Name: Darshan Raju Loni

Srn: PES2UG22CS158

Sec: C

Week #3

Understanding Persistent and Non-persistent HTTP Connections

To understand persistent and non-persistent HTTP connections and corresponding performance impact.

Create a web page with N (e.g. 10) embedded images. Each image should be of minimum 2 MB size. Configure your browser (Firefox) with following settings (each setting requires repeat of experiment)

- Non persistent connection
- 2 persistent connections
- 4 persistent connections
- 6 persistent connections
- 10 persistent connections.

Observation: Note down the time taken to display the entire page in each of the settings. Ensure that (cache is cleared before starting the web request). Explain the response time differences. What is the optimal number of persistent connections for best performance? Explain your answer.

Introduction

The Apache HTTP server is the most widely-used web server in the world. It provides many powerful features including dynamically loadable modules, robust media support, and extensive integration with other popular software.

Objective: Understand persistent and non-persistent HTTP connections and corresponding performance impact.

Experiment: Create a web page with N (e.g. 10) embedded images. Each image should be of minimum 2 MB size. Configure your browser (Firefox) with following settings (each setting requires repeat of experiment)

a) Non-persistent connection
b) 2 persistent connections
c) 4 persistent connections
d) 6 persistent connections
e) 10 persistent connections

Note down the time taken to display the entire page in each of the settings. Ensure that cache is cleared before starting the web request. Explain the response time differences. What is the optimal number of persistent connections for best performance? Explain your answer.

Note: To install Apache server, use the following command,

```
sudo apt-get install apache2
```

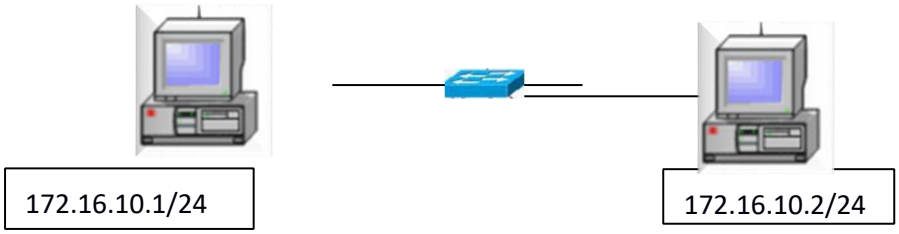
If there is any error during installation, update the package manager by issuing the command,

```
sudo apt-get update
```

EXECUTION STEPS

Step 1: Connect 2 desktops using switch and cables as shown below. (Use 2 VMs on Virtualbox or VMware instead of physical connections.)

Server **Client**



Server: OSBOXES.ORG VM **Client: SEED UBUNTU VM**

Server Side:

Step 2: Check your Web Server

At the end of the installation process, Ubuntu 16.04 starts Apache. The web server should already be up and running. We can check with the `systemctl` command to make sure the service is running by typing:

sudo systemctl status apache2

or

sudo service apache2 status

```
osboxes@osboxes:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enab>
     Active: active (running) since Thu 2024-02-15 10:43:31 EST; 22s ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 3189 (apache2)
      Tasks: 55 (limit: 4613)
     Memory: 5.3M
        CPU: 114ms
     CGroup: /system.slice/apache2.service
             ├─3189 /usr/sbin/apache2 -k start
             ├─3190 /usr/sbin/apache2 -k start
             └─3191 /usr/sbin/apache2 -k start

Feb 15 10:43:30 osboxes systemd[1]: Starting apache2.service - The Apache HTTP >
Feb 15 10:43:31 osboxes apachectl[3188]: AH00558: apache2: Could not reliably d>
Feb 15 10:43:31 osboxes systemd[1]: Started apache2.service - The Apache HTTP S>
```

As you can see above, the service appears to have started successfully. However, the best way to test this is to actually request a page from Apache. You can access the default Apache

landing page to confirm that the software is running properly. You can access this through your server's domain name or IP address.

Step 3: Server IP address can be set by the following command

$sudo ip addr add 172.16.10.1/24 dev enps0

$sudo ip addr

Note: If IP address fluctuates, kindly setup the IP address manually using 'Edit connections'.

```
osboxes@osboxes:~$ sudo ip addr add 172.16.10.1/24 dev enp0s3
osboxes@osboxes:~$ sudo ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen
    link/ether 08:00:27:ab:de:e1 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
       valid_lft 86071sec preferred_lft 86071sec
    inet 172.16.10.1/24 scope global enp0s3
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:feab:dee1/64 scope link
       valid_lft forever preferred_lft forever
```

Step 4: The **apache2.conf** file present in the **etc/apache2** directory is modified as:

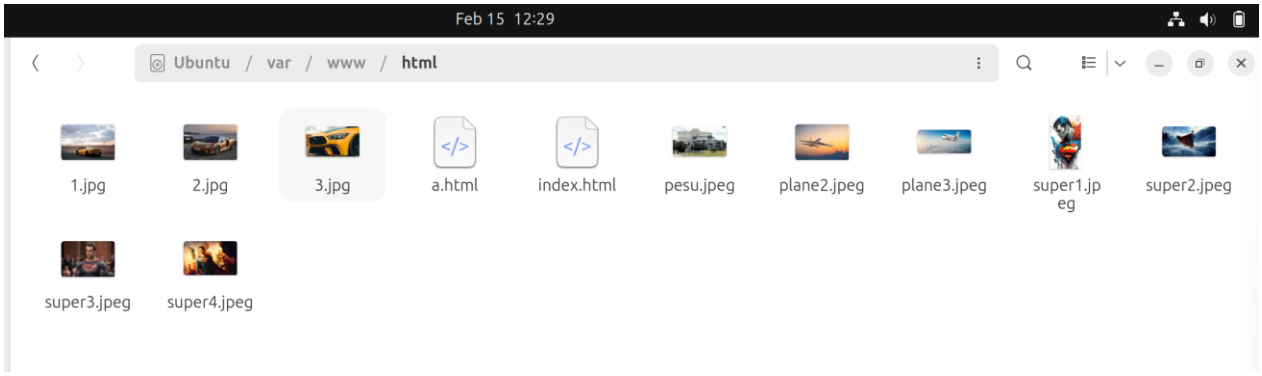
a) The **keep-alive** option was set (i.e. value was made **ON**)

