Computer Networks Lab – Week 3
SRN : PES1UG19CS542
Name: Trisha Jain

Understanding Working of HTTP Headers

I. Password Authentication

1. To generate a password file in order to enable the authentication for HTTP we use the **htpasswd** command.

2. Using the cat command we view the encrypted password (Data Encryption Standard Algorithm).

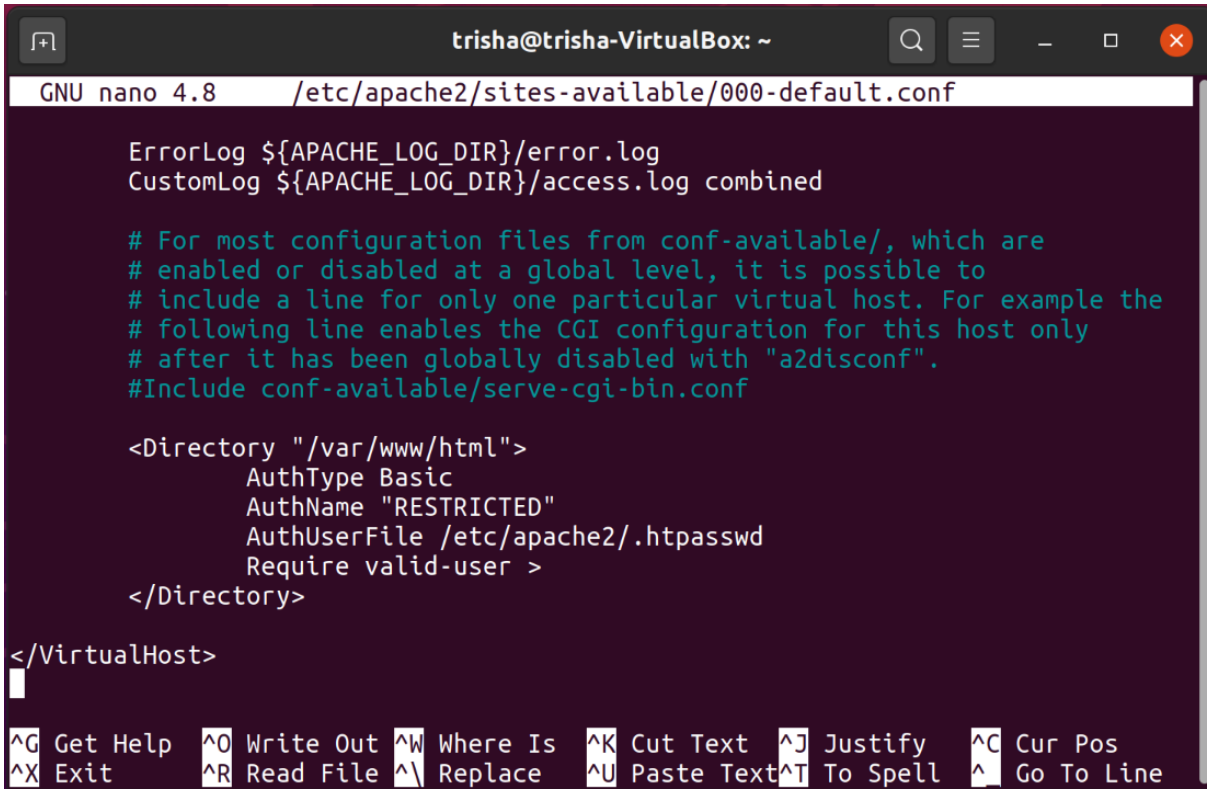
```
trisha@trisha-VirtualBox:~$ sudo htpasswd -c /etc/apache2/.htpasswd trisha
[sudo] password for trisha:
New password:
Re-type new password:
Adding password for user trisha
trisha@trisha-VirtualBox:~$ sudo cat /etc/apache2/.htpasswd
trisha:$apr1$u9hwHt1v$KGNroyhSX/P/0zR5yER3N/
trisha@trisha-VirtualBox:~$
```

sudo htpasswd -c /etc/apache2/.htpasswd

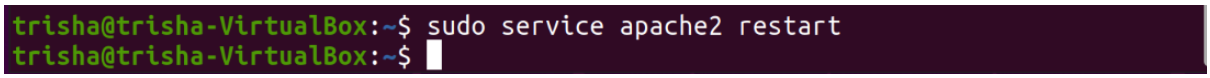
3. For enabling password authentication in the server, the Apache configuration file needs to be modified.

