable for your load balancer. The load balancer routes traffic to the targets in these Availability Zones only. You can specify only one subnet per Availability Zone. You must specify subnets from at least two Availability Zones to increase the availability of your load balancer.

VPC: vpc-5d3b7e25 (172.31.0.0/16) (default)
Availability Zones:
☑ us-west-2a   subnet-7615e53c
   IPv4 address   Assigned by AWS
☑ us-west-2b   subnet-919345e9
   IPv4 address   Assigned by AWS
☐ us-west-2c   subnet-9e8524c3
☐ us-west-2d   subnet-a0ad84fb

Cancel   Next: Configure Security Settings

LetsUpgrade5k.pem   Show all   ×

us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#V2CreateELBWizard:type=application;

Apps ★ Bookmarks ⊕ Suggested Sites ⬛ Prepaid Game Card... ⬛ Online Shopping sit... CS:GO Stash ⬛ Gmail ▶ YouTube (20+) Amin Sk | Fac... Eam - Freeskins.com WhatsApp Online Shopping Si... Maps Collections of HDQ...

aws   Services ⌄   Resource Groups ⌄ ⭐   AminSk007 ⌄   Oregon ⌄   Support ⌄

1. Configure Load Balancer   2. Configure Security Settings   3. Configure Security Groups   4. Configure Routing   5. Register Targets   6. Review

Step 3: Configure Security Groups

A security group is a set of firewall rules that control the traffic to your load balancer. On this page, you can add rules to allow specific traffic to reach your load balancer. First, decide whether to create a new security group or select an existing one.

Assign a security group: ● Create a **new** security group
                           ○ Select an **existing** security group

Security group name: load-balancer-wizard-1

Description: load-balancer-wizard-1 created on 2020-08-23T22:25:00.323+05:30

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	
All traffic ⌄	All	0 - 65535	Anywhere ⌄ 0.0.0.0/0, ::/0	✖

Add Rule

Cancel   Previous   **Next: Configure Routing**

LetsUpgradeSk.pem   ^   Show all   ✕

---

us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#V2CreateELBWizard:type=application;

Apps ★ Bookmarks ⊕ Suggested Sites ⬛ Prepaid Game Card... ⬛ Online Shopping sit... CS:GO Stash ⬛ Gmail ▶ YouTube (20+) Amin Sk | Fac... Eam - Freeskins.com WhatsApp Online Shopping Si... Maps Collections of HDQ...

aws   Services ⌄   Resource Groups ⌄ ⭐   AminSk007 ⌄   Oregon ⌄   Support ⌄

1. Configure Load Balancer   2. Configure Security Settings   3. Configure Security Groups   4. Configure Routing   5. Register Targets   6. Review

Step 4: Configure Routing

Your load balancer routes requests to the targets in this target group using the protocol and port that you specify, and performs health checks on the targets using these health check settings. Note that each target group can be associated with only one load balancer.

Target group

Target group ⓘ   New target group ⌄

Name ⓘ   newtarget

Target type   ● Instance
              ○ IP
              ○ Lambda function

Protocol ⓘ   HTTP ⌄

Port ⓘ   80

Health checks

Protocol ⓘ   HTTP ⌄

Path ⓘ   /

▼ Advanced health check settings

Port ⓘ   ● traffic port
         ○ override

Healthy threshold ⓘ   5

Unhealthy threshold ⓘ   2

Timeout ⓘ   5   seconds

Interval ⓘ   30   seconds

Success codes ⓘ   200

Cancel   Previous   **Next: Register Targets**

LetsUpgradeSk.pem   ^   Show all   ✕

Step 5: Register Targets

Register targets with your target group. If you register a target in an enabled Availability Zone, the load balancer starts routing requests to the targets as soon as the registration process completes and the target passes the initial health checks.

Registered targets

To deregister instances, select one or more registered instances and then click Remove.