b) The **MaximumKeepAliveRequests** were set to **2**

```
PidFile ${APACHE_PID_FILE}

#
# Timeout: The number of seconds before receives and sends time out.
#
Timeout 300

#
# KeepAlive: Whether or not to allow persistent connections (more than
# one request per connection). Set to "Off" to deactivate.
#
KeepAlive On

#
# MaxKeepAliveRequests: The maximum number of requests to allow
# during a persistent connection. Set to 0 to allow an unlimited amount.
# We recommend you leave this number high, for maximum performance.
#
MaxKeepAliveRequests 2

#
# KeepAliveTimeout: Number of seconds to wait for the next request from the
# same client on the same connection.
#
KeepAliveTimeout 5


# These need to be set in /etc/apache2/envvars
User ${APACHE_RUN_USER}
Group ${APACHE_RUN_GROUP}
```

Step 5: Store images in the server path. A html page consisting of 10 images having size > 2MB were placed and accessed by the client. This html page is stored in the location - **/var/www/html/file_name.html**.

Note: Use the images provided by faculty incharges.



Step 6: Prepare a web page as shown below. The html file needs to add 10 images. (Kindly skip the style attribute in the below image)

Open — a.html — /var/www/html

```
<!DOCTYPE html>
<html>
  <body style=background-color:#E1F6FF>
    <h2>random images</h2>
    <img src="1.jpg">
    <img src="2.jpg">
    <img src="3.jpg">
    <img src="pesu.jpeg">
    <img src="plane2.jpeg">
    <img src="plane3.jpeg">
    <img src="super1.jpeg">
    <img src="super2.jpeg">
    <img src="super3.jpeg">
    <img src="super4.jpeg">

  </body>
</html>
```

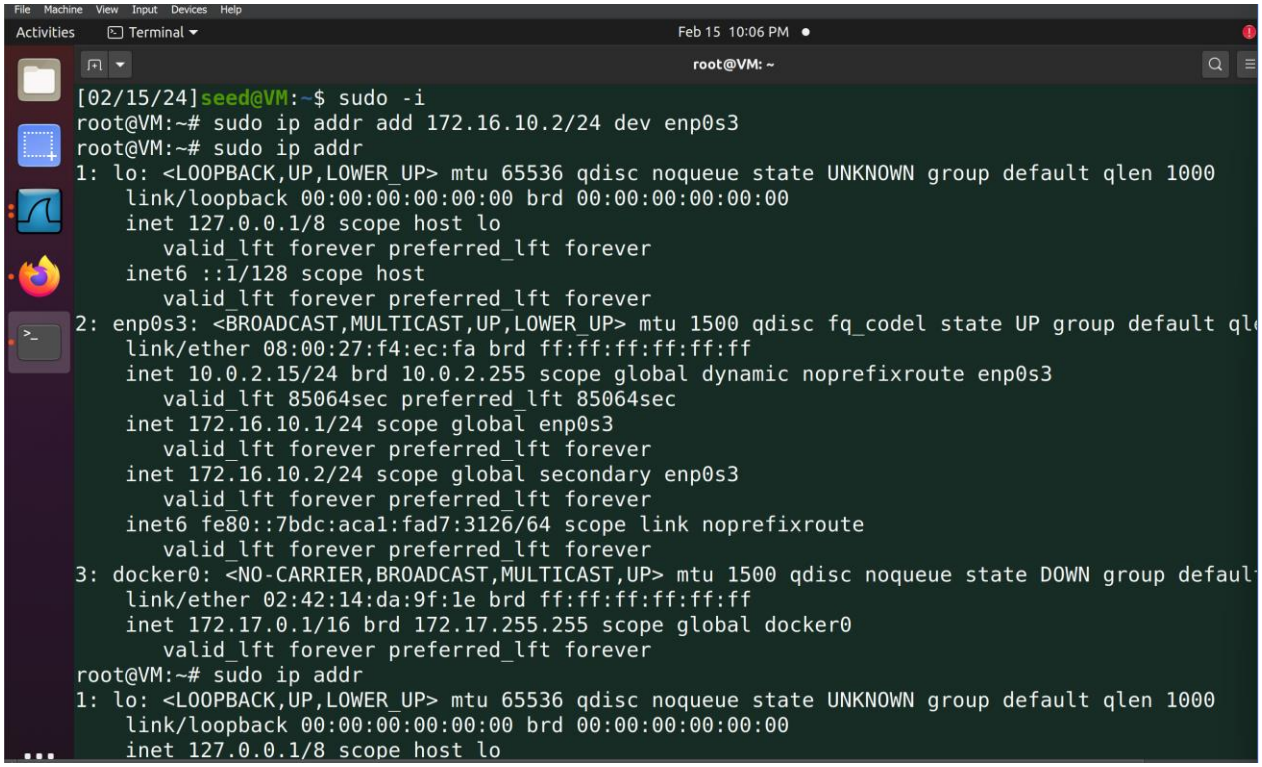
Client side:

Client IP address can be set by the following command.

