Week #4

Week 4 - Persistent, Non-Persistent, Authentication, Cookies and Conditional GET

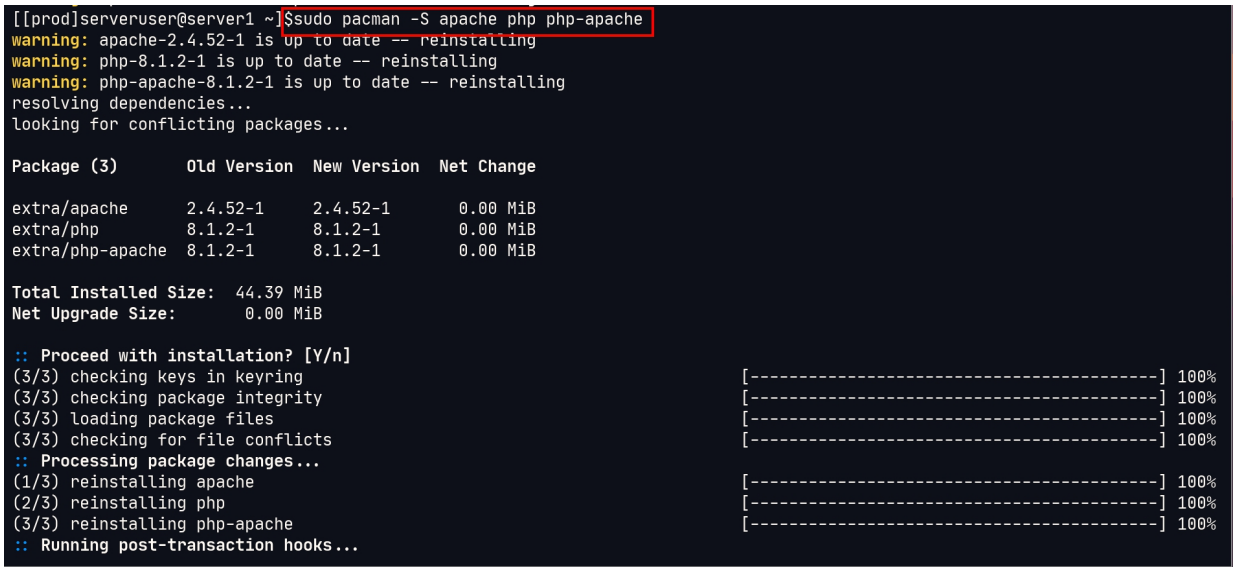
SRN : PES2UG20CS237

Name : P K Navin Shrinivas

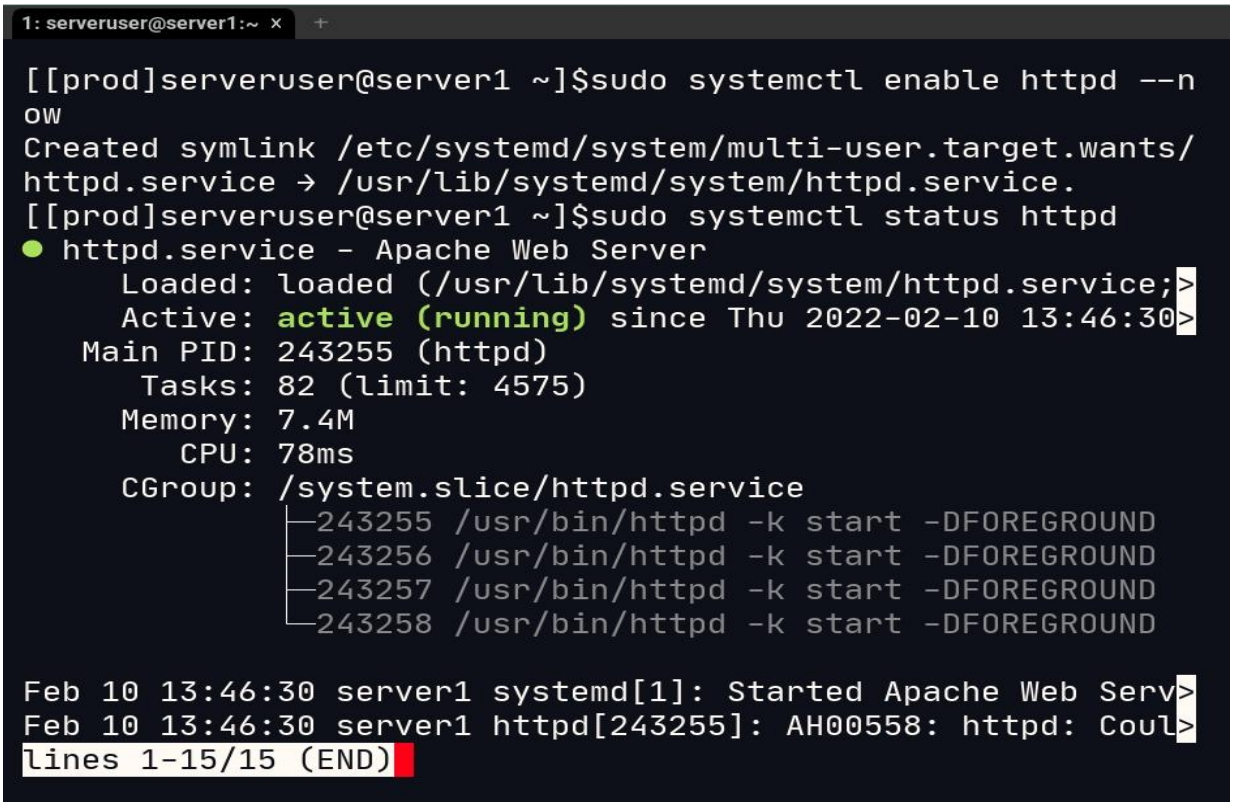
Section : D

Task 1 : Setup a apache server

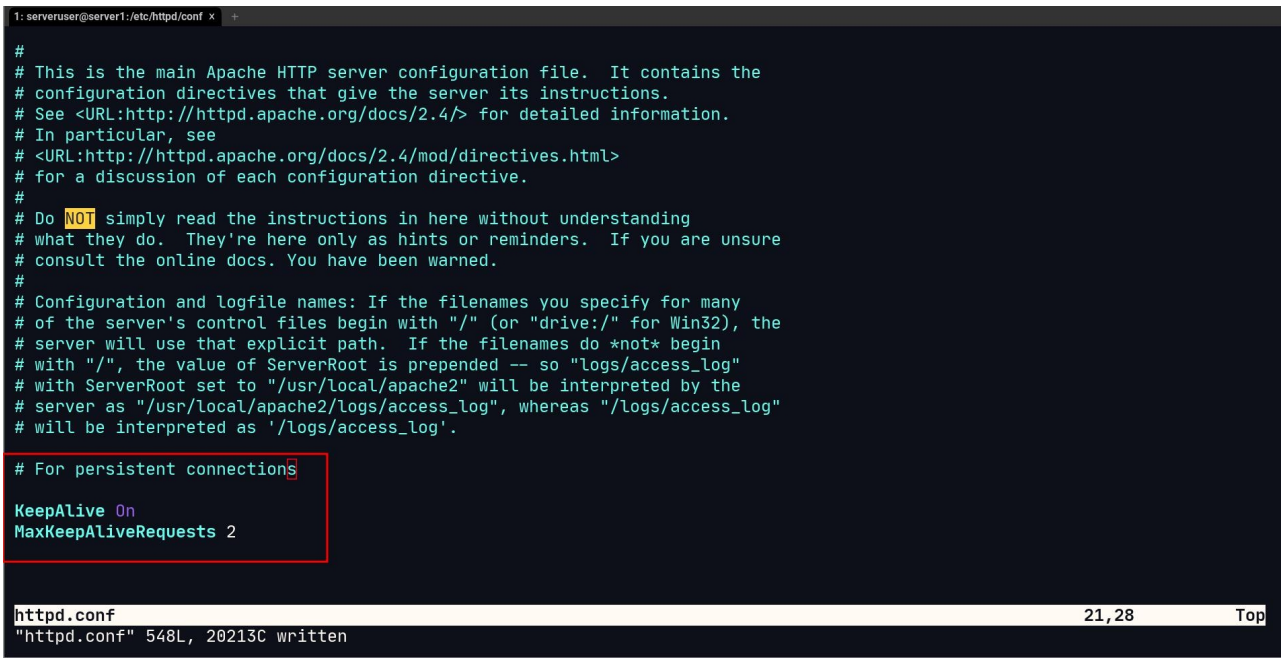
- My setup includes 2 sepereate linux machines within my netowrk, both runnning arch linux.
