LAB 1 (AWS getting started and fundamentals of computer communications)

Fatema Mirza

D7001D Network Programming and Distributed Applications

LAB 1 (AWS getting started and fundamentals of computer communications)

by

Fatema Mirza

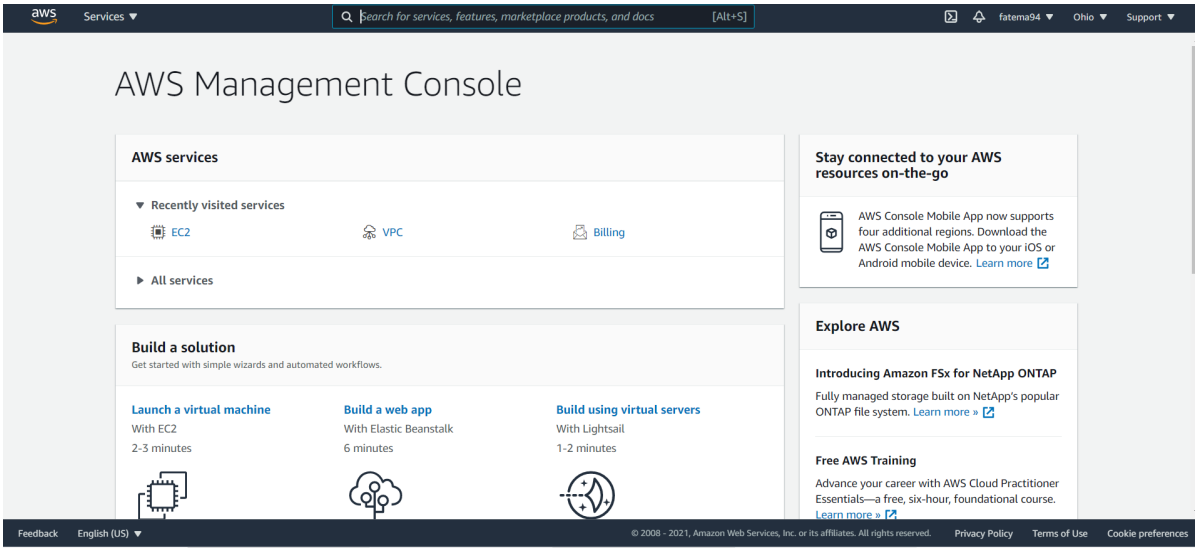
Submission Deadline:	September 10, 2021	
Teachers:	Dr. Evgeny Osipov,	LULEÅ TEKNISKA UNIVERSITET
	Ahmed Afif Monrat,	LULEÅ TEKNISKA UNIVERSITET

Contents

1

Part I -AWS Management Console

AWS account was created with the personal account. After registering and logging in as the root user with the registered email address, the following window appears with the username fatema94. Below is an image of the AWS account successfully created.

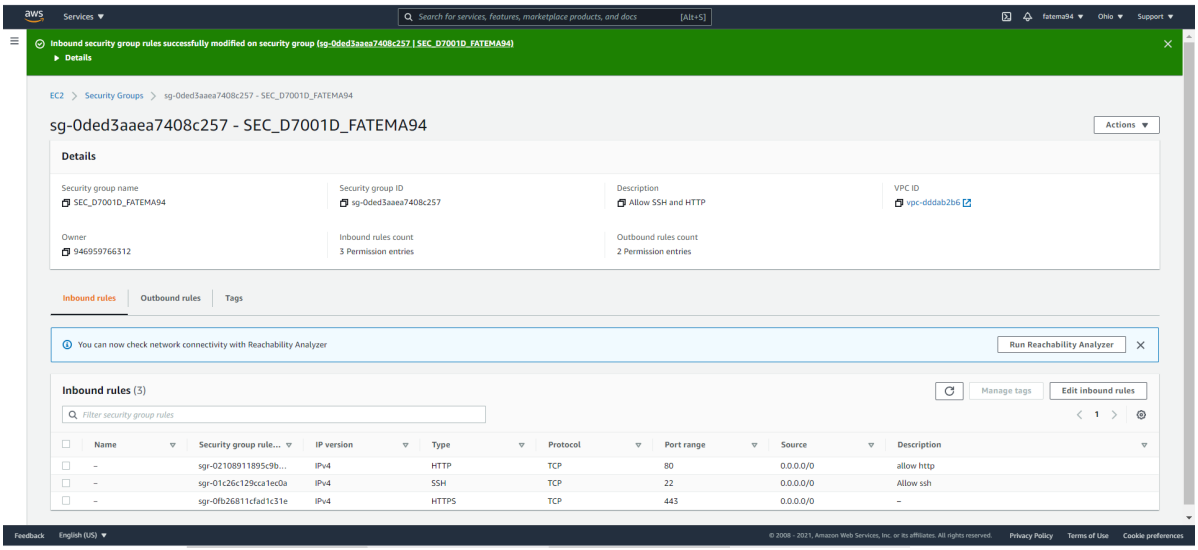


2

Part II -Get your hands dirty

2.1. Create security group: SEC_D7001D_YOURLOGINNAME. Configure it to allow SSH and HTTP.

Security group: SEC_D7001D_FATEMA94 is created with security rules allowing SSH and HTTP/HTTPS. The image below demonstrates the creation of the Security group: SEC_D7001D_FATEMA94 with the appropriate rules.

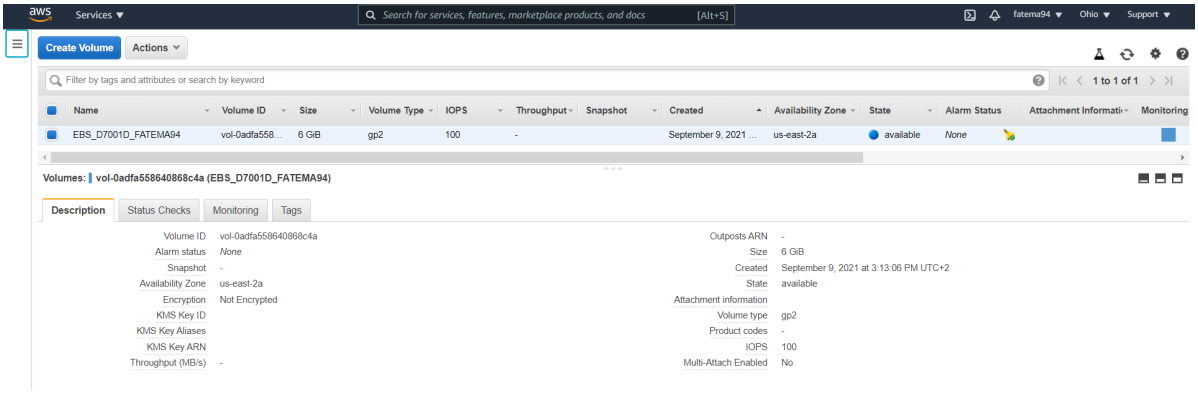


2.1.1. QUESTION 1: What is the purpose of security groups in AWS cloud?

Security groups provide a virtual firewall to manage the traffic that are allowed to enter and exit instances through allowed protocols and/or ports by the creation of various inbound and outbound rules (Checkpoint.com, 2021).

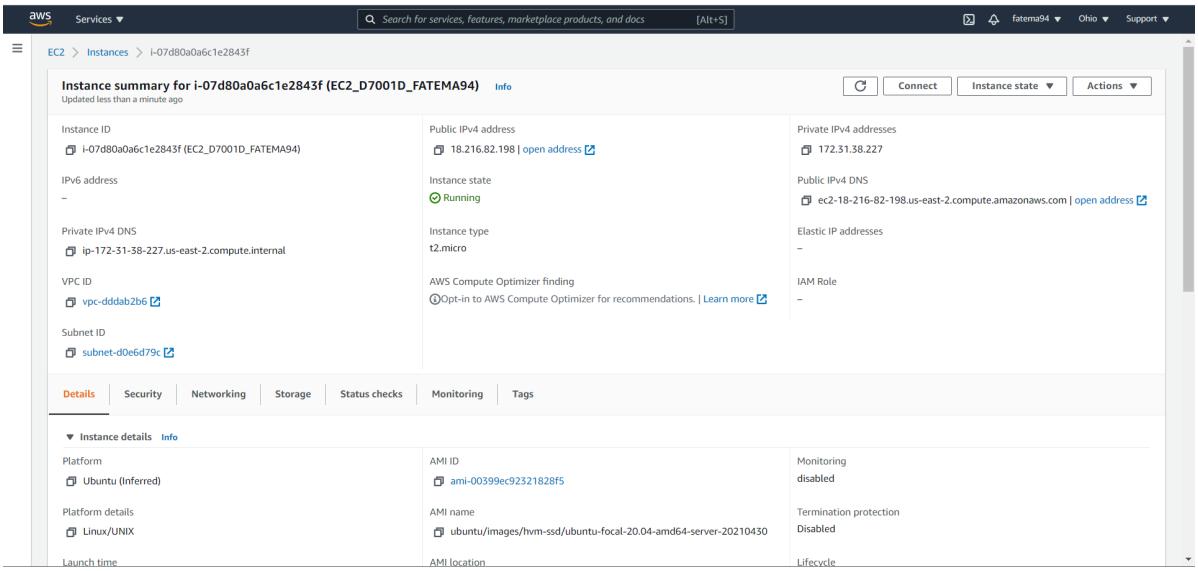
2.2. Create EBS volume with name EBS_D7001D_YOURUSERNAME.

Volume with the name EBS_D7001D_FATEMA94 is created with a size of 6 GiB as demonstrated in the image. EBS stands for Amazon Elastic Block Store and is used to implement scalable block level persistent storage (stored even when the the instance is no longer running) associated with the EC2 instances (aws, 2021).