$sudo ip addr add 172.16.10.2/24 dev enps0

$sudo ip addr

Note: If IP address fluctuates, kindly setup the IP address manually using ‘Edit connections’.



There are broadly two parts of execution:

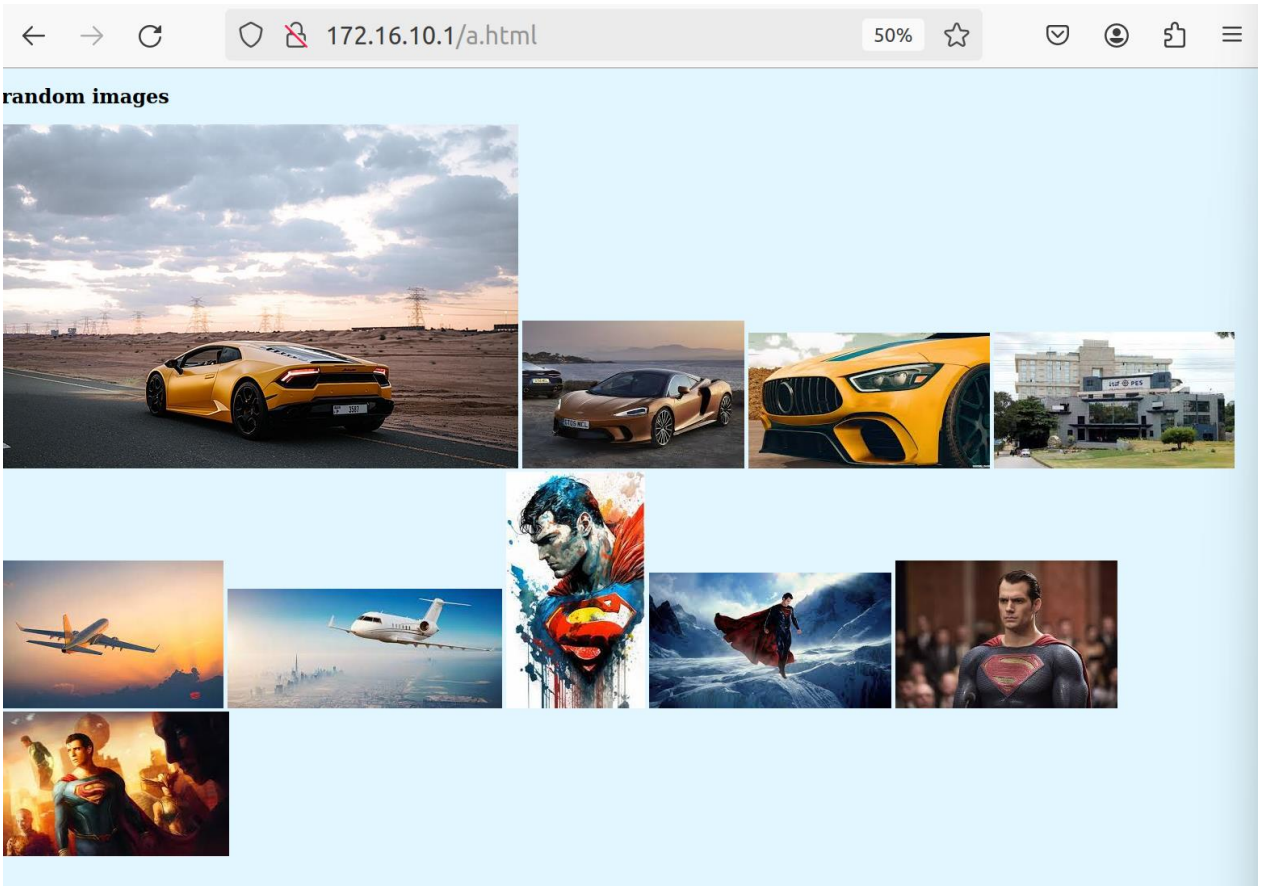
1. Dealing with non-persistent connections
2. Dealing with persistent connections