- First packages needed were installed using the following command :



- Apache server was then turned on and its status was checked to be “active” :



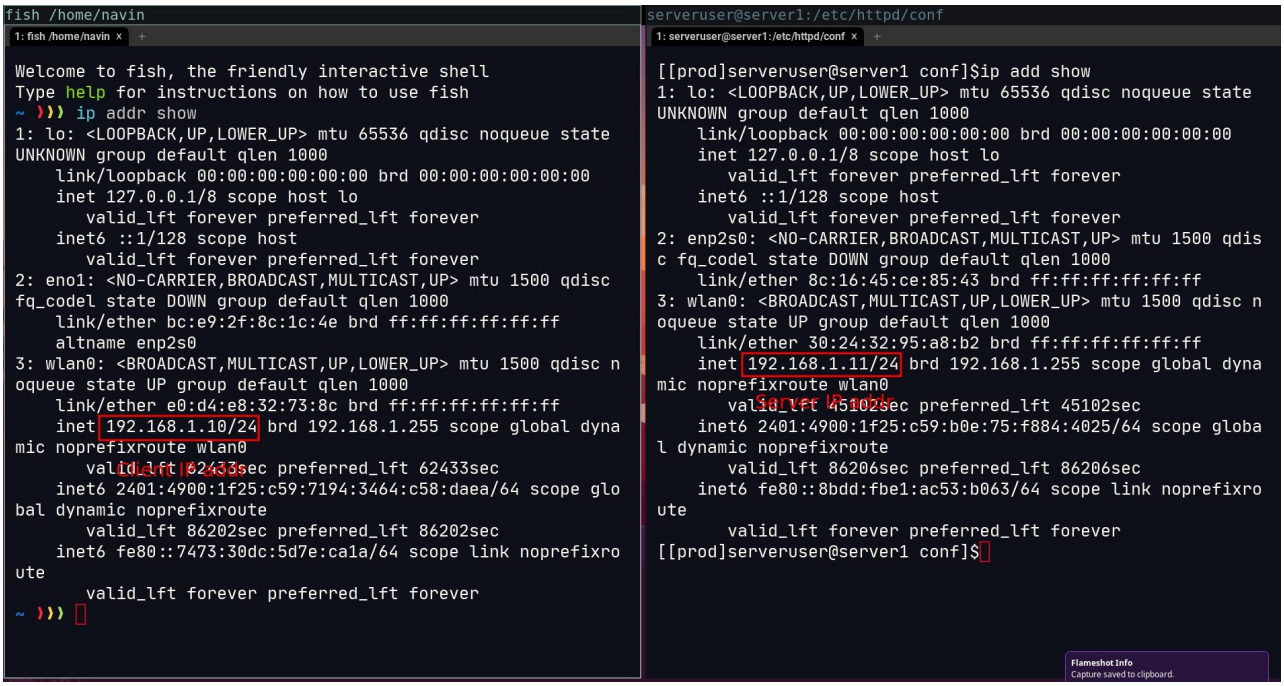
- Apache server was configured to allow persistent connections by modifying the /etc/httpd/conf/httpd.cong (specific to distro) was modified like so :