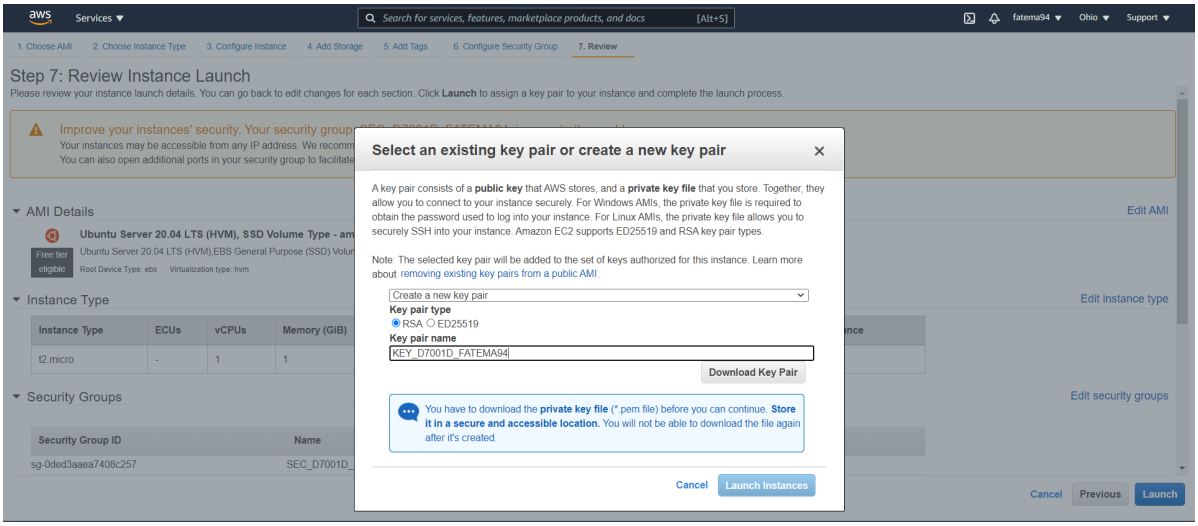


2.3. Create and launch AWS Instance with name EC2_D7001D_YOURUSERNAME. Note that the key pair MUST be tagged as KEY_D7001D_YOURUSERNAME and be associated with YOUR security group.

Instance with the name EC2_D7001D_FATEMA94 is created as shown in the image below.



The instance is associated with the key KEY_D7001D_FATEMA94.



2.3.1. QUESTION 2: What type of instance you created?

t2.micro type of instance is created which is an instance available in the free tier.

2.3.2. QUESTION 3: Which AMI you selected, motivate your choice.

Ubuntu Server 20.04 LTS (HVM), SSD Volume Type is selected because:

1. Ubuntu Server 20.04 LTS falls under the AMIs available in the free tier option.

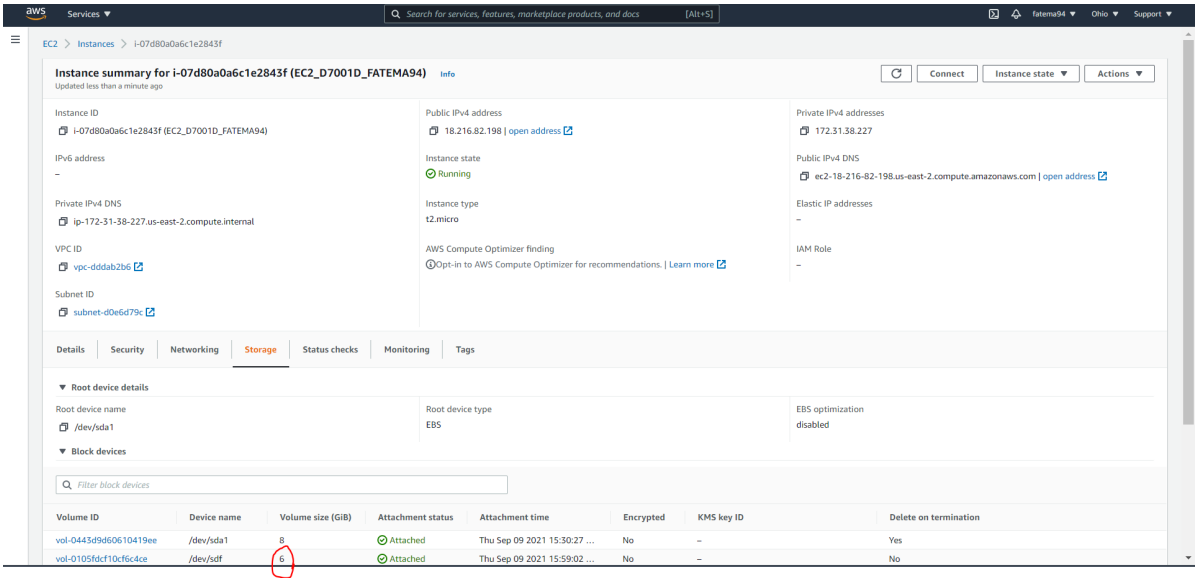
2. Ease of use with Ubuntu as the user is familiar with Ubuntu CLI.

3. Ubuntu performed superior compared to other Linux based operating systems on GraphicsMagick benchmarking tests (which include GraphicsMagick tests and Compilation time tests) (Phoronix, 2012).

4. Linux2 AMI is less stable as it is a rolling distro (or release) - hence making it less manageable than Ubuntu.

2.4. Attach your EBS_D7001D_YOURUSERNAME EBS to the instance. Note: store all your working files MUST BE INSIDE THIS EBS. Make sure you have unchecked "delete on termination" option!

The image below shows that the volume has been attached to the instance (marked in red).



2.4.1. QUESTION 4: Which file system is configured on your volume?

GP2 (General Purpose SSD) file system is configured on the volume.

2.4.2. QUESTION 5: Can you change it?

Yes, it can be changed following the instructions mentioned in (Behera, 2021).

2.5. Log in to your EC2_D7001D_YOURUSERNAME instance.

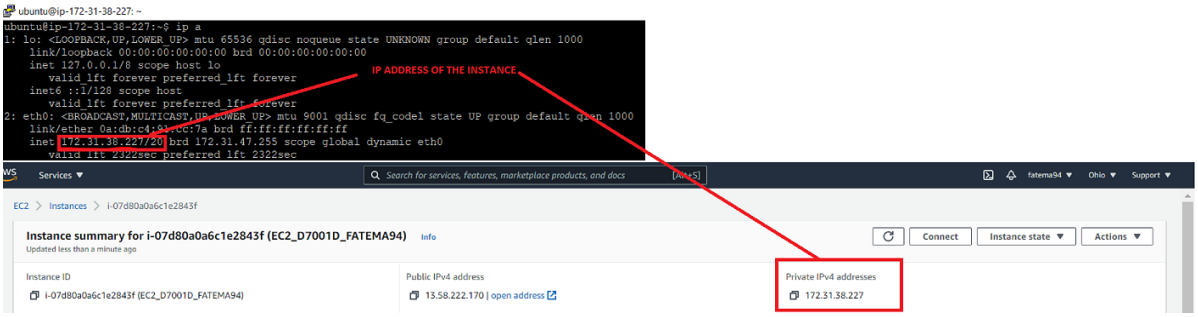
Thie instance EC2_D7001D_FATEMA94 has been logged into using the putty ssh technique (utilizing the key and the hostname) as shown below. The instance can also be logged in using the built-in instance connect.

ubuntu@ip-172-31-38-227: ~

```
Using username "ubuntu".
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-1045-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Thu Sep  9 14:17:34 UTC 2021

  System load:  0.0               Processes:             100
  Usage of /:   16.5% of 7.69GB   Users logged in:       0
  Memory usage: 22%               IPv4 address for eth0: 172.31.38.227
  Swap usage:   0%


1 update can be applied immediately.
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your
Internet connection or proxy settings


Last login: Thu Sep  9 14:07:58 2021 from 3.16.146.2
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-38-227:~$ whoami
ubuntu
ubuntu@ip-172-31-38-227:~$
```

2.5.1. QUESTION 6: What is ip address of your instance?

IP address of the instance EC2_D7001D_FATEMA94 is: **172.31.38.227** which is marked in red in the image. This is found with the use of the command in the CLI ***ip a*** or can be found from going into the instance properties as demonstrated in the image. .



2.5.2. QUESTION 7: What is its public and its private dns name?

Public dns name of the instance EC2_D7001D_FATEMA94 is: **ec2-18-216-82-198.us-east-2.compute.amazonaws.com**. This can be retrieved using the command entered in the CLI ***curl http://169.254.169.254/latest/meta-data/public-hostname; echo***.

ubuntu@ip-172-31-38-227: ~

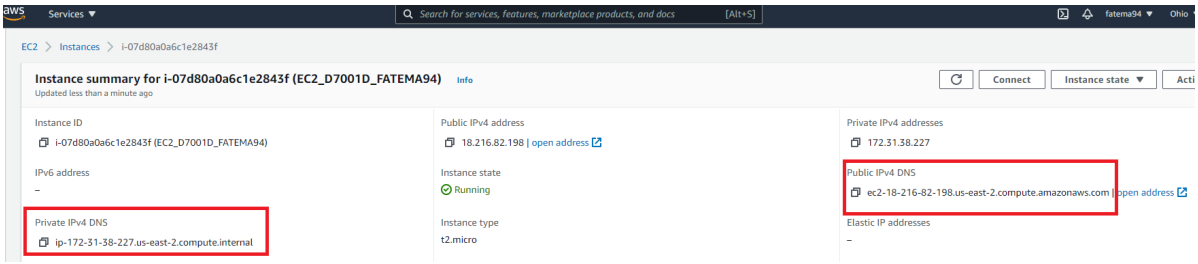
```
ubuntu@ip-172-31-38-227:~$ curl http://169.254.169.254/latest/meta-data/public-hostname; echo
ec2-18-216-82-198.us-east-2.compute.amazonaws.com
ubuntu@ip-172-31-38-227:~$
```

Private dns name of the instance EC2_D7001D_FATEMA94 is: **ip-172-31-38-227.us-east-2.compute.internal**. This can be retrieved using the command entered in the CLI ***curl http://169.254.169.254/latest/meta-data/local-hostname; echo***.

ubuntu@ip-172-31-38-227: ~

```
ubuntu@ip-172-31-38-227:~$ curl http://169.254.169.254/latest/meta-data/local-ho
stname; echo
ip-172-31-38-227.us-east-2.compute.internal
ubuntu@ip-172-31-38-227:~$
```

The IP addresses can also be verified from the instance properties as well where the public DNS is ec2-18-216-82-198.us-east-2.compute.amazonaws.com and private DNS is ip-172-31-38-227.us-east-2.compute.internal



2.6. Install apache server on your instance (For Ubuntu: sudo apt-get install apache2)

Ensure that Apache has been correctly installed on the instance EC2_D7001D_FATEMA94.

```
root@ip-172-31-38-227:/home/ubuntu# sudo apt-get install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libjansson4 liblua5.2-0 ssl-cert
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser openssl-blacklist
```

Once all the relevant packages has been downloaded and updated, check to see whether the apache server is activated using the command ***sudo systemctl status apache2***.