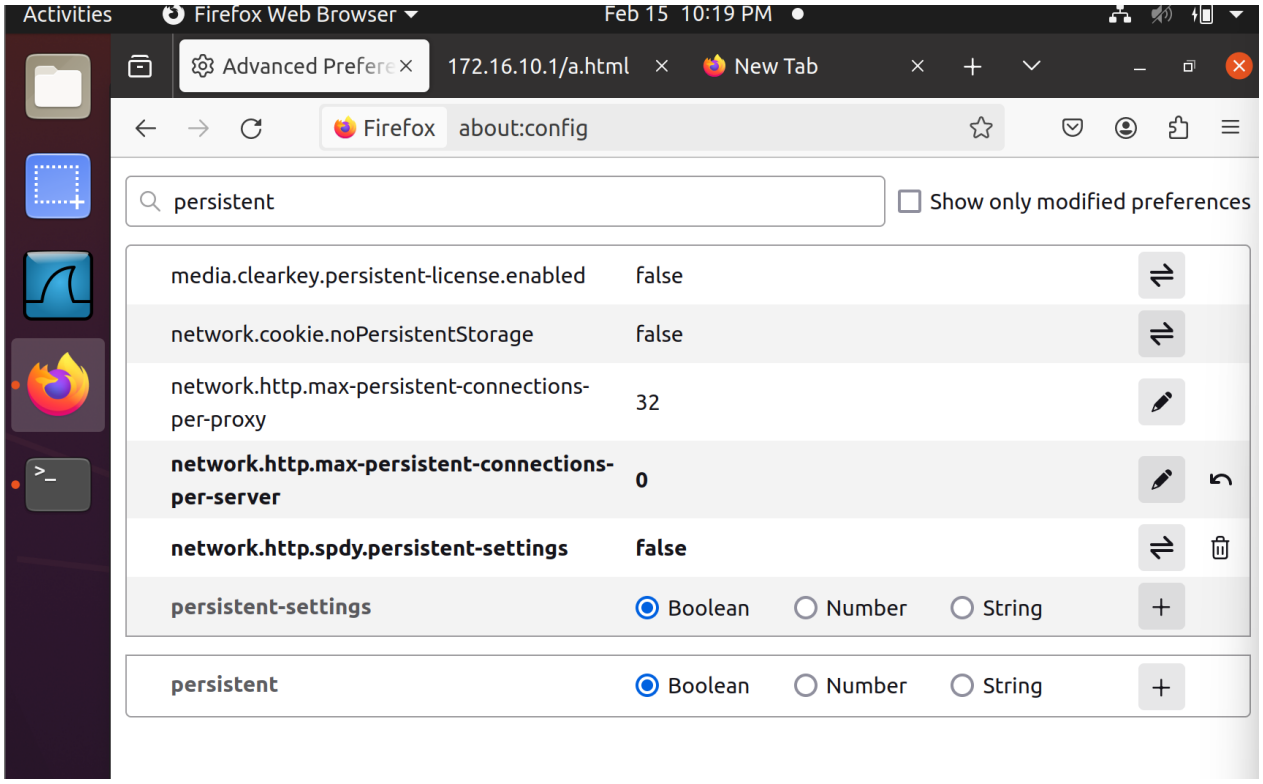
Open Firefox browser to configure for persistent option. Go to browser and type **about:config** and search for the term **'persistent'**

- While using non-persistent connection experiment, the **max-persistent-connectionsper-server** has the value set to **0** and **persistent-settings** value set to false.
- While using persistent connection experiment, the **max-persistent-connections-perserver** should have value greater than 0 (depending on the number of persistent connections needed) and **persistent-settings** value set to true.

PART 1: NON-PERSISTENT CONNECTION

Step 1: This is done by setting the value of max-persistent-connection-per-server to 0 in the client computer.


172.16.10.1/a.html
50%
random images


Activities
Firefox Web Browser
Feb 15 10:19 PM
Advanced Prefere
172.16.10.1/a.html
New Tab
Firefox
about:config
persistent
Show only modified preferences
media.clearkey.persistent-license.enabled
false
network.cookie.noPersistentStorage
false
network.http.max-persistent-connections-per-proxy
32
network.http.max-persistent-connections-per-server
0
network.http.spdy.persistent-settings
false
persistent-settings
Boolean
Number
String
persistent
Boolean
Number
String

Step 2: Access web page on client-side browser (Firefox)

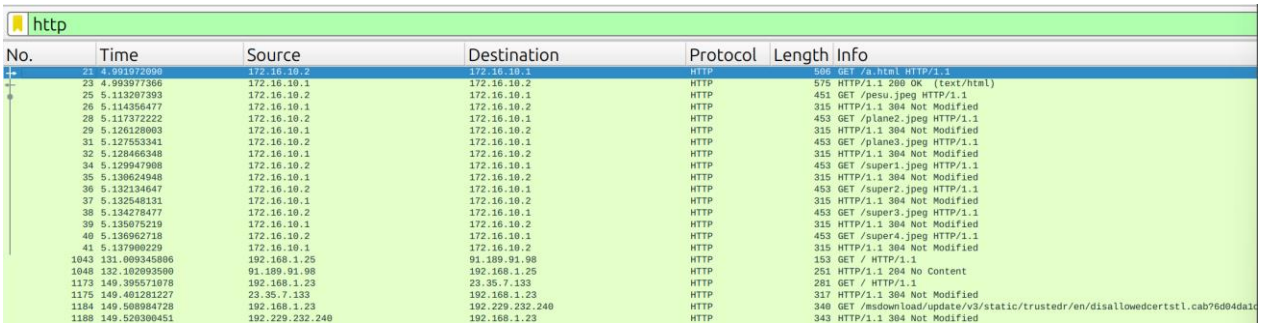
The client could access the file as:

172.16.10.1/file_name.html where--> **172.16.10.1** is Server's IP

Here the file name is **a.html** present in server. So, by tying **172.16.10.1/a.html** in client browser, we will be able to open the requested web page.

Note 1: The wireshark should capture the packets between the client and the server while the file is accessed.
Note 2: The images in the HTML page should have all the permissions specified through the server for the proper access.