   The authentication was added to the **/var/www/html** directory (localhost home directory).

   Then the server is restarted using the command :
   sudo service apahe2 restart

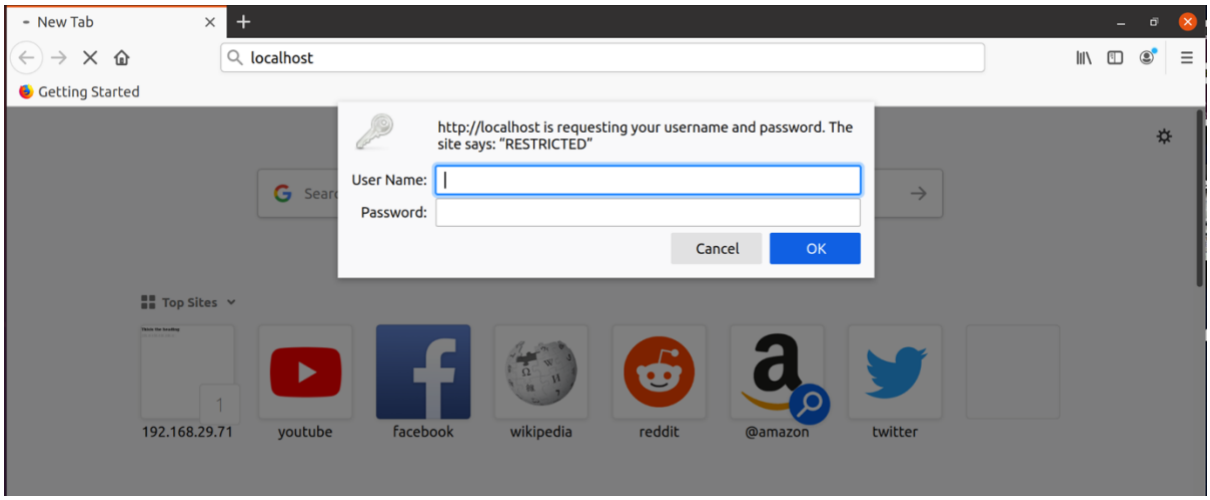


sudo nano /etc/apache2/sites-available/000-default.conf

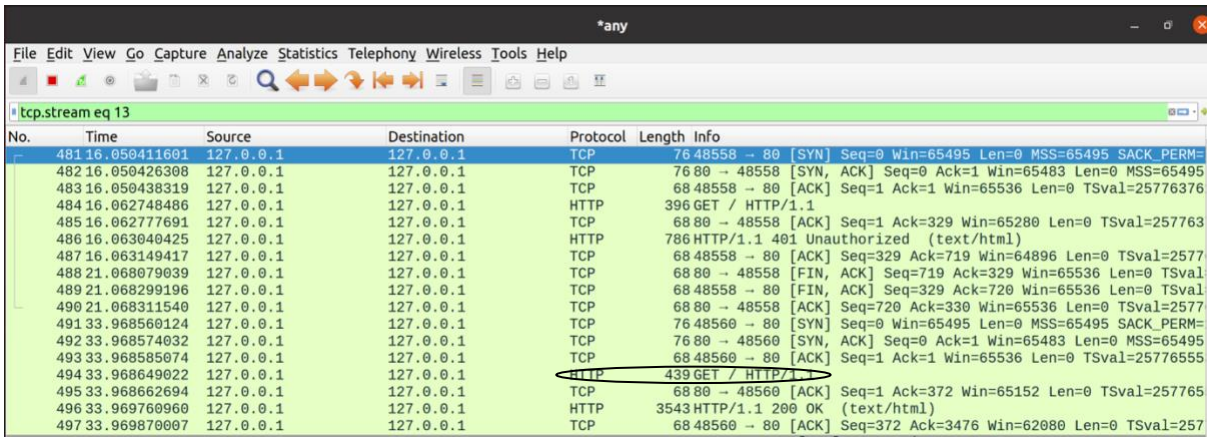


sudo service apache2 restart

The localhost is now accessed using the Firefox browser by entering the username and password that was set by us earlier.