- The apache server was restarted after saving changes to the above

file suing : sudo systemctl reststart httpd

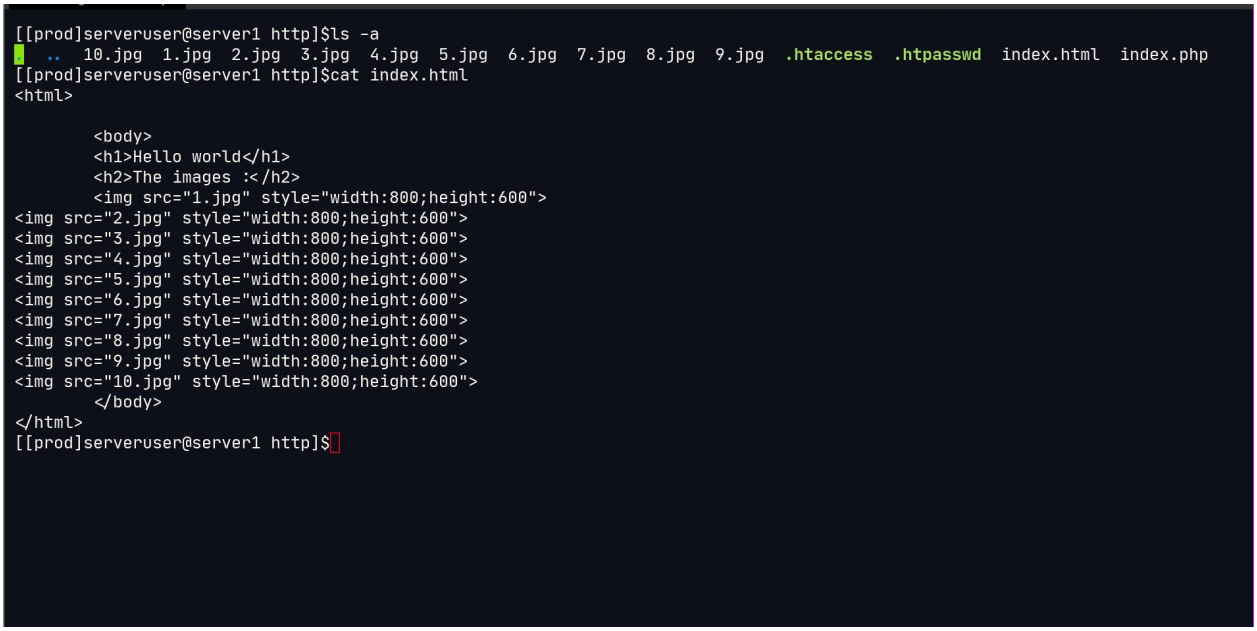
- Both the system were assigned status ip local addresses from the router itself :



Client : 192.168.1.10

Server : 192.168.1.11

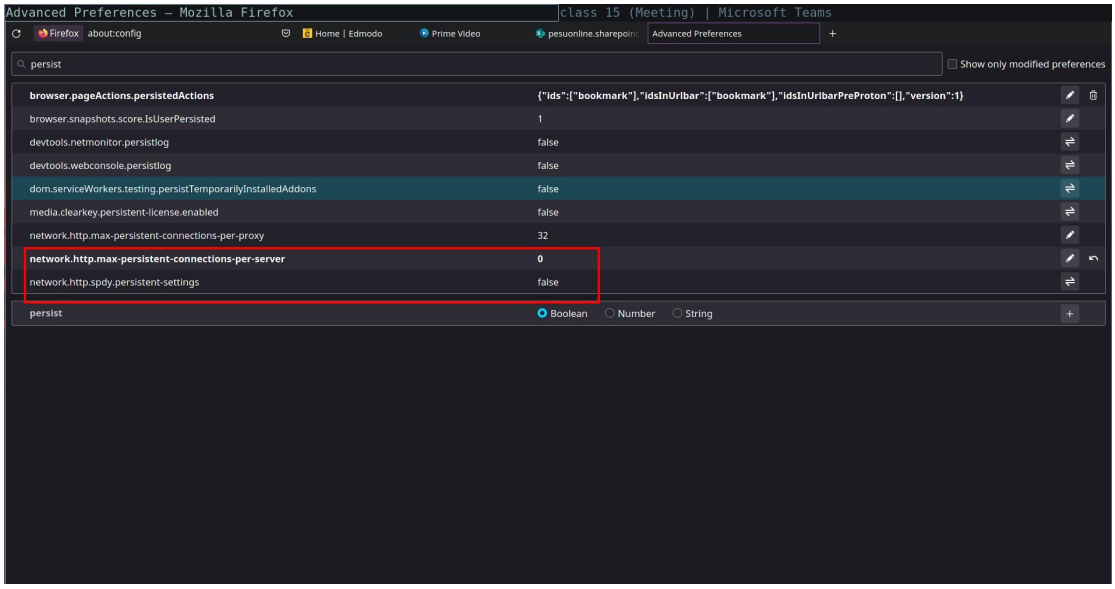
- The Webpage is hosted by copying over the provided photos to the respective folder and writing an index.html file :



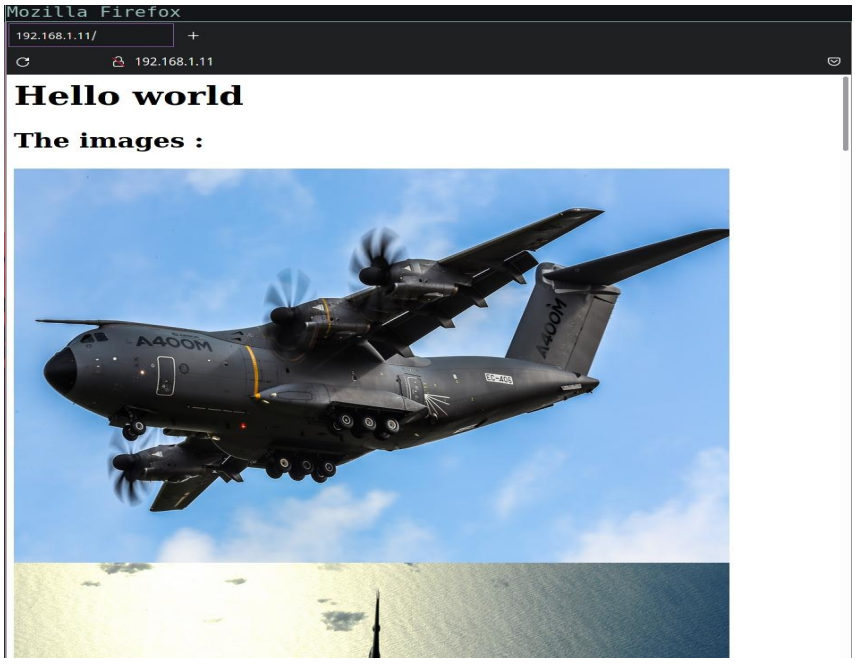
The extra folder are not to confused as they are used in later tasks, only the .jpg and index.html folder is needed for task 1.