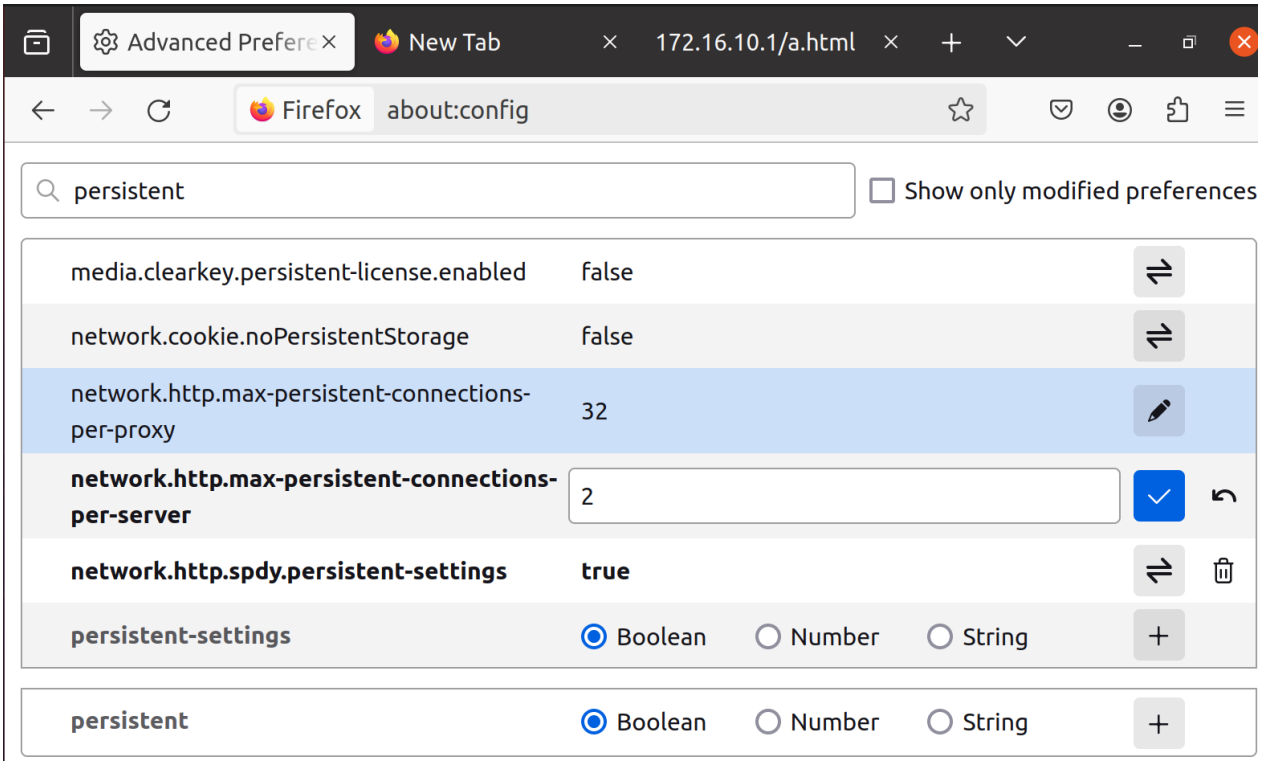
Step 3: Use wireshark**.** Open wireshark in the server computer while client is trying to access the server's local host webpage. Apply 'http' filter and note the time to capture all the 10 images.



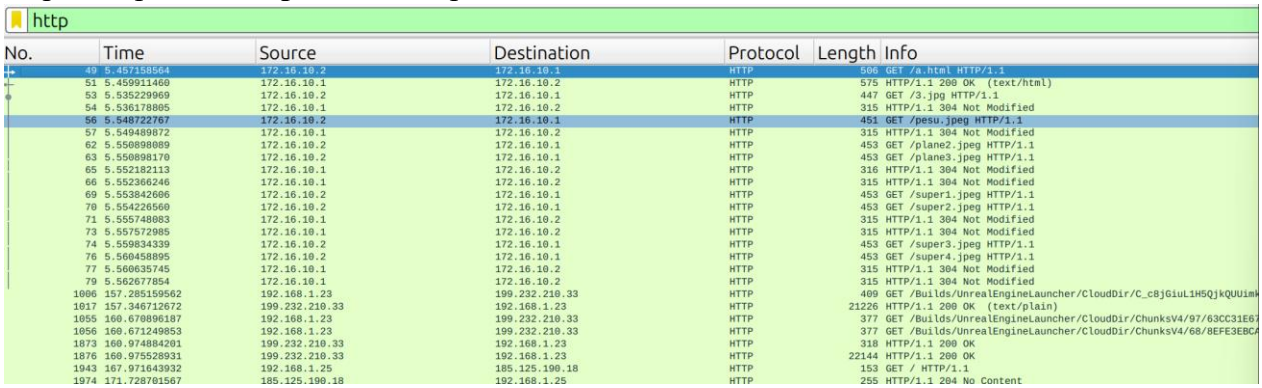
Here it is 149..50089 – 4.9939 = 144.50699

PART 2: PERSISTENT CONNECTIONS

Step 1: For 2 persistent connections, set the value of **max-persistent-connection-per-server to 2** in the client computer.