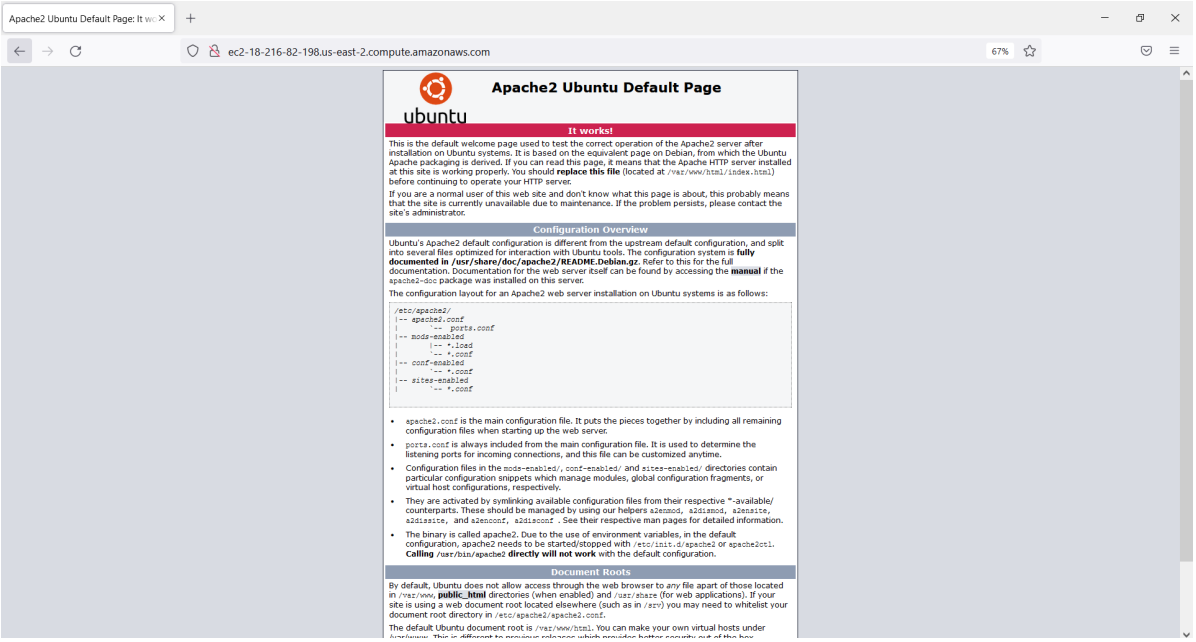
```
root@ip-172-31-38-227:/home/ubuntu# sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2021-09-09 14:45:14 UTC; 1min 1s ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2974 (apache2)
      Tasks: 55 (limit: 1160)
     Memory: 5.2M
     CGroup: /system.slice/apache2.service
             ├─2974 /usr/sbin/apache2 -k start
             ├─2976 /usr/sbin/apache2 -k start
             └─2977 /usr/sbin/apache2 -k start

Sep 09 14:45:14 ip-172-31-38-227 systemd[1]: Starting The Apache HTTP Server...
Sep 09 14:45:14 ip-172-31-38-227 systemd[1]: Started The Apache HTTP Server.
```

2.6.1. QUESTION 8: What is the public address on your server?

The public address on the server is: **18.216.82.198**. This can be retrieved using the command entered in the CLI ***curl ifconfig.me; echo*** or viewed from the instance of the server.

2.6.2. QUESTION 9: What text have been shown when you open public dns name in web browser?

The default welcome page of apache2 on ubuntu is displayed when the public dns name of the instance EC2_D7001D_FATEMA94 is copied in the web browser.



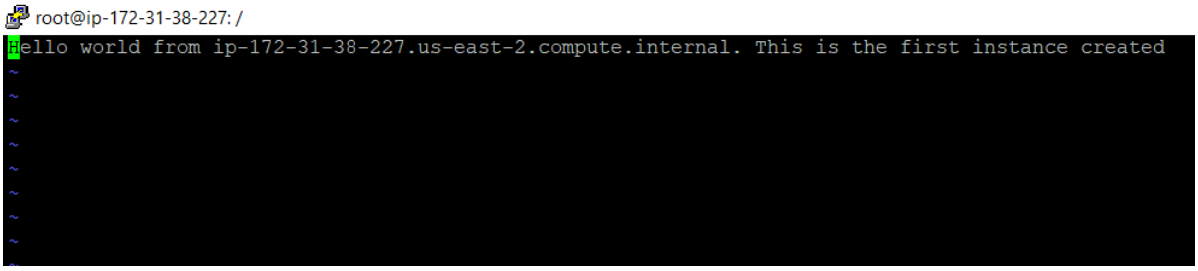
2.7. Now edit /var/www/index.html and enter text so that you can distinguish this instance from others.When you are done go to AWS console, select your instance and choose launch more like this.

To edit the index.html file located at /var/www/html/, type the following command in CLI mode ***echo "Hello world from $(hostname -f). This is the first instance created" > /var/www/html/index.html***

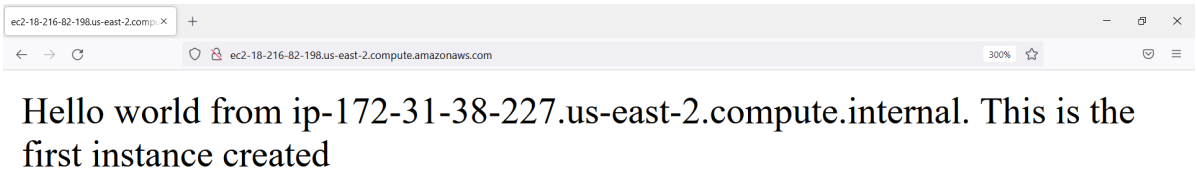
```
root@ip-172-31-38-227:/# echo "Hello world from $(hostname -f). This is the first instance created" > /var/www/html/index.html
root@ip-172-31-38-227:/# sudo vi  /var/www/html/index.html
root@ip-172-31-38-227:/#
```

To ensure that the changes have been reflected, either check the index.html file using command in CLI mode ***sudo vi /var/www/html/index.html*** or copy the public dns of the instance EC2_D7001D_FATEMA94 and view it on the web browser.

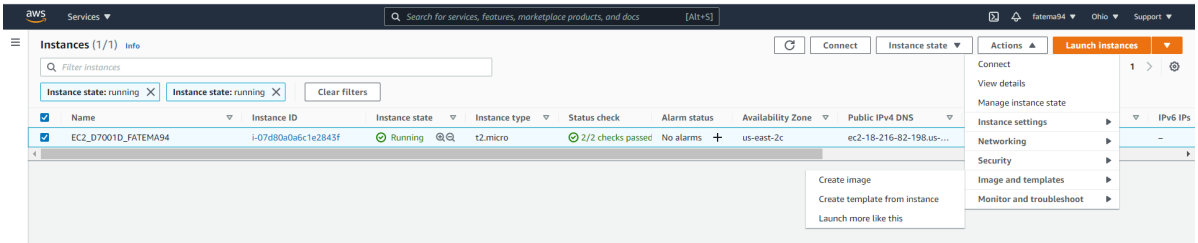
Below is the image from the vim platform for index.html.



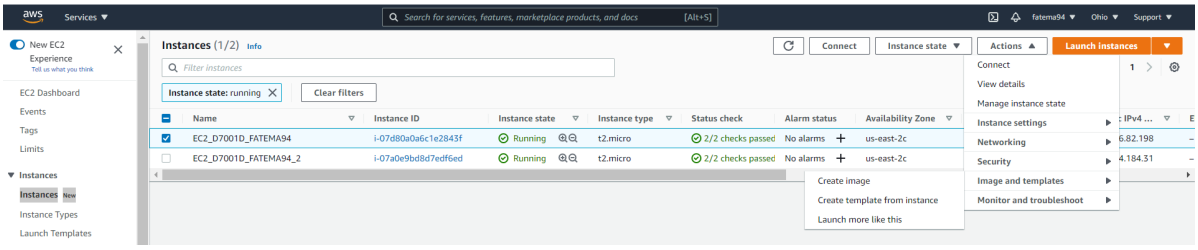
Below is the image of what is displayed when index.html is viewed from the browser.



To create another instance use the option ***launch more like this*** from Actions > Image and templates :



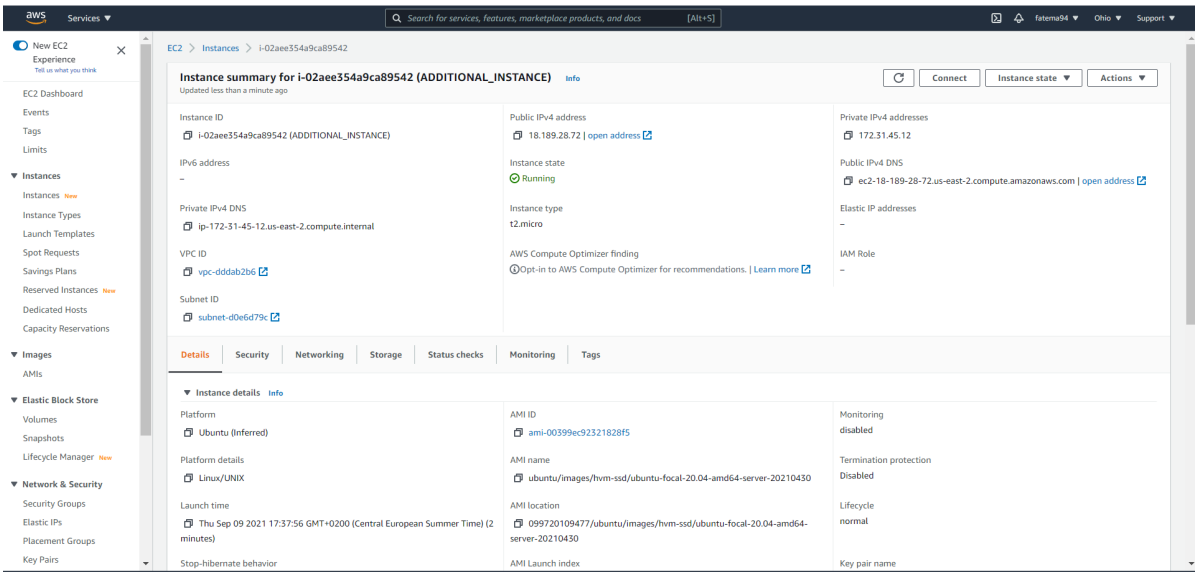
On creation, the following can be seen on the instances dashboard where the name of the new instance has been renamed as EC2_D7001D_FATEMA94_2 to distinguish it from the original EC2_D7001D_FATEMA94.



2.8. Launch one additional instance. Copy its public dns and paste it to web browser.

The new instance is named as ADDITIONAL_INSTANCE to distinguish it from the rest and its properties are displayed in the image below.