	Instance	Name	Port	State	Security groups	Zone
	i-0a86df5a8ac1a9379	LetsUpgradeLinux	80	● running	launch-wizard-3	us-west-2c
	i-047df963999e3b54e	LetsUpgradeLinux	80	● running	launch-wizard-3	us-west-2c

Instances

To register additional instances, select one or more running instances, specify a port, and then click Add. The default port is the port specified for the target group. If the instance is already registered on the specified port, you must specify a different port.

	Instance	Name	State	Security groups	Zone	Subnet ID	Subnet CIDR
■	i-0a86df5a8ac1a9379	LetsUpgradeLinux	● running	launch-wizard-3	us-west-2c	subnet-9e8524c3	172.31.0.0/20
■	i-047df963999e3b54e	LetsUpgradeLinux	● running	launch-wizard-3	us-west-2c	subnet-9e8524c3	172.31.0.0/20

Load Balancer Creation Status

✓ **Successfully created load balancer**
Load balancer LetsUpgradeelb was successfully created.
Note: It might take a few minutes for your load balancer to be fully set up and ready to route traffic, and for the targets to complete the registration process and pass the initial health checks.

Suggested next steps

- Discover other services that you can integrate with your load balancer. Visit the **Integrated services** tab within LetsUpgradeelb
- Consider using AWS Global Accelerator to further improve the availability and performance of your applications. AWS Global Accelerator console
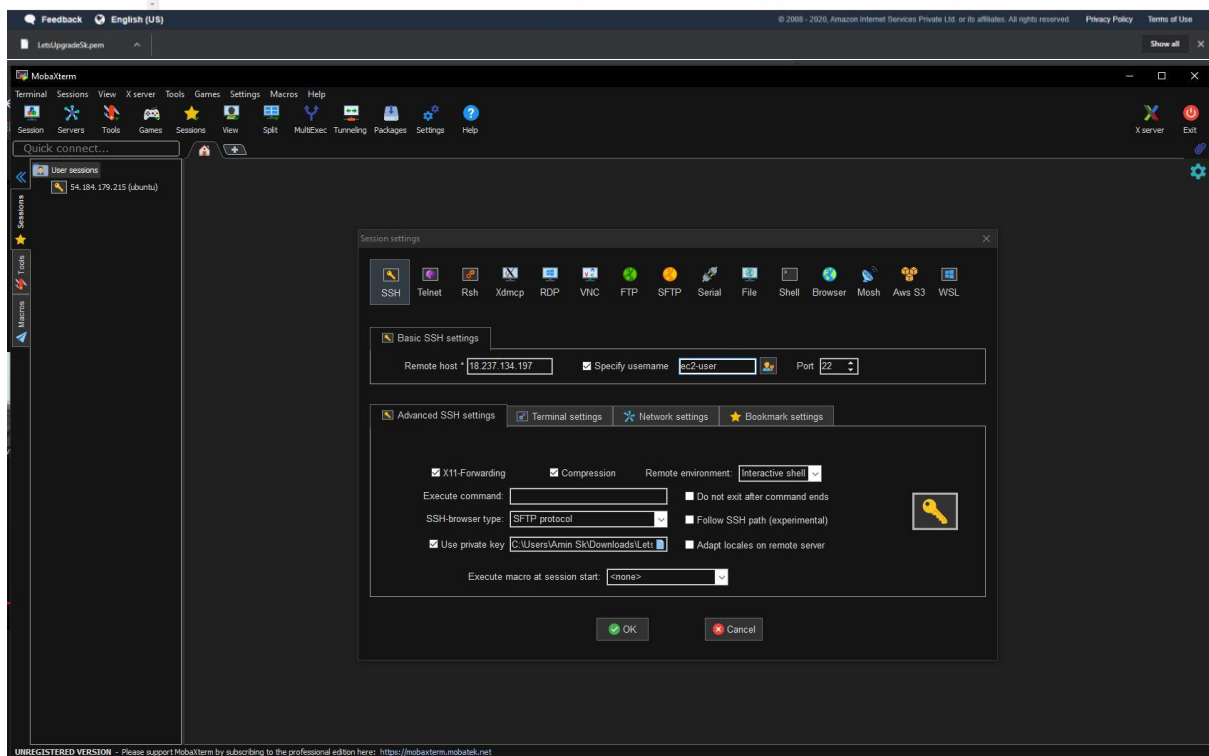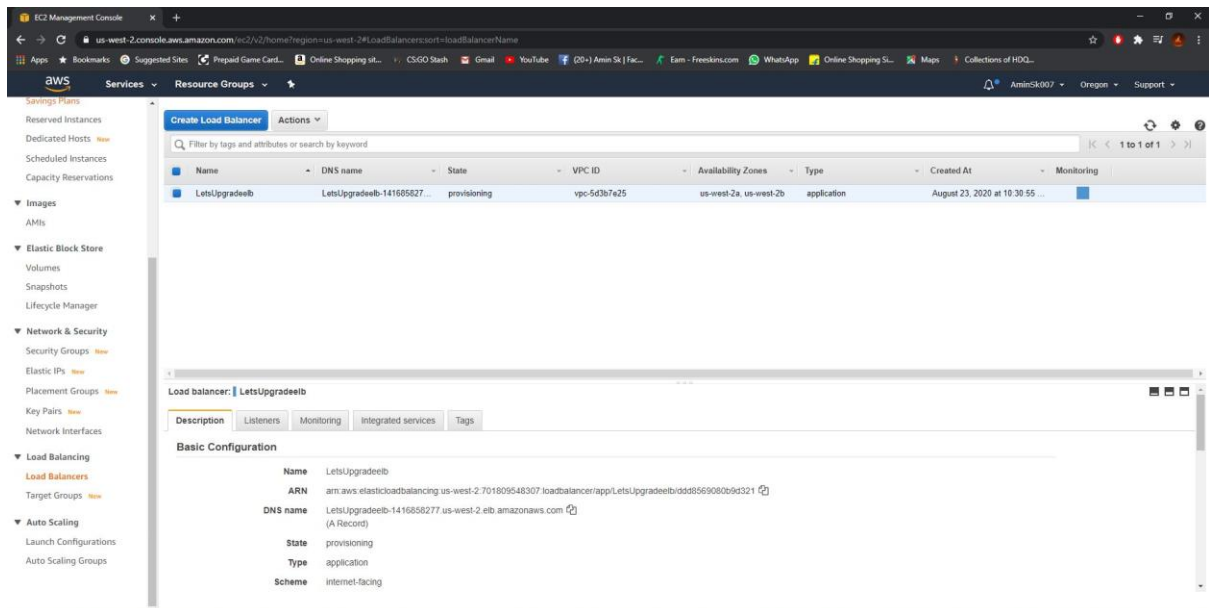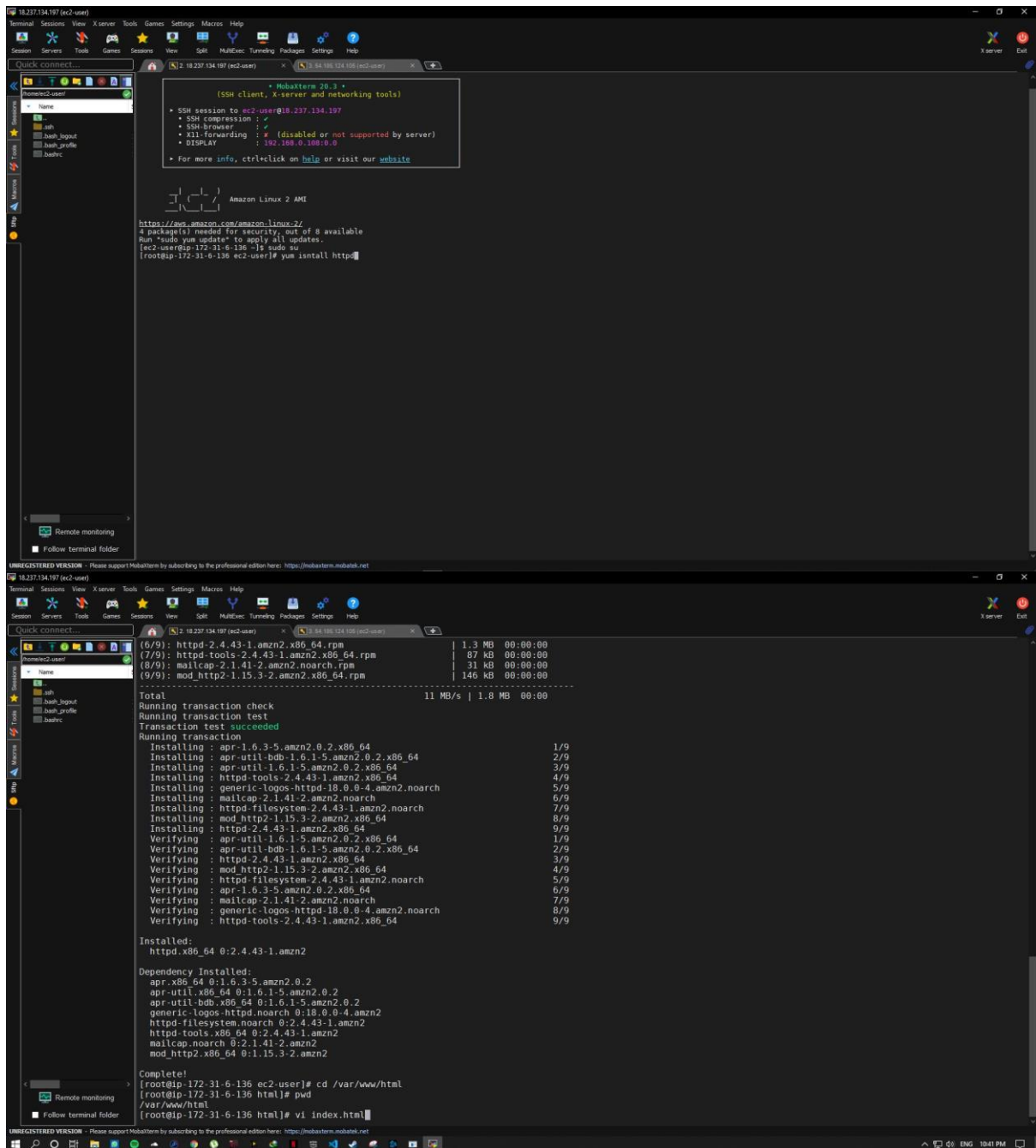
Terminal 1:

```
                    • MobaXterm 20.3 •
              (SSH client, X-server and networking tools)

► SSH session to ec2-user@18.237.134.197
  • SSH compression : ✔
  • SSH-browser     : ✔
  • X11-forwarding  : ✘  (disabled or not supported by server)
  • DISPLAY         : 192.168.0.108:0.0

► For more info, ctrl+click on help or visit our website


    _|  _|  )       Amazon Linux 2 AMI
   _|\_|_|

https://aws.amazon.com/amazon-linux-2/
4 package(s) needed for security, out of 8 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-6-136 ~]$ sudo su
[root@ip-172-31-6-136 ec2-user]# yum isntall httpd
```