Task 2 :Non-Persistent Connections

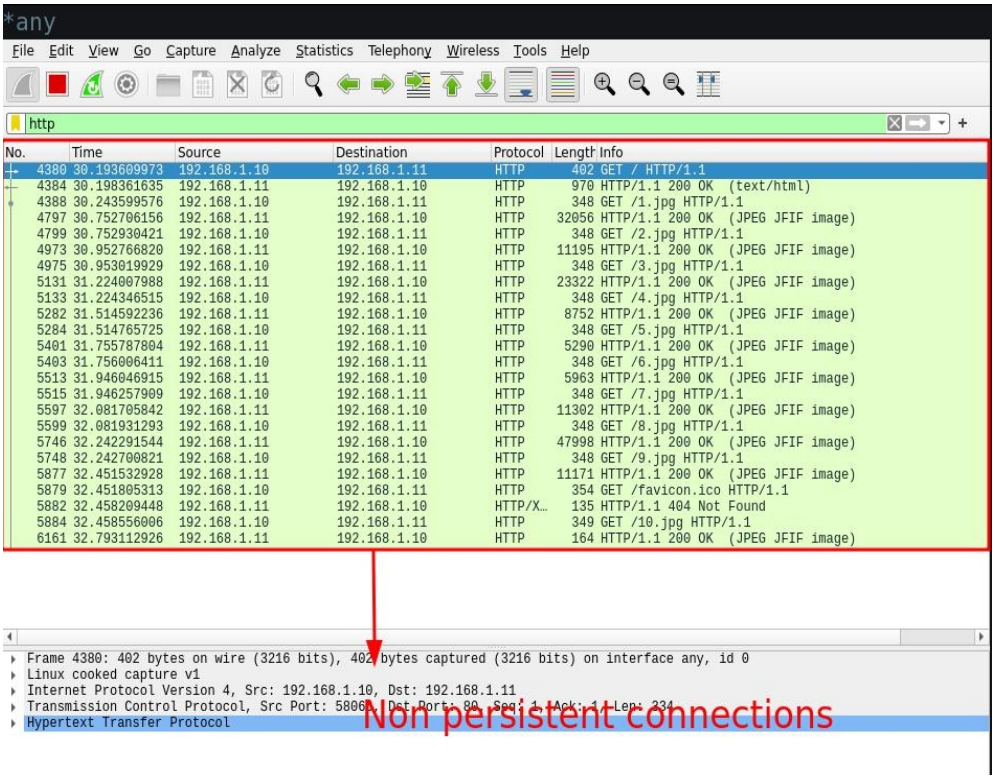
- Open firefox and enter address about:config , and press accpet risk and continue if asked.
- Once done, search for persist and set : **max-persistent-connections-per-server** to 0 and set **persistent-settings** to false, like so :



- Now, ping your server from firefox by simply entering the ip address, like so :

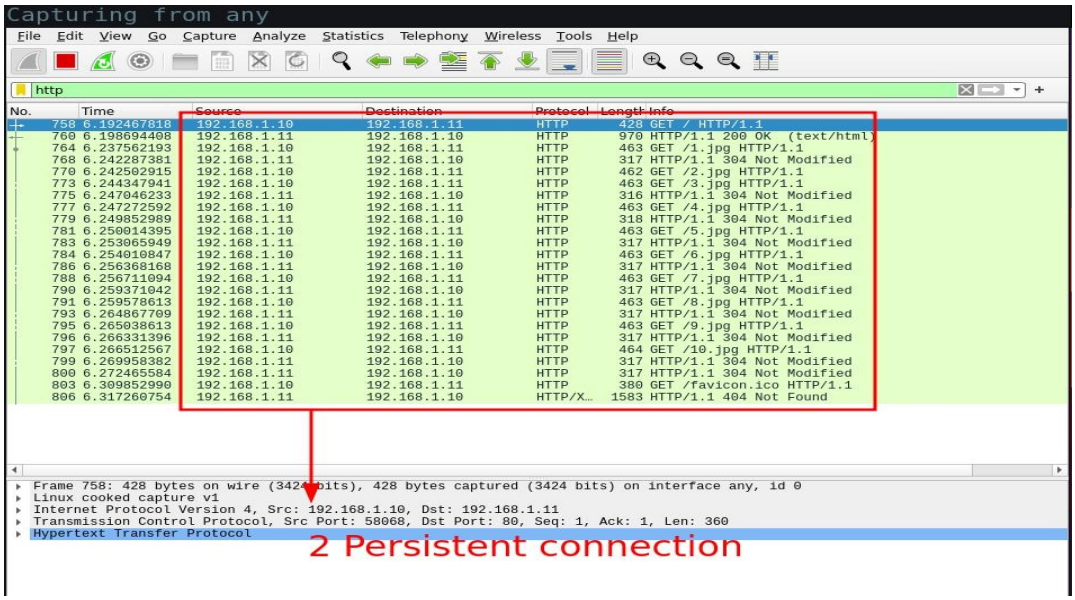


- Analysing the packets in wirehark make it clear that we are handling non persistent connections :