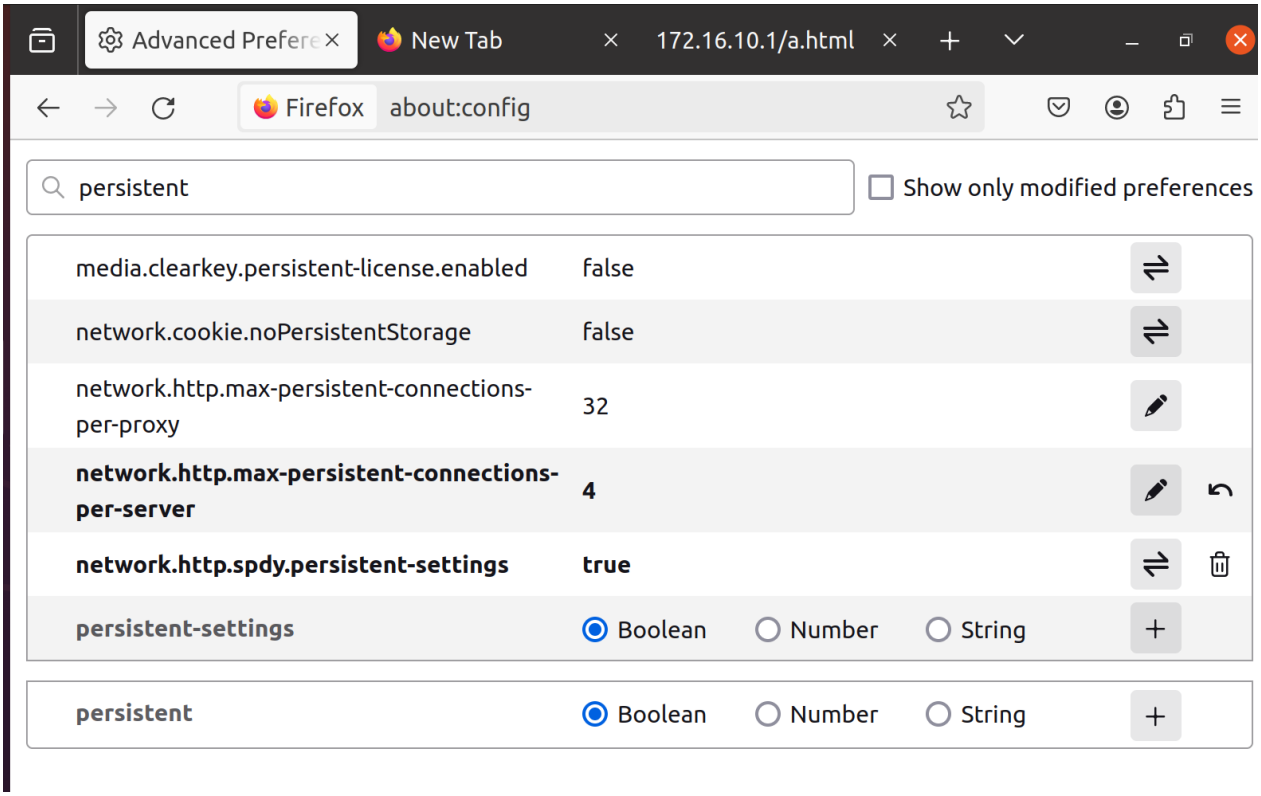


Step 2: Repeat the **steps 1-3** in the previous section.

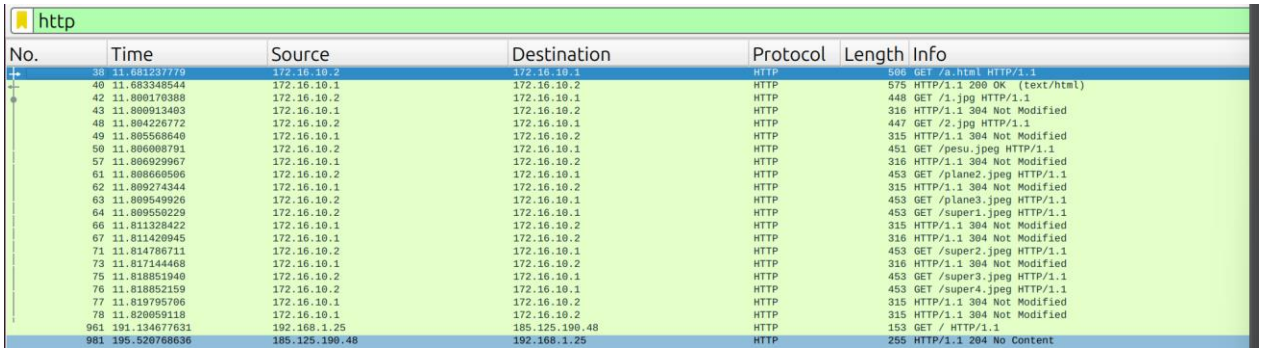


Here it is 167.97164– 5.459911 = 162.511729

Step 3: For 4 persistent connections, Set the value of **max-persistent-connection-per-server to 4** in the client computer.