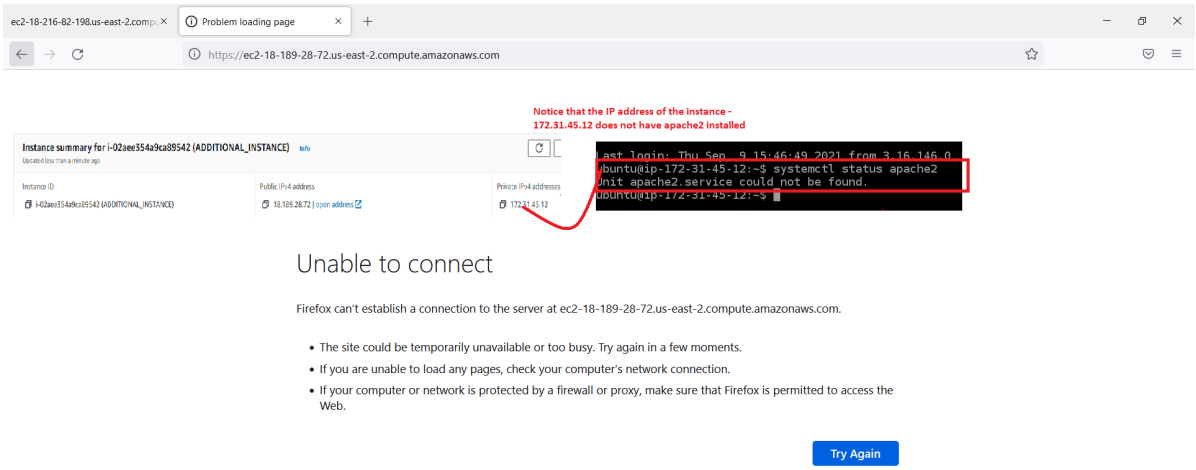
2.9. Stop this instance and change the name of the instance to: “delete-me-username”. Now select the instance where your webserver is running and create AMI image 15_LP1_AMI_D7001D_YOURUSERNAME. 9



2.8.1. QUESTION 10: What was server response?

When the public dns of new instance ADDITIONAL_INSTANCE

ec2-18-189-28-72.us-east-2.compute.amazonaws.com is copied into the web browser, the following is viewed - that is the instance is unable to connect to the server.

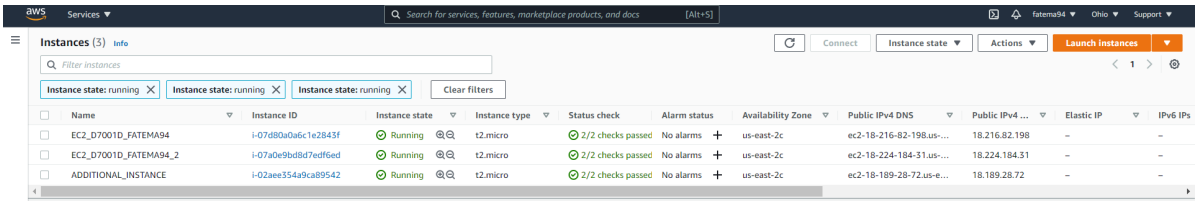


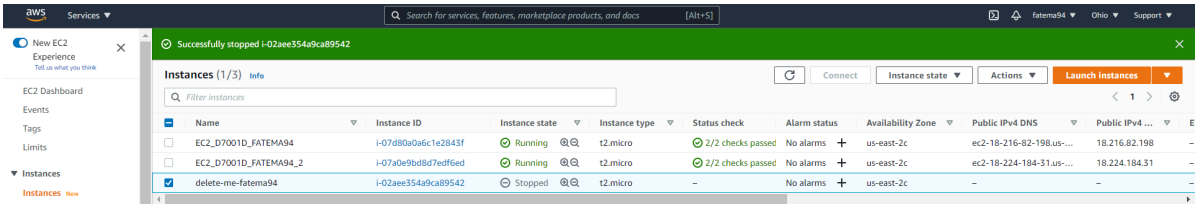
2.8.2. QUESTION 11: Explain why.

This is because we created a new instance from scratch where apache2 is not installed or activated, therefore none of the web services provided by apache2 are accessible.

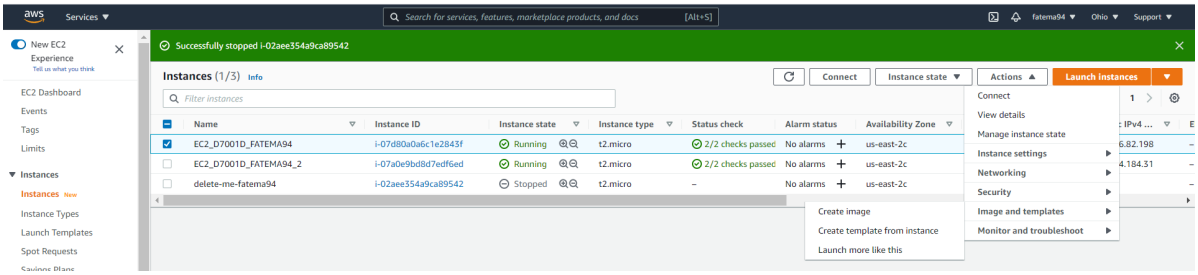
2.9. Stop this instance and change the name of the instance to: “delete-me-username”. Now select the instance where your webserver is running and create AMI image 15_LP1_AMI_D7001D_YOURUSERNAME.

The instance ADDITIONAL_INSTANCE is stopped and renamed as delete-me-fatema94.

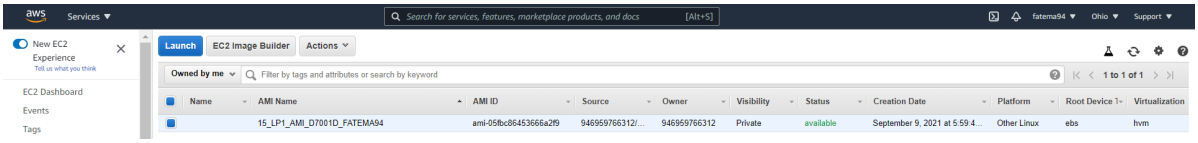




To create the AMI 15_LP1_AMI_D7001D_FATEMA94, select the create image from Actions > Create Image:

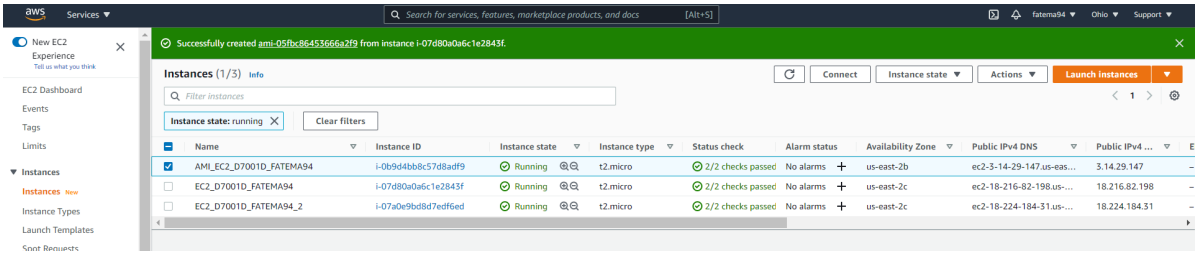


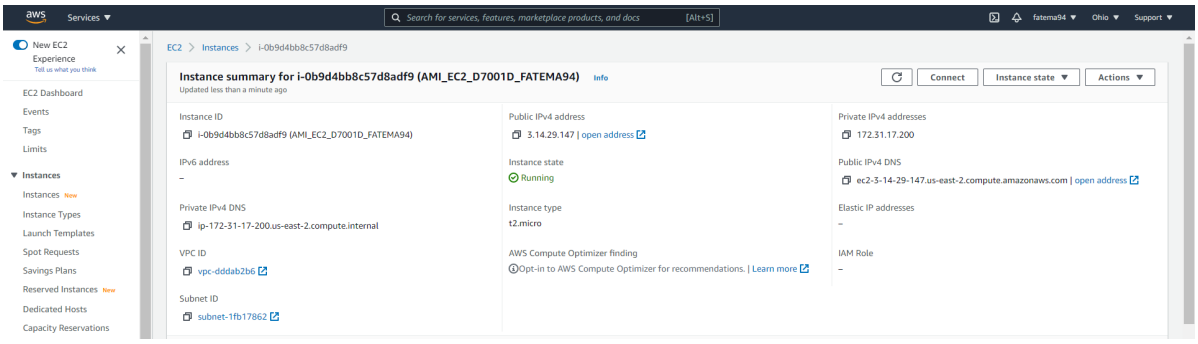
Wait until the AMI has been created and the status is available.



2.10. Launch a new instance from this image. Copy public dns and paste it to web browser.

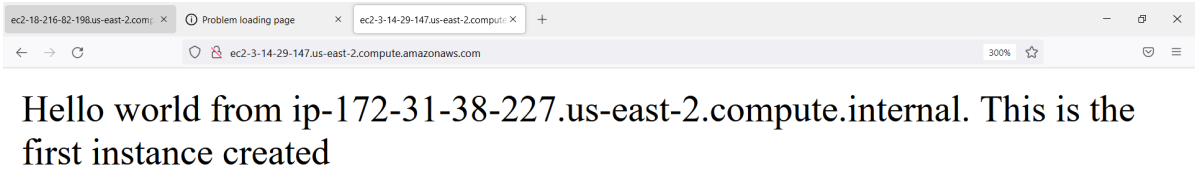
Launch a new instance from the launch option in the AMI dashboard. Select the appropriate security group (SEC_D7001D_FATEMA94) to allow SSH and HTTP connections. This is renamed as AMI_EC2_D7001D_FATEMA94 for ease of understandability.





2.10.1. QUESTION 12: What was the server's response?

Whatever was displayed when the public DNS of the instance EC2_D7001D_FATEMA94 was copied into the browser will be displayed.

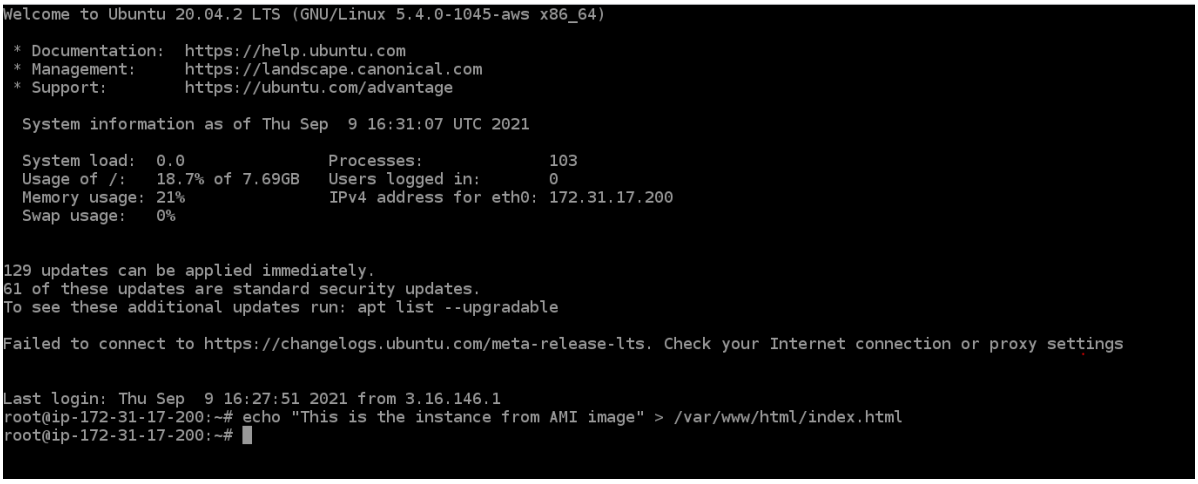


2.10.2. QUESTION 13: Explain why.

This is because when an Image is created, it copies or duplicates the configuration of that instance as well as the state of the EBS volumes attached (in this case, the instance EC2_D7001D_FATEMA94). Therefore whatever was present in the instance EC2_D7001D_FATEMA94, it is also reflected in the newly launched AMI created instance AMI_EC2_D7001D_FATEMA94 (aws, 2019).