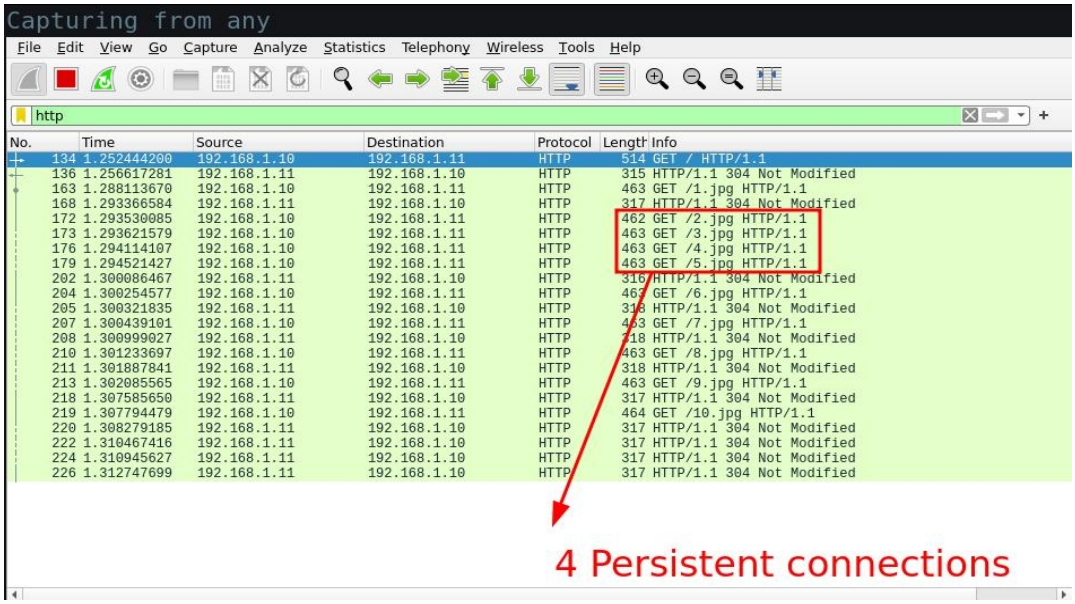


Task 3 : Persistent Connections

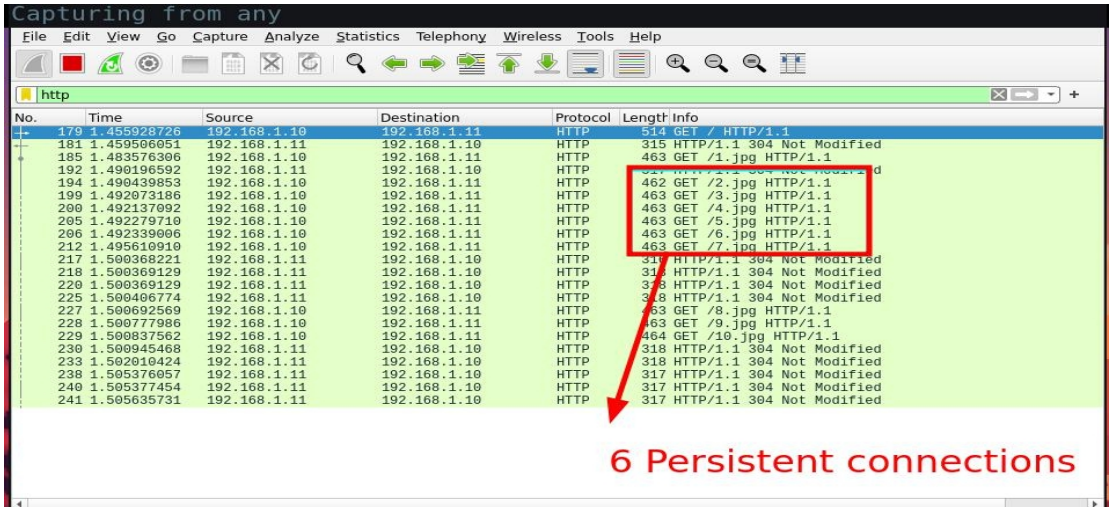
- Open firefox again and got to about:config and set: **max-persistent-connections-per-server** to 2,4,6..
  persistent-settings to true
- 2 persist connections