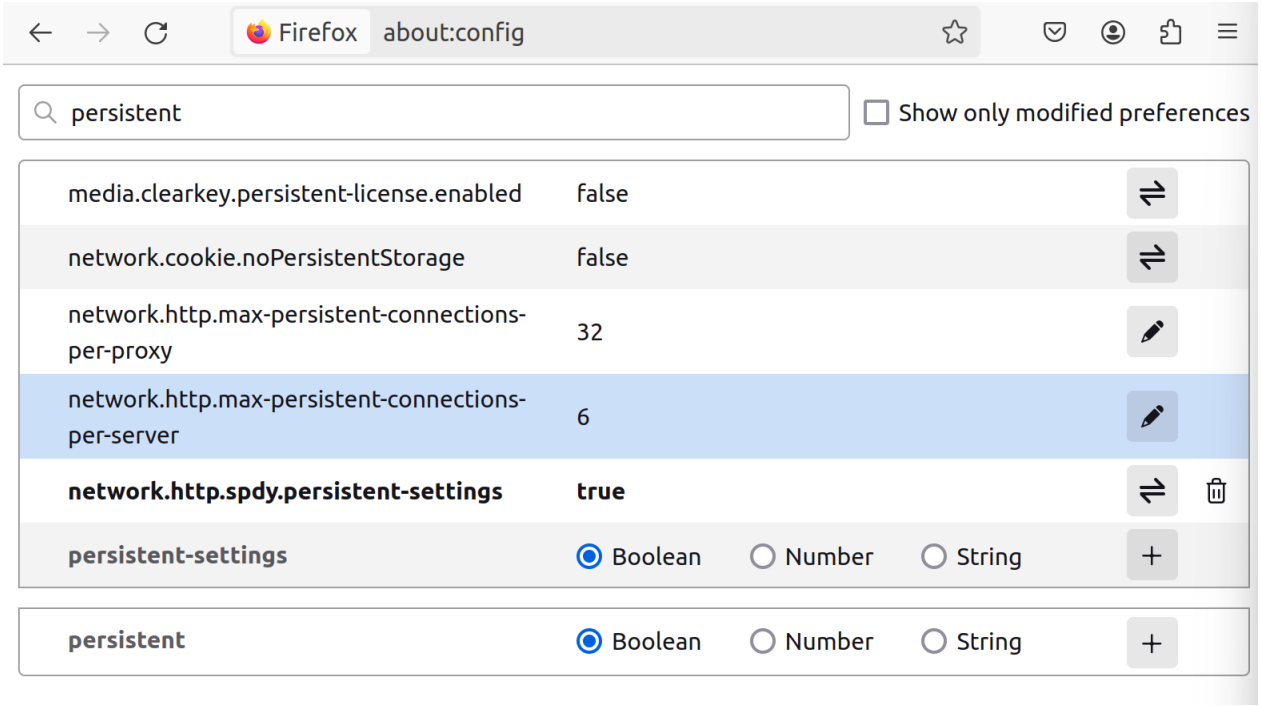


Step 4: Repeat the **steps 1-3** in the previous section.

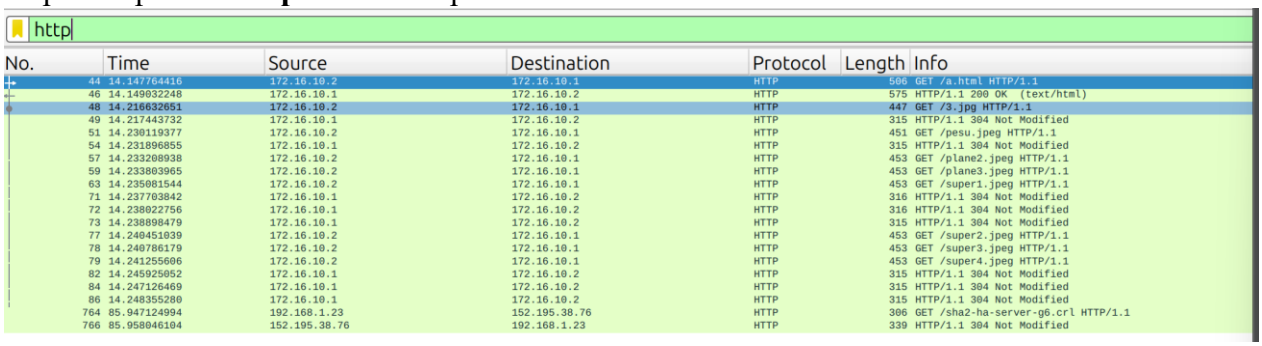


Here it is 195.520768– 11.681237 = 183.839531

Step 5: For 6 persistent connections, set the value of **max-persistent-connection-per-server to 6** in the server computer.