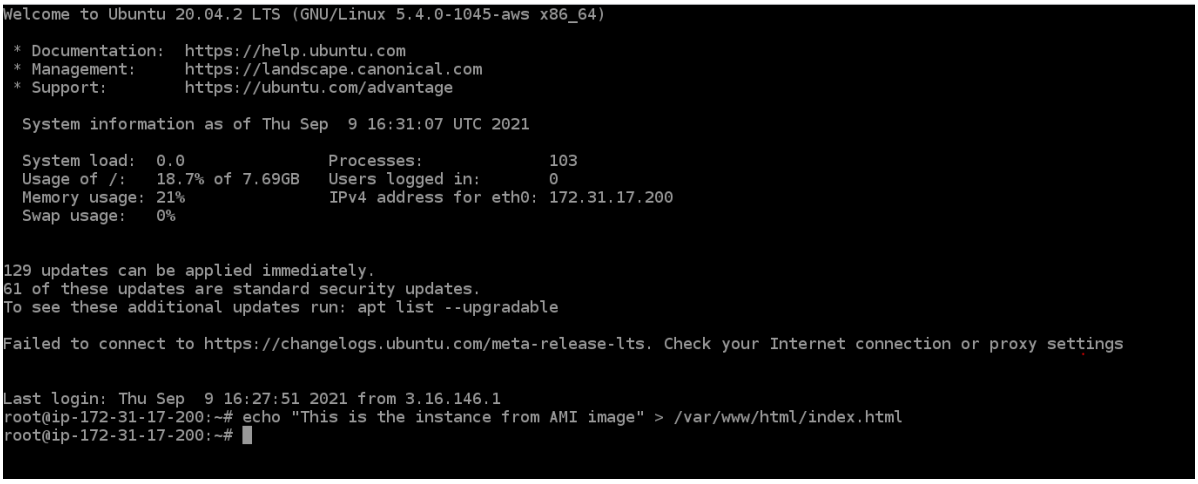
2.11. Now edit /var/www/html/index.html and enter text so that you can distinguish this instance from others.

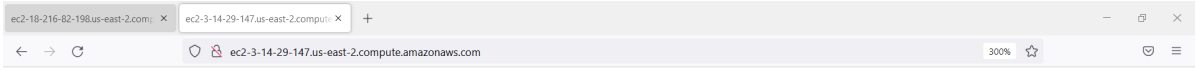
/var/www/html/index.html is edited to distinguish AMI_EC2_D7001D_FATEMA94 (instance created from AMI image) from EC2_D7001D_FATEMA94 (instance where webserver is running).



2.12. Check main webpages on both servers, where did the text changed? Why?

The text changed on the AMI_EC2_D7001D_FATEMA94 (instance created from AMI image). This is because we explicitly modified the index.html file from /var/www/html on this particular instance (AMI_EC2_D7001D_FATEMA94).

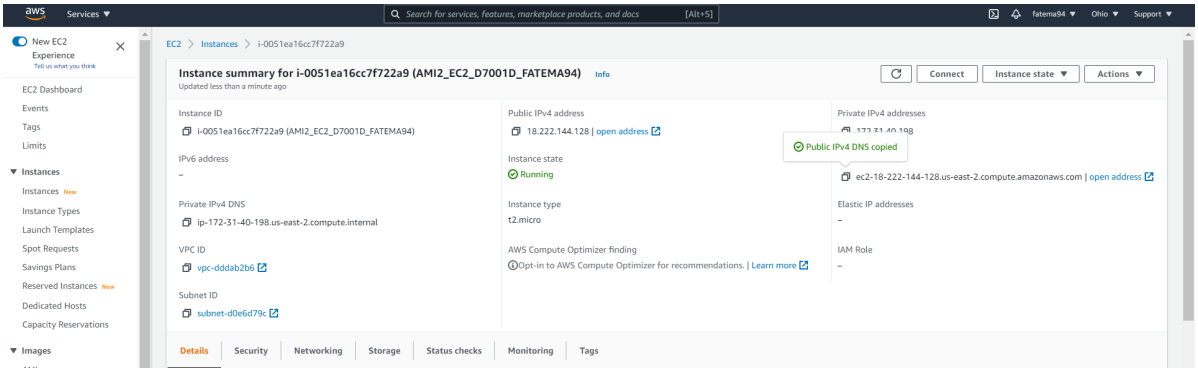




This is the instance from AMI image

2.13. What will be displayed if you launch new instance from your AMI?

If a new instance is launched from the the AMI, Whatever was displayed when the public DNS of the instance EC2_D7001D_FATEMA94 was copied into the browser will be displayed. The newly launched instance from AMI is named as AMI2_EC2_D7001D_FATEMA94. This is because this has been duplicated from the configurations of the instance EC2_D7001D_FATEMA94 and has not been explicitly changed.



Hello world from ip-172-31-38-227.us-east-2.compute.internal. This is the first instance created

3

Troubleshooting Remote Server

3.1. Here is a simple scenario. Suppose you have created a server program to run on TCP port 4032. The remote machine on which you install it is accessible through name myserv.mydomain.net. Next you install your server program and try to access it from a different computer. Your server is not responding!

3.1.1. QUESTION 14: Describe your step-by-step problem searching and troubleshooting approach.

1. Check the error logs to see if the exact error can be identified.

2. Check whether the web server is installed.

3. Check whether the web server is is running.

   (a) Use ***ping myserv.mydomain.net*** and ***tracert myserv.mydomain.net*** to check if the connection has been established.

   (b) Run the following command in CLI ***sudo netstat -plunt | grep <server>*** where the server the server in the question and check whether TCP is listening at port 4032.

   (c) Capture packets with a packet sniffer such as wireshark to analyze the packets to pinpoint the problem on a packet level.

4. Check whether the syntax of the web server configuration files correct.

5. Check whether the ports configured are open (not blocked by a firewall). Use netcat to check whether the TCP port 4032 is configured open.

6. Check whether the DNS settings are directing to the correct place - that is at myserv.mydomain.net.

7. Check whether the web server displaying the correct files and is pointing to the correct directory.

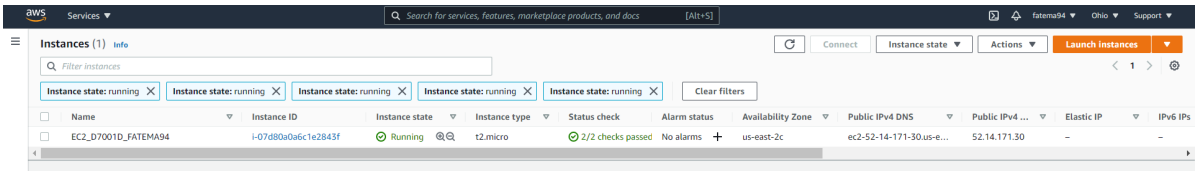
8. Check whether the permissions of the file and directory structures are correct.

9. Check whether it runs on other browsers as it could be a problem of TLS/SSL certificate issues.

The detailed explanation can be found at (Ellingwood, 2014).

3.1.2. QUESTION 15: Demonstrate your approach using appropriate operating system's commands and tools when troubleshooting communications between your local computer and running AWS instance configured with the web server.

1. Check and ensure that the instance is running and completing both status the checks.



2. Check and ensure that the instance has booted correctly - that there is no failed the instance status check due to operating system issues or no "Kernel panic" error.

3. Check and ensure that the instance is attached to the correct security group - that is SSH and HTTP is allowed for **ANYWHERE** IP addresses.

   From the EC2 Dashboard, select Security and then the appropriate security group. Ensure that TCP is allowed in the inbound rule for TCP port 4032. Further ensure that this security group is also attached to the correct instance.

4. Check the error logs to identify the error from /var/log/apache2/error.log.

```
Last login: Thu Sep  9 14:17:35 2021 from 192.165.134.226
ubuntu@ip-172-31-38-227:~$ cd /var/log
ubuntu@ip-172-31-38-227:/var/log$ ls
amazon   apt      btmp                cloud-init.log  dmesg    dmesg.1.gz  journal   landscape  private  unattended-upgrades
apache2  auth.log  cloud-init-output.log  dist-upgrade    dmesg.0  dpkg.log    kern.log  lastlog    syslog   wtmp
ubuntu@ip-172-31-38-227:/var/log$ cd apache2
ubuntu@ip-172-31-38-227:/var/log/apache2$ ls
access.log  error.log  other_vhosts_access.log
ubuntu@ip-172-31-38-227:/var/log/apache2$
```

```
[Thu Sep 09 14:45:14.376617 2021] [mpm_event:notice] [pid 2974:tid 140008089099328] AH00489: Apache/2.4.41 (Ubuntu) configured -- resuming normal operations
[Thu Sep 09 14:45:14.380401 2021] [core:notice] [pid 2974:tid 140008089099328] AH00094: Command line: '/usr/sbin/apache2'
[Thu Sep 09 16:00:41.774404 2021] [mpm_event:notice] [pid 2974:tid 140008089099328] AH00491: caught SIGTERM, shutting down
[Thu Sep 09 16:01:38.740340 2021] [mpm_event:notice] [pid 507:tid 140567266794560] AH00489: Apache/2.4.41 (Ubuntu) configured -- resuming normal operations
[Thu Sep 09 16:01:38.758882 2021] [core:notice] [pid 507:tid 140567266794560] AH00094: Command line: '/usr/sbin/apache2'
[Thu Sep 09 16:49:09.271187 2021] [mpm_event:notice] [pid 507:tid 140567266794560] AH00491: caught SIGTERM, shutting down
[Thu Sep 09 22:23:36.680553 2021] [mpm_event:notice] [pid 517:tid 139885415205952] AH00489: Apache/2.4.41 (Ubuntu) configured -- resuming normal operations
[Thu Sep 09 22:23:36.685459 2021] [core:notice] [pid 517:tid 139885415205952] AH00094: Command line: '/usr/sbin/apache2'
```