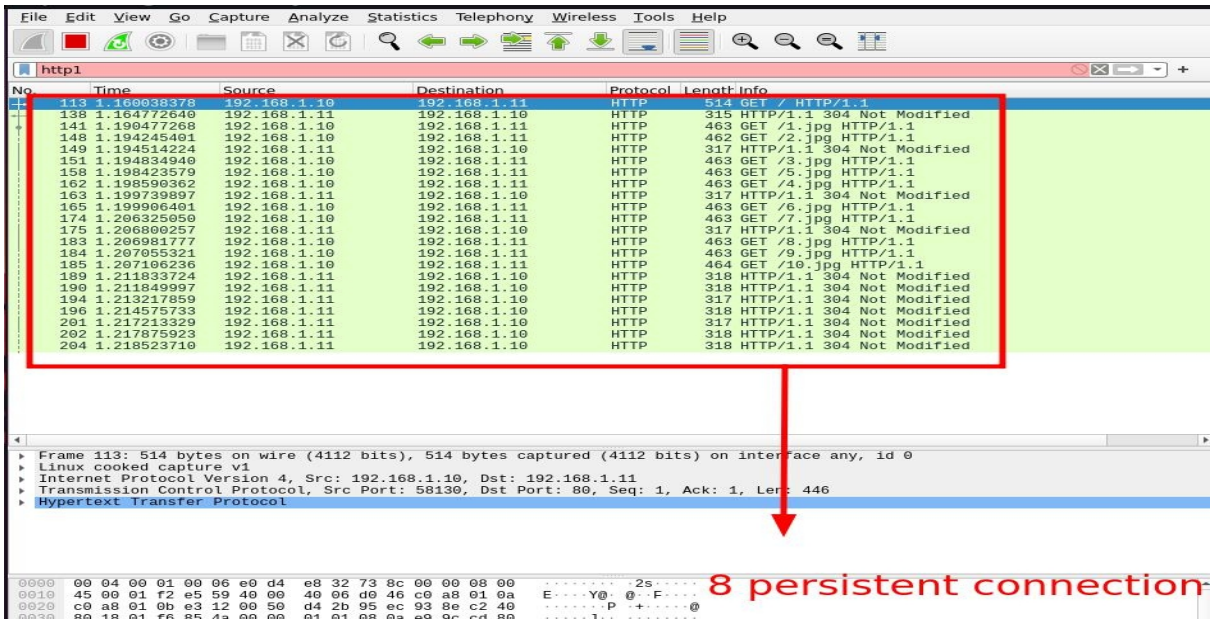
- 4 persist connections



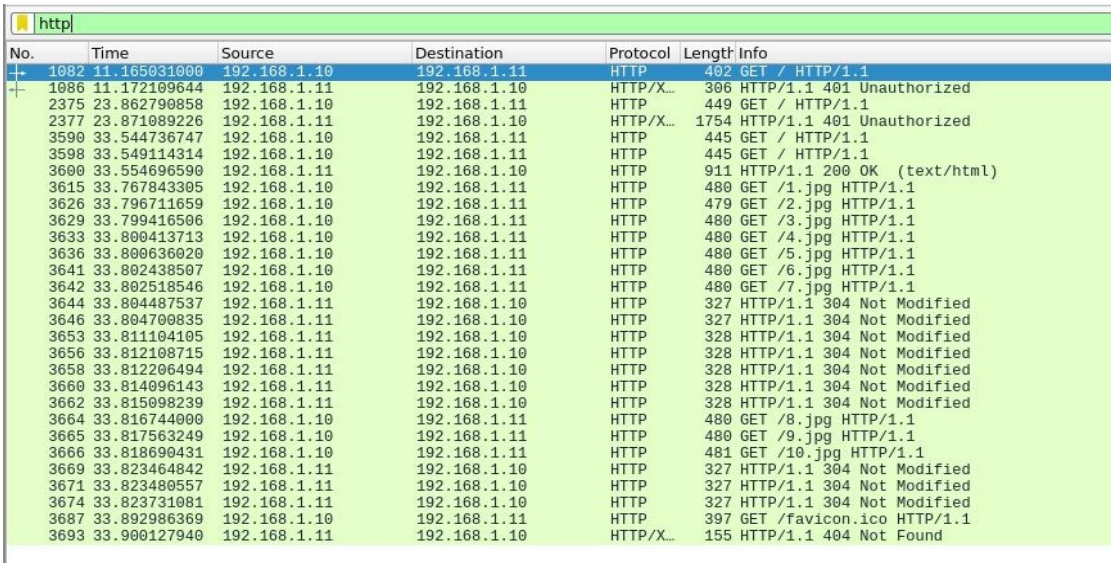
- 6 persist connections



- 8 persist connections



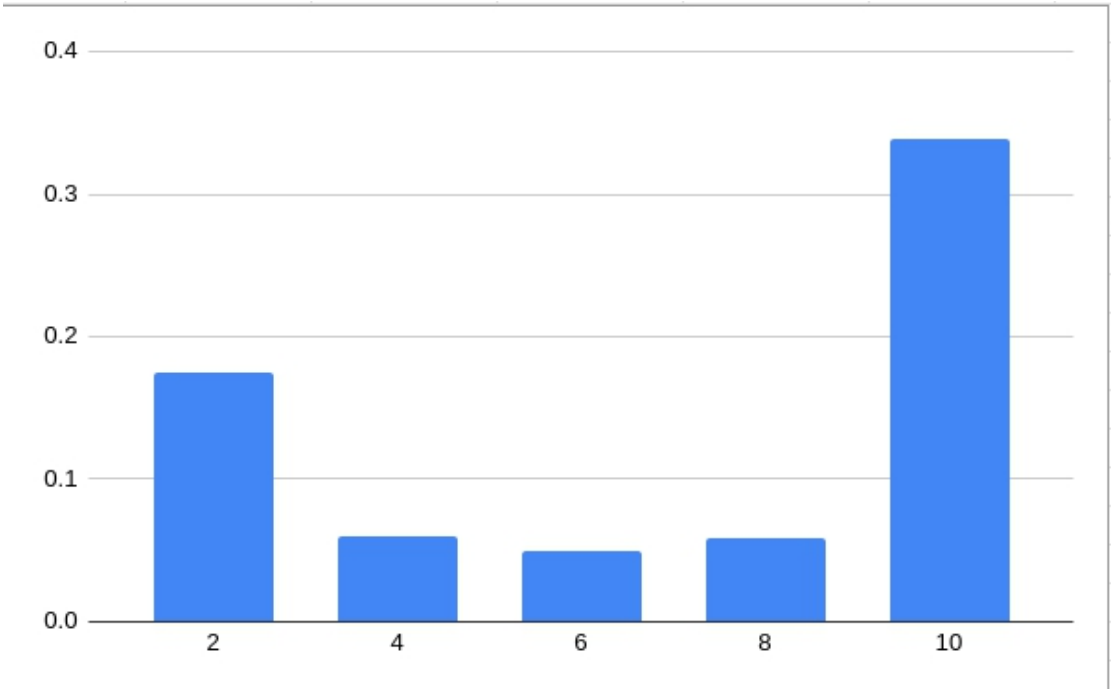
- 10connections



- Obsevations from task 3 :

  Calcuations and tabluating load time using wireshark :

Persistence	First GET time	Last Response	Load time
0	30.19360	32.71399	2.52039
2	6.14296	6.31726	0.1743
4	1.25244	1.31274	0.0603
6	1.45592	1.50563	0.04971
8	1.16003	1.21852	0.05849
10	33.55469	33.89298	0.33829



It is observed that the best number of persistent connection is 6.

Task 4 : Authentication in Apache server

- Create a .htpasswd file in any place in the system preferably outside the server root. Like so :

```
[[prod]serveruser@server1 conf]$sudo htpasswd -c /srv/http/.htpasswd navin
New password:
Re-type new password:
Adding password for user navin
[[prod]serveruser@server1 conf]$
```

- Edit the apache config file in /etc/httpd/conf/httpd.conf file as so :

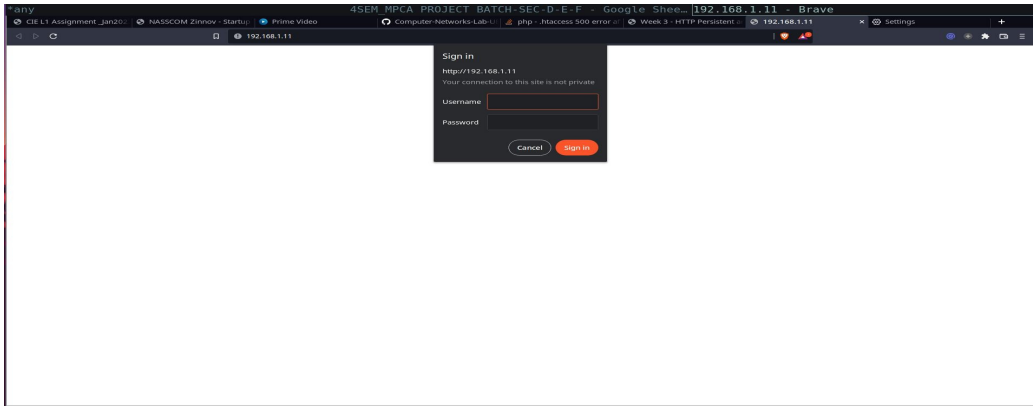
```
1: serveruser@server1:/etc/httpd/conf

#
# Note that from this point forward you must specifically allow
# particular features to be enabled - so if something's not working as
# you might expect, make sure that you have specifically enabled it
# below.
#

#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "/srv/http"
<Directory "/srv/http">
AuthType Basic
AuthName "Restricted Access"
AuthUserFile /srv/http/.htpasswd
Require valid-user >
    #
    # Possible values for the Options directive are "None", "All",
    # or any combination of:
    #   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
    #
    # Note that "MultiViews" must be named *explicitly* --- "Options All"
    # doesn't give it to you.
    #
httpd.conf                                                    262,16        47%
/srv                                                          [1/6]
```