Wireshark is used to capture the packets when the localhost is fetched.



Using the follow TCP Stream on the HTTP message segment the password was retrieved (encrypted by base64 algorithm).

Decrypting the password using the base64 algorithm

Encryption technique used by this algorithm :-

1) Each character is converted into 8-bit binary ASCII representation and grouped into chunks of 6-bits.
2) These chunks are converted into their decimal equivalent and assigned the corresponding Base64 character.

Decryption technique to be used :-

1) Each character's 6-bit binary equivalent is found.
2) Eight bit chunks are made and then decoded to ASCII.
3) This reveals the password that was encrypted by using Base64 algorithm.

ENCRYPTED PASSWORD = dHJpc2hhOm5ldHdvcmtz

Converting to 6-bit binary equivalent :-

d	011101
H	000111
J	001001
p	101001
c	011100
2	110110
h	100001
h	100001
O	001110
m	100110
5	111001
l	100101
d	011101
H	000111
d	011101

v	101111
c	011100
m	100110
t	101101
z	110011

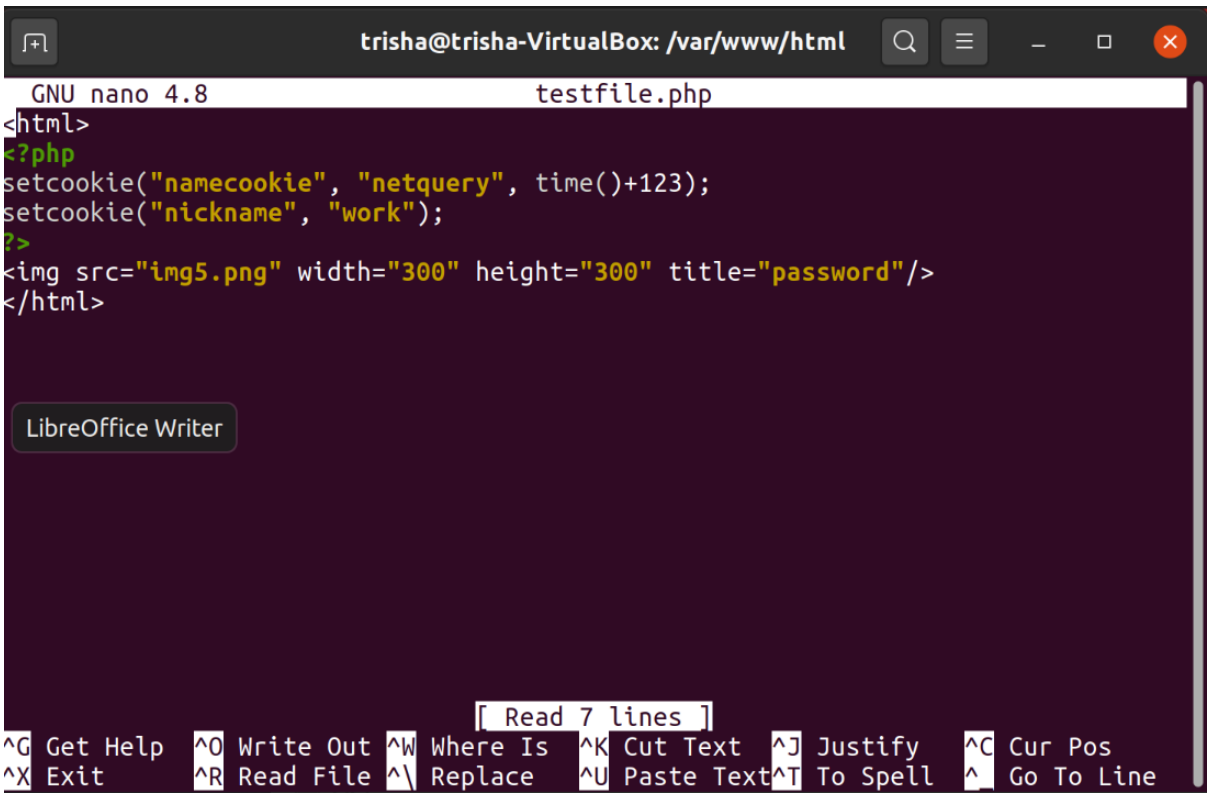
Grouping these binary equivalents to get the ASCII code :-