5. Check whether the web server (apache2) is correctly installed installed. Also check whether the TCPservice is running properly (not shown since the package is not installed).

```
ubuntu@ip-172-31-38-227:/var/log/apache2$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2021-09-09 22:23:36 UTC; 9min ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 407 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 517 (apache2)
      Tasks: 55 (limit: 1160)
     Memory: 7.8M
     CGroup: /system.slice/apache2.service
             ├─517 /usr/sbin/apache2 -k start
             ├─521 /usr/sbin/apache2 -k start
             └─522 /usr/sbin/apache2 -k start

Sep 09 22:23:35 ip-172-31-38-227 systemd[1]: Starting The Apache HTTP Server...
Sep 09 22:23:36 ip-172-31-38-227 systemd[1]: Started The Apache HTTP Server.
ubuntu@ip-172-31-38-227:/var/log/apache2$
```

6. Check whether the web server is running.
   (a) Use ping to determine the apache2 server can be reached. For ping to work, ICMP has to enabled for the server address on both inbound and outbound rules.
   (b) Run the following command in CLI ***sudo netstat -plunt | grep apache2*** to see if it configured to the right port.
7. Check whether the syntax of the web server configuration files correct.
8. Check whether the ports configured are open (not blocked by a firewall). Run the following command in CLI ***nc -zv <server_name/IP> <port_number>*** to check whether it is open and reachable.
9. Check whether the apache server displaying the correct files and is pointing to the correct directory(/var/www/html/index.html).
10. Check whether the permissions of the file and directory structures are correct.
11. Check whether it runs on other browsers as it could be a problem of TLS/SSL certificate issues.

3.2. Set up wireshark (tshark) on one of the instances and locally on your computer (http://shieldroute.blogspot.se/2012/08/wireshark-on-aws-ec2.html).Start monitoring traffic.

Use the following commands on the instance to create the packet capture file:

1. sudo fdisk -l
2. sudo mkfs -t ext4 /dev/xvdf
3. sudo mkdir /home/data-storage
4. sudo mount /dev/xvdf /home/data-storage
5. sudo mkdir /home/data-storage/wireshark
6. sudo chown root:ubuntu /home/data-storage/wireshark
7. sudo chmod -R 774 /home/data-storage/wireshark
8. sudo apt-get install wireshark tshark
9. sudo su
10. cd /home/data-storage/wireshark
11. tshark -i eth0 -a duration:10 -w my.pcap
12. chmod u+rwx,g+rwx,o+rwx my.pcap

Move the pcap file from the instance to the local machine using the following command on the command line where the key is situated: scp -i KEy_D7001D_FATEMA94.pem ubuntu@ec2-52-14-171-30.us-east-2.compute.amazonaws.com:/home/data-storage/wireshark/my.pcap .

The detailed explanation can be found at (shieldroute, 2014) . Images have been captured to document the entire wireshark and tshark execution process.

```
Last login: Thu Sep  9 22:27:10 2021 from 3.16.146.2
ubuntu@ip-172-31-38-227:~$ sudo fdisk -l
Disk /dev/loop0: 33.35 MiB, 34959360 bytes, 68280 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
```

```
Disk /dev/xvdf: 6 GiB, 6442450944 bytes, 12582912 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
ubuntu@ip-172-31-38-227:~$ sudo mkfs -t ext4 /dev/xvdf
mke2fs 1.45.5 (07-Jan-2020)
Creating filesystem with 1572864 4k blocks and 393216 inodes
```

```
ubuntu@ip-172-31-38-227:~$ sudo mkdir /home/data-storage
ubuntu@ip-172-31-38-227:~$ sudo mount /dev/DEVICE /home/data-storage
mount: /home/data-storage: special device /dev/DEVICE does not exist.
ubuntu@ip-172-31-38-227:~$ sudo mount /dev/xvdf /home/data-storage
ubuntu@ip-172-31-38-227:~$ sudo mkdir /home/data-storage/wireshark
ubuntu@ip-172-31-38-227:~$
ubuntu@ip-172-31-38-227:~$
ubuntu@ip-172-31-38-227:~$
ubuntu@ip-172-31-38-227:~$ sudo chown root:ubuntu /home/data-storage/wireshark
ubuntu@ip-172-31-38-227:~$ sudo chmid -R 774 wireshark
sudo: chmid: command not found
ubuntu@ip-172-31-38-227:~$ sudo chmod -R 774 wireshark
chmod: cannot access 'wireshark': No such file or directory
ubuntu@ip-172-31-38-227:~$
ubuntu@ip-172-31-38-227:~$ sudo chmod -R 774 wireshark
chmod: cannot access 'wireshark': No such file or directory
ubuntu@ip-172-31-38-227:~$ cd /home/data-storage
ubuntu@ip-172-31-38-227:/home/data-storage$ ls
lost+found  wireshark
ubuntu@ip-172-31-38-227:/home/data-storage$ cd wireshark/
ubuntu@ip-172-31-38-227:/home/data-storage/wireshark$ ls
ubuntu@ip-172-31-38-227:/home/data-storage/wireshark$ sudo chmod -R 774 wireshar
k
chmod: cannot access 'wireshark': No such file or directory
ubuntu@ip-172-31-38-227:/home/data-storage/wireshark$ ^C
ubuntu@ip-172-31-38-227:/home/data-storage/wireshark$ cd ..
ubuntu@ip-172-31-38-227:/home/data-storage$ cd ..
ubuntu@ip-172-31-38-227:/home$ cd ..
ubuntu@ip-172-31-38-227:/$ sudo su
root@ip-172-31-38-227:/# sudo chown root:ubuntu /home/data-storage/wireshark
root@ip-172-31-38-227:/# sudo shmod -R 774 /home/data-storage/wireshark
sudo: shmod: command not found
root@ip-172-31-38-227:/# sudo chmod -R 774 /home/data-storage/wireshark
root@ip-172-31-38-227:/# sudo apt-get install wireshark tshark
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

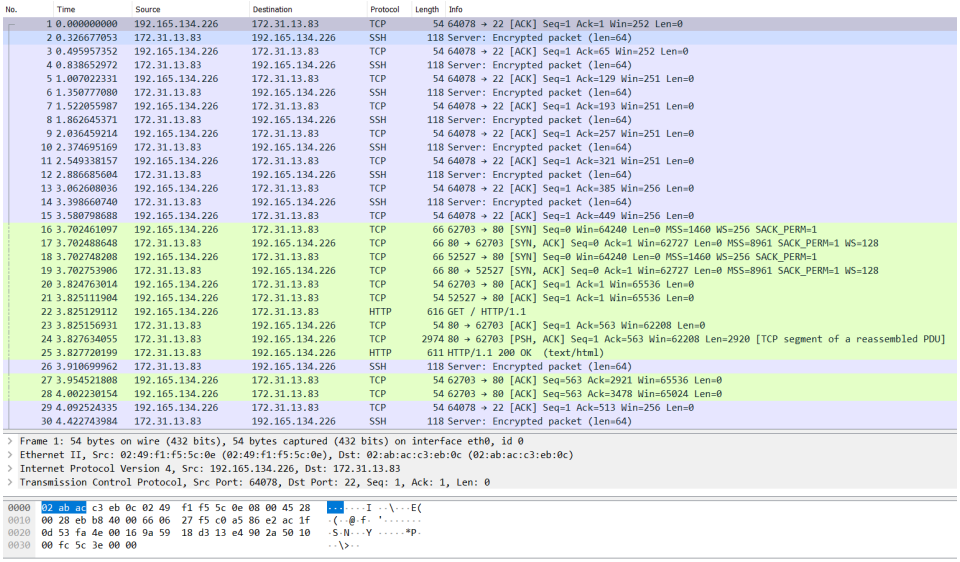
```
root@ip-172-31-38-227:/# cd /home/data-storage/wireshark
root@ip-172-31-38-227:/home/data-storage/wireshark# tshark -i eth0 -a duration:1
0 -w my.pcap
Running as user "root" and group "root". This could be dangerous.
Capturing on 'eth0'
45
root@ip-172-31-38-227:/home/data-storage/wireshark# root@ip-172-31-38-227:/home/
data-storage/wireshark#
bash: root@ip-172-31-38-227:/home/data-storage/wireshark#: No such file or direc
tory
root@ip-172-31-38-227:/home/data-storage/wireshark# root@ip-172-31-38-227:/home/
data-storage/wireshark#
bash: root@ip-172-31-38-227:/home/data-storage/wireshark#: No such file or direc
tory
root@ip-172-31-38-227:/home/data-storage/wireshark# root@ip-172-31-38-227:/home/
data-storage/wireshark#
bash: root@ip-172-31-38-227:/home/data-storage/wireshark#: No such file or direc
tory
root@ip-172-31-38-227:/home/data-storage/wireshark# chmod u+rwx,g+rwx,o+rwx my.p
cap
root@ip-172-31-38-227:/home/data-storage/wireshark#
```

Command Prompt

```
Microsoft Windows [Version 10.0.19042.1165]
(c) Microsoft Corporation. All rights reserved.