Auth config for apache

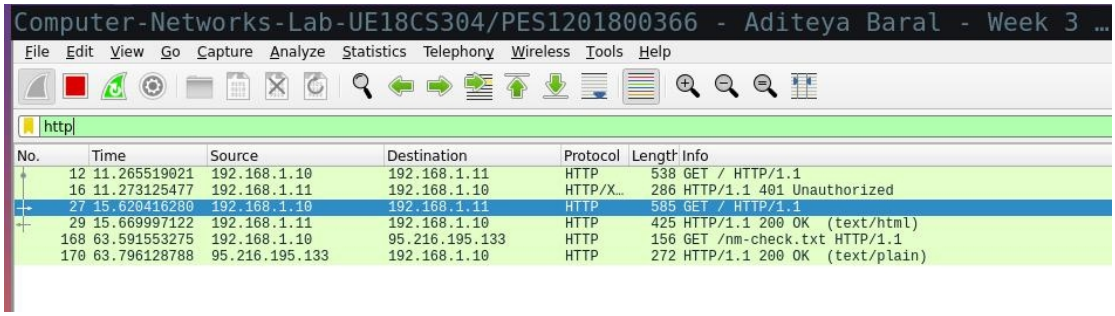
- Create an .htaccess file in the server route with the following contents :

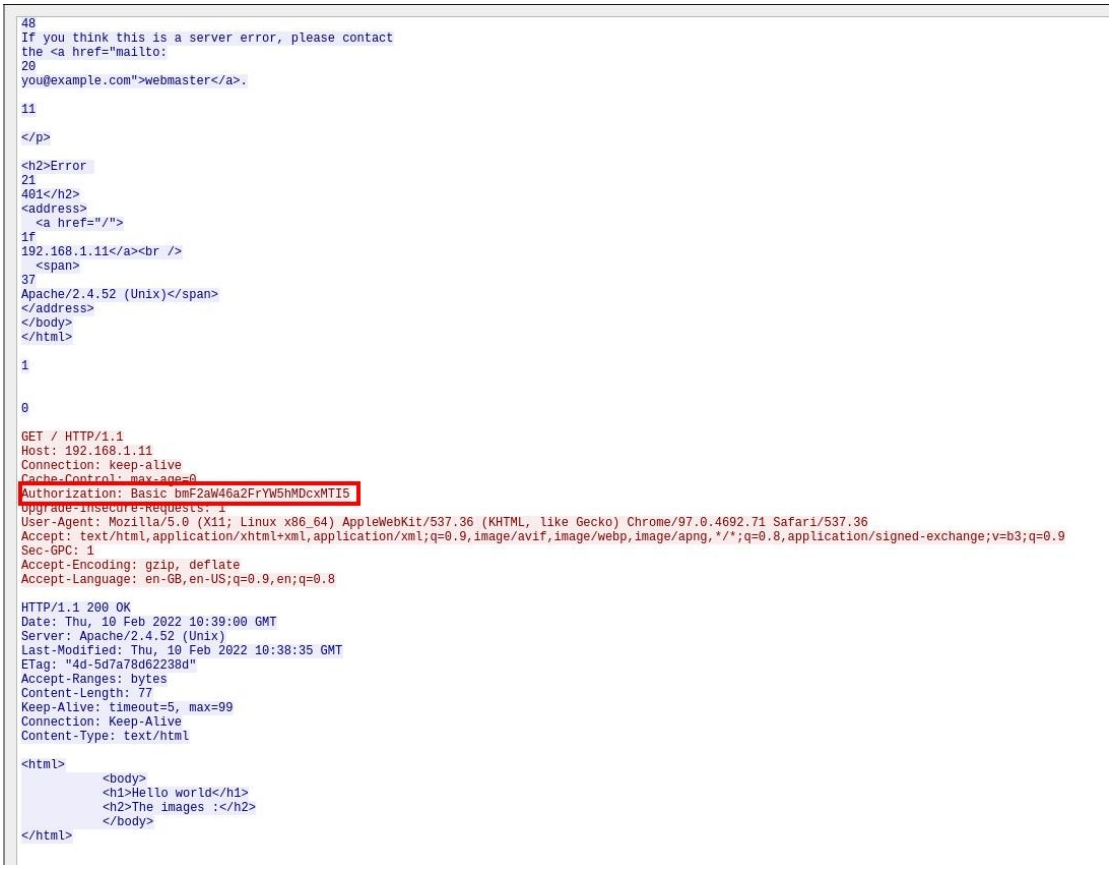
```
[[prod]serveruser@server1 http]$ls -a
.  ..  10.jpg  1.jpg  2.jpg  3.jpg  4.jpg  5.jpg  6.jpg  7.jpg  8.jpg  9.jpg  .htaccess  .htpasswd  index.html  index.php
[[prod]serveruser@server1 http]$cat .htaccess
AuthType Basic
AuthName "Restricted Access"
AuthUserFile /srv/http/.htpasswd
Require valid-user >
```

- Opening the webpage should lead to prompt :



- Analyzing the packets in wireshark :





- Analyzing base64 Auth cipher text :

As the Authneitcation follows a base 64 addressing, each letter in cipher is of 6 bits and which can later be coverted to ascii of 8 bits each, this goves us :