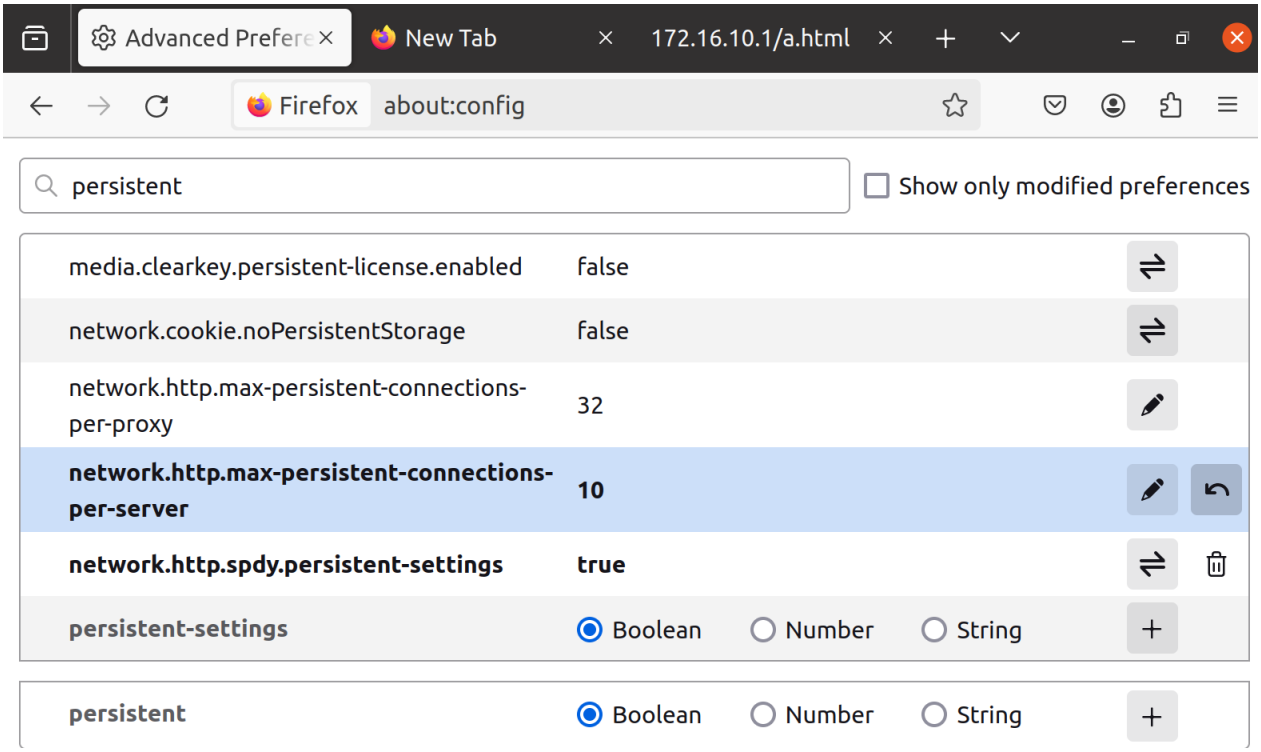


Step 6: Repeat the **steps 1-3** in the previous section.

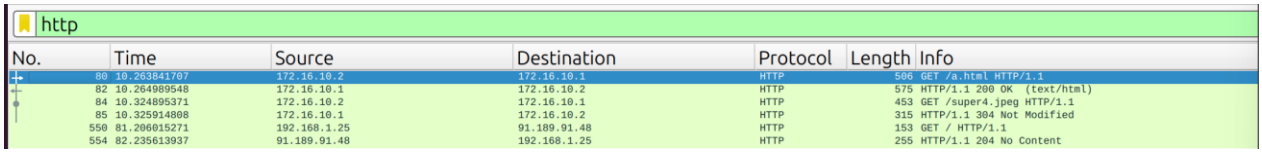


Here it is 85.958046 – 14.149032 = 71.899014

Step 7: For 10 persistent connections, set the value of **max-persistent-connection-perserver to 10** in the client computer.



Step 8: Repeat the **steps 1-3** in the previous section.

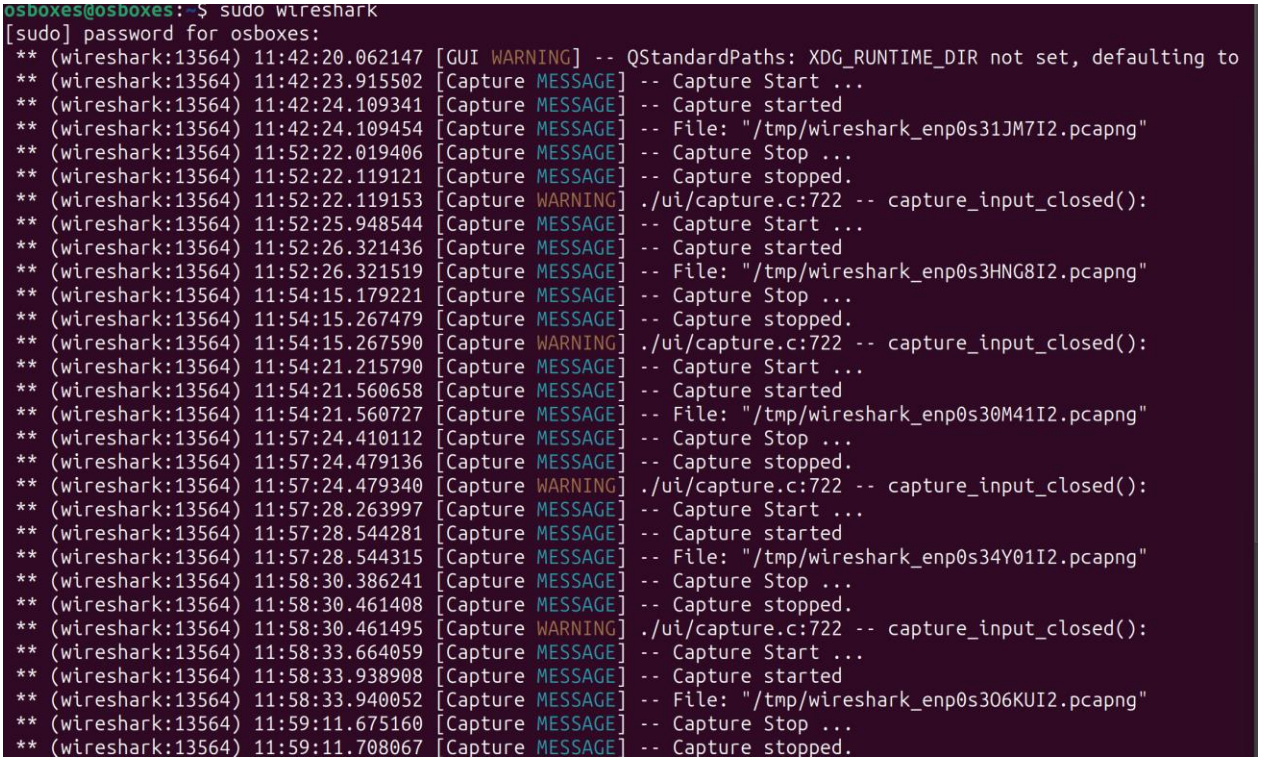


Here it is 81.23561 – 10.263841 =70.971769

OBSERVATIONS REQUIRED ON EDMODO:

Find out the time taken to load images for 2 4 6 persistent connections is lesser or greater than 10 persistent compared to non-persistent. Why? Find out the optimal persistent connections.

SCREENSHOTS REQUIRED FOR EDMODO:



1) Non-persistent connection wireshark capture (should include all 10 images)
2) Persistent connections wireshark capture – 2, 4, 6, 8 & 10 respectively (should include all 10 images).