Terminal 2:

```
(6/9): httpd-2.4.43-1.amzn2.x86_64.rpm                    | 1.3 MB  00:00:00
(7/9): httpd-tools-2.4.43-1.amzn2.x86_64.rpm             |  87 kB  00:00:00
(8/9): mailcap-2.1.41-2.amzn2.noarch.rpm                |  31 kB  00:00:00
(9/9): mod_http2-1.15.3-2.amzn2.x86_64.rpm              | 146 kB  00:00:00
--------------------------------------------------------------------------------
Total                                         11 MB/s | 1.8 MB  00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : apr-1.6.3-5.amzn2.0.2.x86_64                         1/9
  Installing : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64                2/9
  Installing : apr-util-1.6.1-5.amzn2.0.2.x86_64                    3/9
  Installing : httpd-tools-2.4.43-1.amzn2.x86_64                    4/9
  Installing : generic-logos-httpd-18.0.0-4.amzn2.noarch            5/9
  Installing : mailcap-2.1.41-2.amzn2.noarch                        6/9
  Installing : httpd-filesystem-2.4.43-1.amzn2.noarch               7/9
  Installing : mod_http2-1.15.3-2.amzn2.x86_64                      8/9
  Installing : httpd-2.4.43-1.amzn2.x86_64                          9/9
  Verifying  : apr-util-1.6.1-5.amzn2.0.2.x86_64                    1/9
  Verifying  : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64                2/9
  Verifying  : httpd-2.4.43-1.amzn2.x86_64                          3/9
  Verifying  : mod_http2-1.15.3-2.amzn2.x86_64                      4/9
  Verifying  : httpd-filesystem-2.4.43-1.amzn2.noarch               5/9
  Verifying  : apr-1.6.3-5.amzn2.0.2.x86_64                         6/9
  Verifying  : mailcap-2.1.41-2.amzn2.noarch                        7/9
  Verifying  : generic-logos-httpd-18.0.0-4.amzn2.noarch            8/9
  Verifying  : httpd-tools-2.4.43-1.amzn2.x86_64                    9/9

Installed:
  httpd.x86_64 0:2.4.43-1.amzn2

Dependency Installed:
  apr.x86_64 0:1.6.3-5.amzn2.0.2
  apr-util.x86_64 0:1.6.1-5.amzn2.0.2
  apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2
  generic-logos-httpd.noarch 0:18.0.0-4.amzn2
  httpd-filesystem.noarch 0:2.4.43-1.amzn2
  httpd-tools.x86_64 0:2.4.43-1.amzn2
  mailcap.noarch 0:2.1.41-2.amzn2
  mod_http2.x86_64 0:1.15.3-2.amzn2

Complete!
[root@ip-172-31-6-136 ec2-user]# cd /var/www/html
[root@ip-172-31-6-136 html]# pwd
/var/www/html
[root@ip-172-31-6-136 html]# vi index.html
```

```html
<form action="action_page.php" method="post">
<div class="imgcontainer">
<img src="img_avatar2.png" alt="Avatar" class="avatar">
</div>

<div class="container">
<label for="uname"><b>Username</b></label>
<input type="text" placeholder="Enter Username" name="uname" required>


<label for="psw"><b>Password</b></label>
<input type="password" placeholder="Enter Password" name="psw" required>


<button type="submit">Login</button>


<label>
<input type="checkbox" checked="checked" name="remember"> Remember me
</label>
</div>


<div class="container" style="background-color:#f1f1f1">
<button type="button" class="cancelbtn">Cancel</button>
<span class="psw">Forgot <a href="#">password?</a></span>
</div>
</form>
```