011011 100110 000101 110110 011010 010110 111000 111010

011011 100110 000101 110110 011010 010110 111000 110001

001100 100011 001100 110100

rearraging to ascii 8 bits :

01101110 01100001 01110110 01101001 01101110 00111010 01101110 01100001 01110110 01101001 01101110 00110001 00110010 00110011 00110100

This leads to : navin:navin1234, whcih was indeed my username:password

Task 5 : Cookies using PHP

- For this a new index.php file is created in the server root, which has html content with php embedded, like so :

```
[[prod]serveruser@server1 http]$ls -a
.  ..  10.jpg  1.jpg  2.jpg  3.jpg  4.jpg  5.jpg  6.jpg  7.jpg  8.jpg  9.jpg  .htaccess  .htpasswd  index.html  index.php
[[prod]serveruser@server1 http]$cat index.php
<html>
    <?php
        setcookie("SRN","PES2UG20CS237");
        setcookie("Name","Navin Shrinivas",time()+125);
    ?>
        <body>
        <h1>Hello world</h1>
        </body>
</html>
[[prod]serveruser@server1 http]$
```

- PHP is enabled in the apache server by editing the conf file, like so :

```
1: serveruser@server1:/etc/httpd/conf

# have to place corresponding `LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Statically compiled modules (those listed by `httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#

LoadModule php_module modules/libphp.so
AddHandler php-script .php
Include conf/extra/php_module.conf

#LoadModule mpm_event_module modules/mod_mpm_event.so
LoadModule mpm_prefork_module modules/mod_mpm_prefork.so
#LoadModule mpm_worker_module modules/mod_mpm_worker.so
LoadModule authn_file_module modules/mod_authn_file.so
#LoadModule authn_dbm_module modules/mod_authn_dbm.so
#LoadModule authn_anon_module modules/mod_authn_anon.so
#LoadModule authn_dbd_module modules/mod_authn_dbd.so
#LoadModule authn_socache_module modules/mod_authn_socache.so
LoadModule authn_core_module modules/mod_authn_core.so
LoadModule authz_host_module modules/mod_authz_host.so
LoadModule authz_groupfile_module modules/mod_authz_groupfile.so
LoadModule authz_user_module modules/mod_authz_user.so
#LoadModule authz_dbm_module modules/mod_authz_dbm.so
#LoadModule authz_owner_module modules/mod_authz_owner.so
httpd.conf                                                    91,1          12%
```

Configuring for apache after installing php package

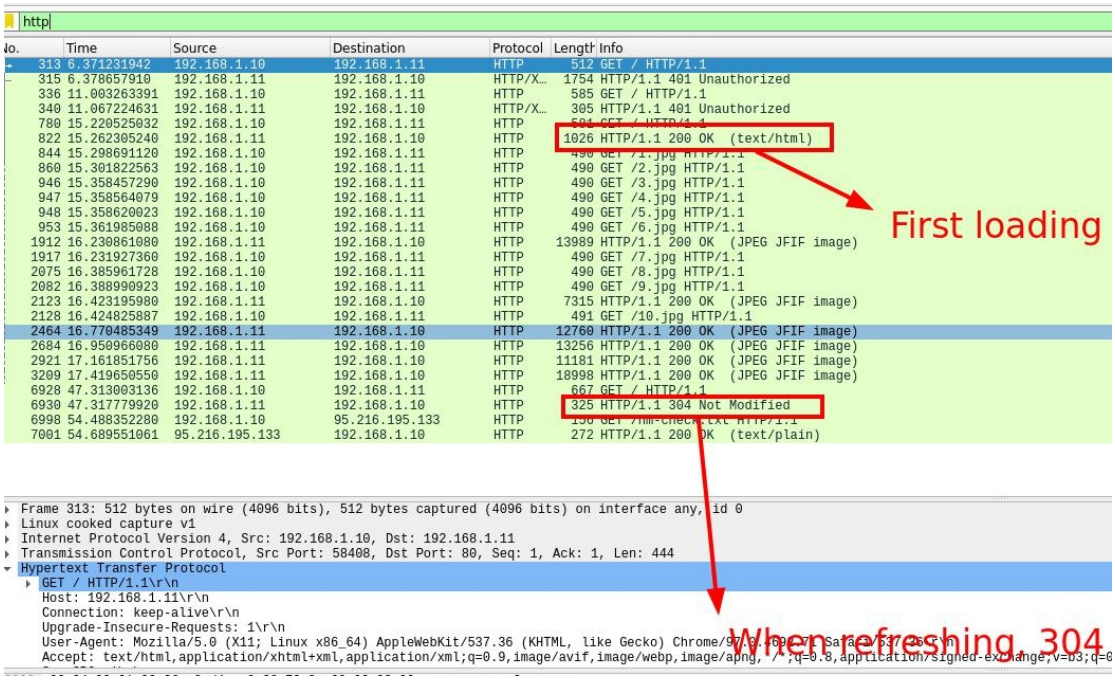
- The PHP file is then accessed from the client and wireshark packets are analysed, the **set-cookie** can be seen :

```
GET /index.php HTTP/1.1
Host: 192.168.1.11
Connection: keep-alive
Cache-Control: max-age=0
Authorization: Basic bmF2aW46bmF2aW4xMjM0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/97.0.4692.71 Safari/537.36
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
Sec-GPC: 1
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8

HTTP/1.1 200 OK
Date: Thu, 10 Feb 2022 12:22:00 GMT
Server: Apache/2.4.52 (Unix) PHP/8.1.2
X-Powered-By: PHP/8.1.2
Set-Cookie: SRN=PES2UG20CS237
Set-Cookie: Name=Navin%20Shrinivas; expires=Thu, 10-Feb-2022 12:24:05 GMT; Max-Age=125
Content-Length: 79
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

<html>
        <body>
        <h1>Hello world</h1>
        </body>
</html>
```

Loaded Cookies

Task 6 : Conditional get

- A conditional HTTP request that carries data only if data has been modified. Apache servers are configured by default to do this
- Code 304 is for not modified, 200 is when the data is actually fetched successfully.
- Below is a demonstration of Conditional get using refreshing pages and wireshark :



Wire shark overlook

200 status on first load after clearing cache

304 not modified status after refreshing