C:\Users\GL62M>cd Desktop

C:\Users\GL62M\Desktop>cd "deep learning course"

C:\Users\GL62M\Desktop\deep learning course>scp -i KEY_D7001D_FATEMA94.pem   ubuntu@ec2-52-14-171-30.us-east-2.compute.amazonaws.com
usage: scp [-346BCpqrTv] [-c cipher] [-F ssh_config] [-i identity_file]
           [-J destination] [-l limit] [-o ssh_option] [-P port]
           [-S program] source ... target

C:\Users\GL62M\Desktop\deep learning course>scp -i KEY_D7001D_FATEMA94.pem   ubuntu@ec2-52-14-171-30.us-east-2.compute.amazonaws.com:/home/data-storage/wireshark/my.pcap  .
The authenticity of host 'ec2-52-14-171-30.us-east-2.compute.amazonaws.com (52.14.171.30)' can't be established.
ECDSA key fingerprint is SHA256:JmTpWRReQHcPkKr8CRqQWFwJTVp4Bsk9bfEcp8wjHCs.
Are you sure you want to continue connecting (yes/no/[fingerprint])?
Warning: Permanently added 'ec2-52-14-171-30.us-east-2.compute.amazonaws.com,52.14.171.30' (ECDSA) to the list of known hosts.
scp: /home/data-storage/wireshark/my.pcap: Permission denied

C:\Users\GL62M\Desktop\deep learning course>scp -i KEY_D7001D_FATEMA94.pem   ubuntu@ec2-52-14-171-30.us-east-2.compute.amazonaws.com:/home/data-storage/wireshark/my.pcap  .
my.pcap                                                                                   100% 5648    45.1KB/s   00:00

C:\Users\GL62M\Desktop\deep learning course>
```

3.2.1. QUESTION 16: Be able to interpret and explain the information about different protocols, their fields etc.

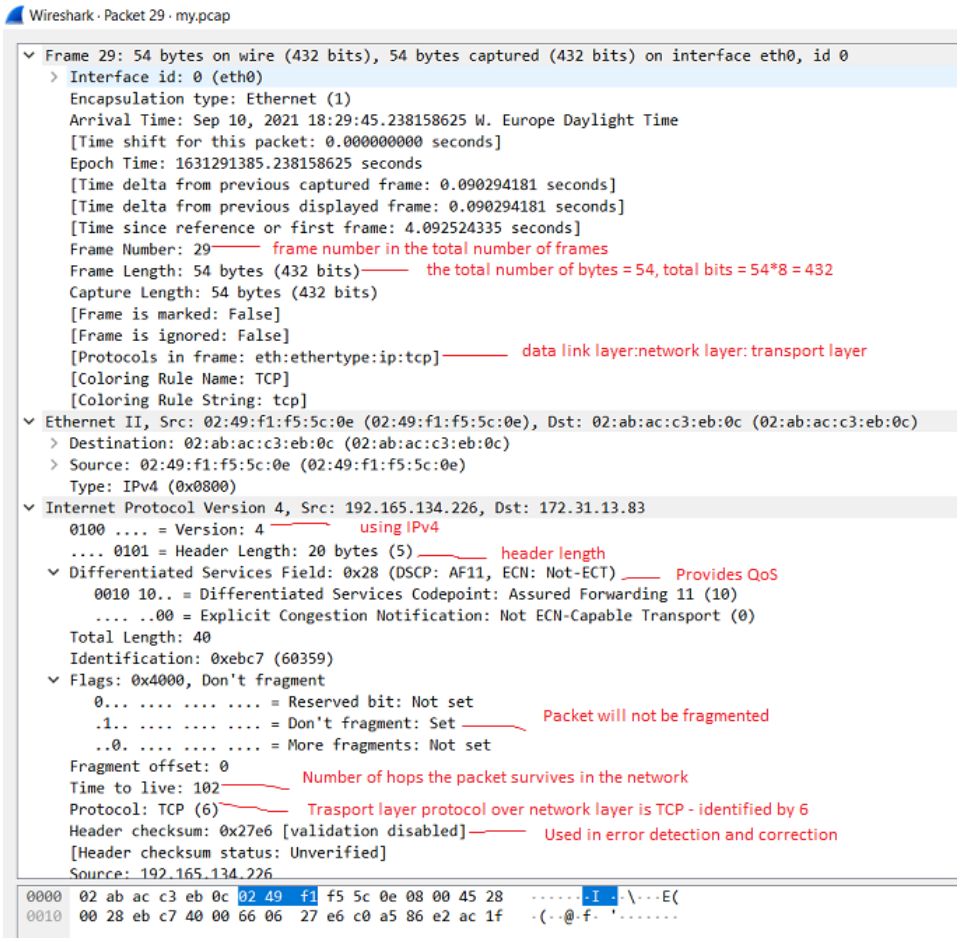
Load the my.pcap file in wireshark.



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.165.134.226	172.31.13.83	TCP	54	64078 → 22 [ACK] Seq=1 Ack=1 Win=252 Len=0
2	0.326677053	172.31.13.83	192.165.134.226	SSH	118	Server: Encrypted packet (len=64)
3	0.495957352	192.165.134.226	172.31.13.83	TCP	54	64078 → 22 [ACK] Seq=1 Ack=65 Win=252 Len=0
4	0.838652972	172.31.13.83	192.165.134.226	SSH	118	Server: Encrypted packet (len=64)
5	1.007022331	192.165.134.226	172.31.13.83	TCP	54	64078 → 22 [ACK] Seq=1 Ack=129 Win=251 Len=0
6	1.350777080	172.31.13.83	192.165.134.226	SSH	118	Server: Encrypted packet (len=64)
7	1.522055987	192.165.134.226	172.31.13.83	TCP	54	64078 → 22 [ACK] Seq=1 Ack=193 Win=251 Len=0
8	1.862645371	172.31.13.83	192.165.134.226	SSH	118	Server: Encrypted packet (len=64)
9	2.036459214	192.165.134.226	172.31.13.83	TCP	54	64078 → 22 [ACK] Seq=1 Ack=257 Win=251 Len=0
10	2.374695169	172.31.13.83	192.165.134.226	SSH	118	Server: Encrypted packet (len=64)
11	2.549338157	192.165.134.226	172.31.13.83	TCP	54	64078 → 22 [ACK] Seq=1 Ack=321 Win=251 Len=0
12	2.886685604	172.31.13.83	192.165.134.226	SSH	118	Server: Encrypted packet (len=64)
13	3.062608036	192.165.134.226	172.31.13.83	TCP	54	64078 → 22 [ACK] Seq=1 Ack=385 Win=256 Len=0
14	3.398660740	172.31.13.83	192.165.134.226	SSH	118	Server: Encrypted packet (len=64)
15	3.580798688	192.165.134.226	172.31.13.83	TCP	54	64078 → 22 [ACK] Seq=1 Ack=449 Win=256 Len=0
16	3.702461097	192.165.134.226	172.31.13.83	TCP	66	62703 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
17	3.702488648	172.31.13.83	192.165.134.226	TCP	66	80 → 62703 [SYN, ACK] Seq=0 Ack=1 Win=62727 Len=0 MSS=8961 SACK_PERM=1 WS=128
18	3.702748208	192.165.134.226	172.31.13.83	TCP	66	52527 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
19	3.702753906	172.31.13.83	192.165.134.226	TCP	66	80 → 52527 [SYN, ACK] Seq=0 Ack=1 Win=62727 Len=0 MSS=8961 SACK_PERM=1 WS=128
20	3.824763014	192.165.134.226	172.31.13.83	TCP	54	62703 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
21	3.825111904	192.165.134.226	172.31.13.83	TCP	54	52527 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
22	3.825129112	192.165.134.226	172.31.13.83	HTTP	616	GET / HTTP/1.1
23	3.825156931	172.31.13.83	192.165.134.226	TCP	54	80 → 62703 [ACK] Seq=1 Ack=563 Win=62208 Len=0
24	3.827634055	172.31.13.83	192.165.134.226	TCP	2974	80 → 62703 [PSH, ACK] Seq=1 Ack=563 Win=62208 Len=2920 [TCP segment of a reassembled PDU]
25	3.827720199	172.31.13.83	192.165.134.226	HTTP	611	HTTP/1.1 200 OK (text/html)
26	3.910699962	172.31.13.83	192.165.134.226	SSH	118	Server: Encrypted packet (len=64)
27	3.954521808	192.165.134.226	172.31.13.83	TCP	54	62703 → 80 [ACK] Seq=563 Ack=2921 Win=65536 Len=0
28	4.002230154	192.165.134.226	172.31.13.83	TCP	54	62703 → 80 [ACK] Seq=563 Ack=3478 Win=65024 Len=0
29	4.092524335	192.165.134.226	172.31.13.83	TCP	54	64078 → 22 [ACK] Seq=1 Ack=513 Win=256 Len=0
30	4.422743984	172.31.13.83	192.165.134.226	SSH	118	Server: Encrypted packet (len=64)

```
> Frame 1: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface eth0, id 0
> Ethernet II, Src: 02:49:f1:f5:5c:0e (02:49:f1:f5:5c:0e), Dst: 02:ab:ac:c3:eb:0c (02:ab:ac:c3:eb:0c)
> Internet Protocol Version 4, Src: 192.165.134.226, Dst: 172.31.13.83
> Transmission Control Protocol, Src Port: 64078, Dst Port: 22, Seq: 1, Ack: 1, Len: 0

0000  02 ab ac c3 eb 0c 02 49  f1 f5 5c 0e 08 00 45 28   ·····I ··\···E(
0010  00 28 eb b8 40 00 66 06  27 f5 c0 a5 86 e2 ac 1f   ·(··@·f· '·····
0020  0d 53 fa 4e 00 16 9a 59  18 d3 13 e4 90 2a 50 10   ·S·N···Y ·····*P·
0030  00 fc 5c 3e 00 00                                  ··\>··
```

The TCP packets (transport layer) are explained in the images:



```
Wireshark · Packet 29 · my.pcap
Frame 29: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface eth0, id 0
  Interface id: 0 (eth0)
  Encapsulation type: Ethernet (1)
  Arrival Time: Sep 10, 2021 18:29:45.238158625 W. Europe Daylight Time
  [Time shift for this packet: 0.000000000 seconds]
  Epoch Time: 1631291385.238158625 seconds
  [Time delta from previous captured frame: 0.090294181 seconds]
  [Time delta from previous displayed frame: 0.090294181 seconds]
  [Time since reference or first frame: 4.092524335 seconds]
  Frame Number: 29 ——— frame number in the total number of frames
  Frame Length: 54 bytes (432 bits) ——— the total number of bytes = 54, total bits = 54*8 = 432
  Capture Length: 54 bytes (432 bits)
  [Frame is marked: False]
  [Frame is ignored: False]
  [Protocols in frame: eth:ethertype:ip:tcp] ——— data link layer:network layer: transport layer
  [Coloring Rule Name: TCP]
  [Coloring Rule String: tcp]
Ethernet II, Src: 02:49:f1:f5:5c:0e (02:49:f1:f5:5c:0e), Dst: 02:ab:ac:c3:eb:0c (02:ab:ac:c3:eb:0c)
  Destination: 02:ab:ac:c3:eb:0c (02:ab:ac:c3:eb:0c)
  Source: 02:49:f1:f5:5c:0e (02:49:f1:f5:5c:0e)
  Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.165.134.226, Dst: 172.31.13.83
  0100 .... = Version: 4 ——— using IPv4
  .... 0101 = Header Length: 20 bytes (5) ——— header length
  Differentiated Services Field: 0x28 (DSCP: AF11, ECN: Not-ECT) ——— Provides QoS
    0010 10.. = Differentiated Services Codepoint: Assured Forwarding 11 (10)
    .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
  Total Length: 40
  Identification: 0xebc7 (60359)
  Flags: 0x4000, Don't fragment
    0... .... .... .... = Reserved bit: Not set
    .1.. .... .... .... = Don't fragment: Set ——— Packet will not be fragmented
    ..0. .... .... .... = More fragments: Not set
  Fragment offset: 0
  Time to live: 102 ——— Number of hops the packet survives in the network
  Protocol: TCP (6) ——— Trasport layer protocol over network layer is TCP - identified by 6
  Header checksum: 0x27e6 [validation disabled] ——— Used in error detection and correction
  [Header checksum status: Unverified]
  Source: 192.165.134.226