-- INSERT --

```html
<form action="action_page.php" method="post">
<div class="imgcontainer">
<img src="img_avatar2.png" alt="Avatar" class="avatar">
</div>

<div class="container">
<label for="uname"><b>Username</b></label>
<input type="text" placeholder="Enter Username" name="uname" required>


<label for="psw"><b>Password</b></label>
<input type="password" placeholder="Enter Password" name="psw" required>


<button type="submit">Login</button>


<label>
<input type="checkbox" checked="checked" name="remember"> Remember me
</label>
</div>


<div class="container" style="background-color:#f1f1f1">
<button type="button" class="cancelbtn">Cancel</button>
<span class="psw">Forgot <a href="#">password?</a></span>
</div>
</form>
```

:wq

```
generic-logos-httpd.noarch 0:18.0.0-4.amzn2
httpd-filesystem.noarch 0:2.4.43-1.amzn2
httpd-tools.x86_64 0:2.4.43-1.amzn2
mailcap.noarch 0:2.1.41-2.amzn2
mod_http2.x86_64 0:1.15.3-2.amzn2

Complete!
[root@ip-172-31-6-136 ec2-user]# cd /var/www/html
[root@ip-172-31-6-136 html]# pwd
/var/www/html
[root@ip-172-31-6-136 html]# vi index.html
[root@ip-172-31-6-136 html]# more index.html
<form action="action_page.php" method="post">
<div class="imgcontainer">
<img src="img_avatar2.png" alt="Avatar" class="avatar">
</div>

<div class="container">
<label for="uname"><b>Username</b></label>
<input type="text" placeholder="Enter Username" name="uname" required>

<label for="psw"><b>Password</b></label>
<input type="password" placeholder="Enter Password" name="psw" required>

<button type="submit">Login</button>

<label>
<input type="checkbox" checked="checked" name="remember"> Remember me
</label>
</div>

<div class="container" style="background-color:#f1f1f1">
<button type="button" class="cancelbtn">Cancel</button>
<span class="psw">Forgot <a href="#">password?</a></span>
</div>
</form>

[root@ip-172-31-6-136 html]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-6-136 html]#
```


Avatar
Username Enter Username   **Password** Enter Password   Login   ☑ Remember me
Cancel | Forgot password?

Top terminal window (vi editor - INSERT mode):

```
<form action="action_page.php" method="post">
<div class="imgcontainer">
<img src="img_avatar2.png" alt="Avatar" class="avatar">
</div>

<div class="container">
<label for="uname"><b>UserID</b></label>
<input type="text" placeholder="Enter Username" name="uname" required>

<label for="psw"><b>Passkey</b></label>
<input type="password" placeholder="Enter Password" name="psw" required>

<button type="submit">Login</button>

<label>
<input type="checkbox" checked="checked" name="remember"> Remember me
</label>
</div>

<div class="container" style="background-color:#f1f1f1">
<button type="button" class="cancelbtn">Cancel</button>
<span class="psw">Forgot <a href="#">password?</a></span>
</div>
</form>

-- INSERT --
```

Bottom terminal window:

```
Verifying   : httpd-filesystem-2.4.43-1.amzn2.noarch                    5/9
Verifying   : apr-1.6.3-5.amzn2.0.2.x86_64                              6/9
Verifying   : mailcap-2.1.41-2.amzn2.noarch                            7/9
Verifying   : generic-logos-httpd-18.0.0-4.amzn2.noarch                8/9
Verifying   : httpd-tools-2.4.43-1.amzn2.x86_64                        9/9

Installed:
  httpd.x86_64 0:2.4.43-1.amzn2

Dependency Installed:
  apr.x86_64 0:1.6.3-5.amzn2.0.2       apr-util.x86_64 0:1.6.1-5.amzn2.0.2  apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2  generic-logos-httpd.noarch 0:18.0.0-4.amzn2  httpd-filesystem.noarch 0:2.4.43-1.amzn2
  httpd-tools.x86_64 0:2.4.43-1.amzn2  mailcap.noarch 0:2.1.41-2.amzn2       mod_http2.x86_64 0:1.15.3-2.amzn2

Complete!
[root@ip-172-31-5-169 ec2-user]# cd /var/www/html
[root@ip-172-31-5-169 html]# pwd
/var/www/html
[root@ip-172-31-5-169 html]# vi index.html
[root@ip-172-31-5-169 html]# more index.html
<form action="action_page.php" method="post">
<div class="imgcontainer">
<img src="img_avatar2.png" alt="Avatar" class="avatar">
</div>

<div class="container">
<label for="uname"><b>UserID</b></label>
<input type="text" placeholder="Enter Username" name="uname" required>

<label for="psw"><b>Passkey</b></label>
<input type="password" placeholder="Enter Password" name="psw" required>

<button type="submit">Login</button>

<label>
<input type="checkbox" checked="checked" name="remember"> Remember me
</label>
</div>

<div class="container" style="background-color:#f1f1f1">
<button type="button" class="cancelbtn">Cancel</button>
<span class="psw">Forgot <a href="#">password?</a></span>
</div>
</form>

[root@ip-172-31-5-169 html]# sevice start httpd
bash: sevice: command not found
[root@ip-172-31-5-169 html]# service start httpd
The service command supports only basic LSB actions (start, stop, restart, try-restart, reload, force-reload, status). For other actions, please try to use systemctl.
[root@ip-172-31-5-169 html]# start httpd.service
bash: start: command not found
[root@ip-172-31-5-169 html]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-5-169 html]#
```

503 Service Temporarily Unavailable