0000  02 ab ac c3 eb 0c 02 49  f1 f5 5c 0e 08 00 45 28   ·····I ··\···E(
0010  00 28 eb c7 40 00 66 06  27 e6 c0 a5 86 e2 ac 1f   ·(··@·f· '·····
```

Wireshark · Packet 29 · my.pcap

```
    Protocol: TCP (6)
    Header checksum: 0x27e6 [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.165.134.226
    Destination: 172.31.13.83
Transmission Control Protocol, Src Port: 64078, Dst Port: 22, Seq: 1, Ack: 513, Len: 0
    Source Port: 64078
    Destination Port: 22
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 1    (relative sequence number)
    Sequence number (raw): 2589530323
    [Next sequence number: 1    (relative sequence number)]
    Acknowledgment number: 513    (relative ack number)
    Acknowledgment number (raw): 333746730
    0101 .... = Header Length: 20 bytes (5)
  Flags: 0x010 (ACK)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): Not set
      .... .0.. .... = ECN-Echo: Not set
      .... ..0. .... = Urgent: Not set
      .... ...1 .... = Acknowledgment: Set
      .... .... 0... = Push: Not set
      .... .... .0.. = Reset: Not set
      .... .... ..0. = Syn: Not set
      .... .... ...0 = Fin: Not set
      [TCP Flags: ·····A····]
    Window size value: 256
    [Calculated window size: 256]
    [Window size scaling factor: -1 (unknown)]
    Checksum: 0x5a3a [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
  [SEQ/ACK analysis]
      [This is an ACK to the segment in frame: 26]
      [The RTT to ACK the segment was: 0.181824373 seconds]
  [Timestamps]
```

```
0000  02 ab ac c3 eb 0c 02 49  f1 f5 5c 0e 08 00 45 28   ·····I··\···E(
0010  00 28 eb c7 40 00 66 06  27 e6 c0 a5 86 e2 ac 1f   ·(··@·f· '·····
```

source IP addr

dest ip addr

source trasport port

destination transport port

All values of the flags are unset except for acknowledgement, which means this is an ack frame

used to ensure flow control

Since this is an ack frame, this shows that this is the ack for frame 26 and the rount trip time for the ack is also mentioned

Since the headers have been already explained in detail for TCP, for SSH (application layer) the important thing to note is the direction of the communication.

Wireshark · Packet 30 · my.pcap

```
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x10 (DSCP: Unknown, ECN: Not-ECT)
    Total Length: 104
    Identification: 0x4db8 (19896)
  > Flags: 0x4000, Don't fragment
    Fragment offset: 0
    Time to live: 64
    Protocol: TCP (6)
    Header checksum: 0xebcd [validation disabled]
    [Header checksum status: Unverified]
    Source: 172.31.13.83
    Destination: 192.165.134.226
v Transmission Control Protocol, Src Port: 22, Dst Port: 64078, Seq: 513, Ack: 1, Len: 64
    Source Port: 22
    Destination Port: 64078
    [Stream index: 0]
    [TCP Segment Len: 64]
    Sequence number: 513    (relative sequence number)
    Sequence number (raw): 333746730
    [Next sequence number: 577    (relative sequence number)]
    Acknowledgment number: 1    (relative ack number)
    Acknowledgment number (raw): 2589530323
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x018 (PSH, ACK)
    Window size value: 463
    [Calculated window size: 463]
    [Window size scaling factor: -1 (unknown)]
    Checksum: 0x0155 [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
  > [SEQ/ACK analysis]
  > [Timestamps]
    TCP payload (64 bytes)
v SSH Protocol
    Packet Length (encrypted): d049a50e
    Encrypted Packet: 67f415b192b7abd0d16abb8b47729c20460e3a136b853c22…
    [Direction: server-to-client]

0000  02 49 f1 f5 5c 0e 02 ab  ac c3 eb 0c 08 00 45 10   ·I··\··· ------E·
0010  00 68 4d b8 40 00 40 06  eb cd ac 1f 0d 53 c0 a5   ·hM·@·@· -----S··
0020  86 e2 00 16 fa 4e 13 e4  92 2a 9a 59 18 d3 50 18   -----N·· ·*·Y··P·
```

Since the headers have been already explained in detail for TCP, for HTTP (application layer) only the relevant fields will be detailed. First is the HTTP GET request.

Wireshark · Packet 22 · my.pcap

```
> Frame 22: 616 bytes on wire (4928 bits), 616 bytes captured (4928 bits) on interface eth0, id 0
> Ethernet II, Src: 02:49:f1:f5:5c:0e (02:49:f1:f5:5c:0e), Dst: 02:ab:ac:c3:eb:0c (02:ab:ac:c3:eb:0c)
> Internet Protocol Version 4, Src: 192.165.134.226, Dst: 172.31.13.83
> Transmission Control Protocol, Src Port: 62703, Dst Port: 80, Seq: 1, Ack: 1, Len: 562
v Hypertext Transfer Protocol
  > GET / HTTP/1.1\r\n                                  HTTP GET req version
    Host: 18.118.146.214\r\n                            Host IP of server
    Connection: keep-alive\r\n                          single TCP connection to remain open for multiple HTTP requests/responses
    Cache-Control: max-age=0\r\n
    Upgrade-Insecure-Requests: 1\r\n                    Allowed browsers
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.63 Safari/537.36\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9\r\n
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: en-GB,en-US;q=0.9,en;q=0.8\r\n     Allowed languages
    If-None-Match: "2aa6-5cba67a05a425-gzip"\r\n
    If-Modified-Since: Fri, 10 Sep 2021 16:19:15 GMT\r\n   server send response only if modified
    \r\n
    [Full request URI: http://18.118.146.214/]
    [HTTP request 1/1]
    [Response in frame: 25]
```

Next is the response for the particular GET request.

Wireshark · Packet 25 · my.pcap

```
> Frame 25: 611 bytes on wire (4888 bits), 611 bytes captured (4888 bits) on interface eth0, id 0
> Ethernet II, Src: 02:ab:ac:c3:eb:0c (02:ab:ac:c3:eb:0c), Dst: 02:49:f1:f5:5c:0e (02:49:f1:f5:5c:0e)
> Internet Protocol Version 4, Src: 172.31.13.83, Dst: 192.165.134.226
> Transmission Control Protocol, Src Port: 80, Dst Port: 62703, Seq: 2921, Ack: 563, Len: 557
> [2 Reassembled TCP Segments (3477 bytes): #24(2920), #25(557)]
v Hypertext Transfer Protocol
  > HTTP/1.1 200 OK\r\n                    HTTP Successful Response status identified with 200 OK response
    Date: Fri, 10 Sep 2021 16:29:44 GMT\r\n    Date and time of response
    Server: Apache/2.4.41 (Ubuntu)\r\n    Origin server used to handle the response
    Last-Modified: Fri, 10 Sep 2021 16:19:15 GMT\r\n    Last modified date and time
    ETag: "2aa6-5cba67a05a425-gzip"\r\n
    Accept-Ranges: bytes\r\n
    Vary: Accept-Encoding\r\n
    Content-Encoding: gzip\r\n
  > Content-Length: 3138\r\n
    Keep-Alive: timeout=5, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html\r\n    the data type of the response is in html
    \r\n
    [HTTP response 1/1]
    [Time since request: 0.002591087 seconds]
    [Request in frame: 22]
    [Request URI: http://18.118.146.214/]
    Content-encoded entity body (gzip): 3138 bytes -> 10918 bytes
    File Data: 10918 bytes
v Line-based text data: text/html (375 lines)    The whole html response is also returned with the response
    \n
    <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">\n
    <html xmlns="http://www.w3.org/1999/xhtml">\n
      <!--\n
```

```
0000  02 49 f1 f5 5c 0e 02 ab  ac c3 eb 0c 08 00 45 00   ·I··\··· ######E·
0010  02 55 04 27 40 00 40 06  33 82 ac 1f 0d 53 c0 a5   ·U·'@·@· 3····S··
0020  86 e2 00 50 f4 ef 89 e9  00 ac 0b 8f b5 50 50 18   ···P···· #####PP·
0030  01 e6 03 42 00 00 f9 b6  ec 4b bb c2 52 5f 08 15   ···B···· ·K··R_··
0040  b8 92 b2 6d b8 e4 5e 79  9f 40 3e c0 2a fd 28 c6   ···m··^y ·@>·*·(·
0050  07 1b c2 f6 f2 fc f5 36  b7 4d 3f af b3 63 09 91   ·******6 ·M?··c··
0060  81 16 a6 58 61 fb 5e 03  0c ac 64 b6 58 16 85 d7   ···Xa·^· ··d·X···
0070  4c 41 f5 6d db a0 66 cf  c5 56 17 50 53 4c 42 78   LA·m··f· ·V·PSLBx
0080  77 35 07 45 ec 76 e3 05  09 8e 9d 0f 47 c0 57 cb   w5·E·v·· ····G·W·
0090  c3 90 3d b9 bf 68 6f fc  cf 89 f0 2d 0e 9c 77 d8   ··=··ho· ###··w·
00a0  1f 15 81 35 75 78 12 bc  08 01 64 f1 ef 0e 87 19   ···5ux·· ··d·····
```

Bibliography

aws. (2019). *How do i create an ami that is based on my ebs-backed ec2 instance?* Retrieved September 9, 2021, from https://aws.amazon.com/premiumsupport/knowledge-center/create-ami-ebs-backed/

aws. (2021). *Easy to use, high performance block storage at any scale.* Retrieved September 9, 2021, from https://aws.amazon.com/ebs/

Behera, S. (2021). *Migrate your amazon ebs volumes from gp2 to gp3 and save up to 20% on costs.* Retrieved September 9, 2021, from https://aws.amazon.com/blogs/storage/migrate-your-amazon-ebs-volumes-from-gp2-to-gp3-and-save-up-to-20-on-costs/

Checkpoint.com. (2021). *What are aws security groups?* Retrieved September 9, 2021, from https://www.checkpoint.com/cyber-hub/cloud-security/what-is-aws-security-groups/

Ellingwood, J. (2014). *How to troubleshoot common site issues on a linux server.* Retrieved September 9, 2021, from https://www.digitalocean.com/community/tutorials/how-to-troubleshoot-common-site-issues-on-a-linux-server

Phoronix. (2012). *Ubuntu, rhel, suse, amazon linux on the amazon ec2 cloud.* Retrieved September 9, 2021, from https://www.phoronix.com/scan.php?page=article&item=amazon_ec2_distros&num=4

shieldroute. (2014). *Wireshark on aws ec2.* Retrieved September 9, 2021, from http://shieldroute.blogspot.com/2012/08/wireshark-on-aws-ec2.html