01110100	t
01110010	r
01101001	i
01110011	s
01101000	h
01100001	a
00111010	:
01101110	n
01100101	e
01110100	t
01110111	w
01101111	o
01110010	r
01101011	k
01110011	s

Username : trisha
Password : networks

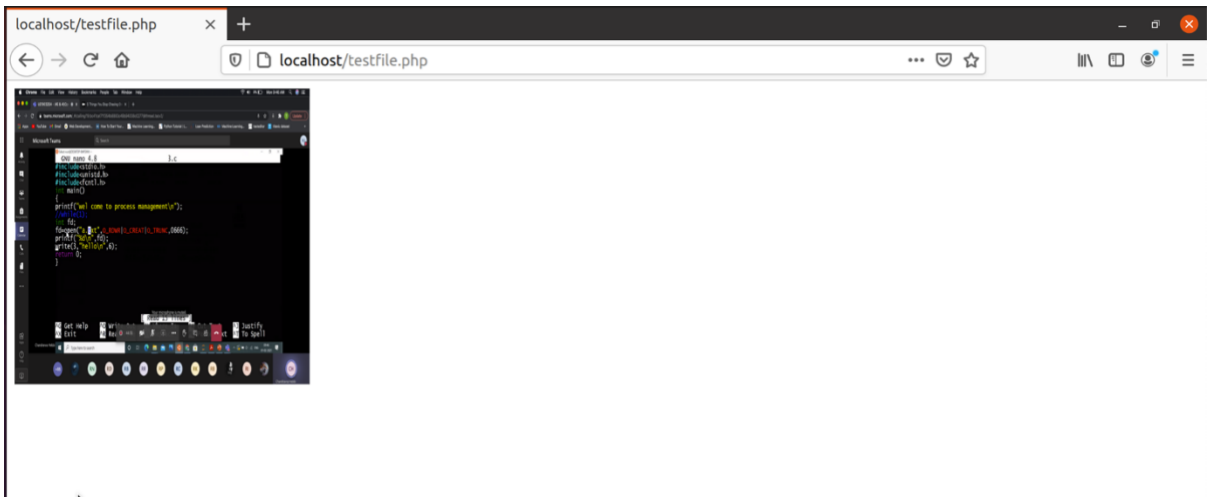
II. Cookie Setting

We set cookies using a PHP file and the setcookie command as follows :-



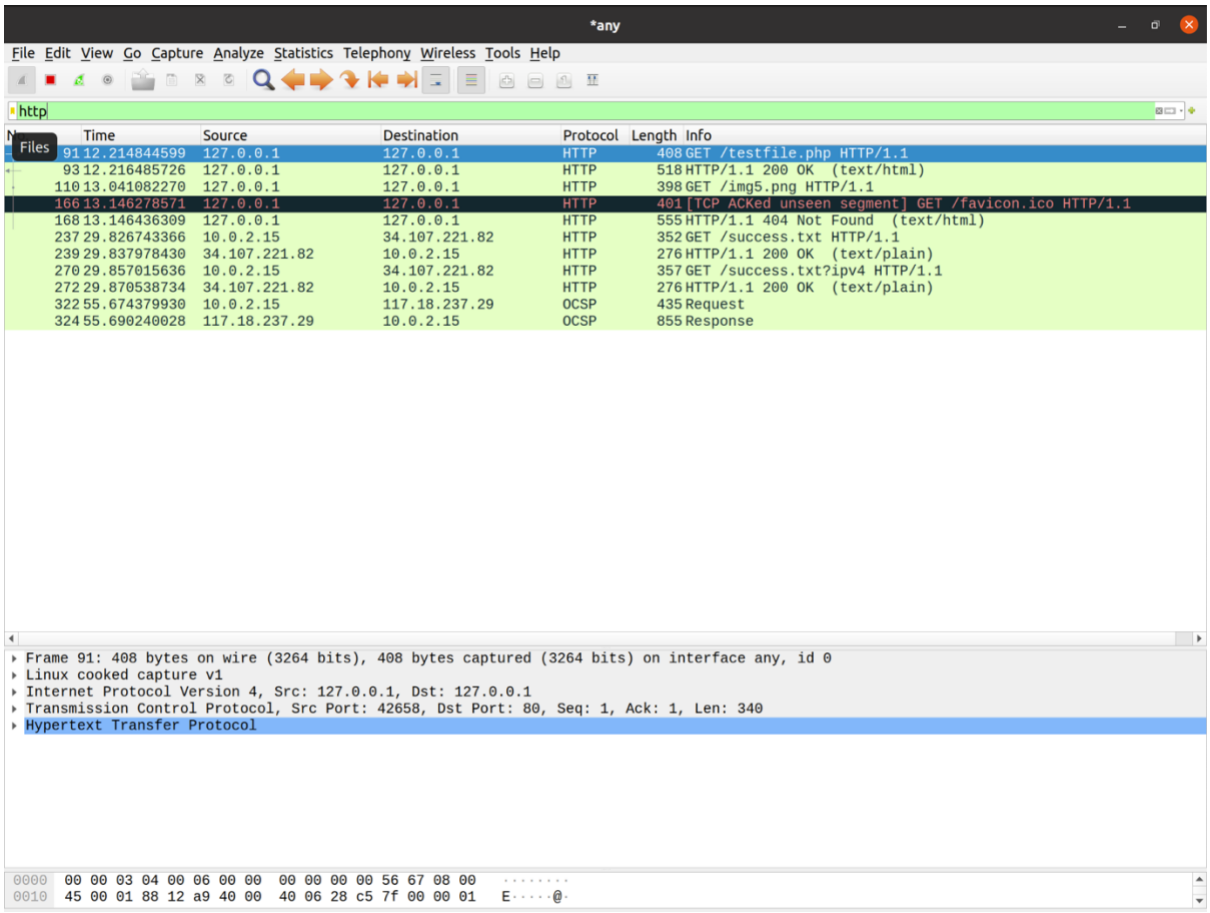
sudo nano testfile.php

Accessing the page from Firefox :-



localhost/testfile.php

Capturing the packets using Wireshark :-



In the TCP capture of the wireshark packet highlighted above, we notice that there are two additional fields in the HTTP response section named setcookie. This demonstrates that the cookie was set successfully.

TCP Capture :-

Wireshark · Follow TCP Stream (tcp.stream eq 2) · any

```
GET /testfile.php HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:84.0) Gecko/20100101 Firefox/84.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1

HTTP/1.1 200 OK
Date: Wed, 10 Feb 2021 16:21:35 GMT
Server: Apache/2.4.41 (Ubuntu)
Set-Cookie: namecookie=netquery; expires=Wed, 10-Feb-2021 16:23:38 GMT; Max-Age=123
Set-Cookie: nickname=work
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 87
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

............(.......MW(.J.U.2L.
.....3SJ2l..
...2R3.3J......T[.........%}.~}.9.h.X.O...GET /img5.png HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:84.0) Gecko/20100101 Firefox/84.0
Accept: image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://localhost/testfile.php
Cookie: namecookie=netquery; nickname=work

HTTP/1.1 200 OK
Date: Wed, 10 Feb 2021 16:21:36 GMT
Server: Apache/2.4.41 (Ubuntu)
Last-Modified: Mon, 01 Feb 2021 15:25:37 GMT
ETag: "165ad2-5ba47f73f23d7"
Accept-Ranges: bytes
Content-Length: 1465042
Keep-Alive: timeout=5, max=99
Connection: Keep-Alive
Content-Type: image/png

.PNG
```

3 client pkts, 31 server pkts, 5 turns.

Entire conversation (918kB) | Show data as ASCII | Stream 2

Find: | Find Next

Help | Filter Out This Stream | Print | Save as… | Back | Close

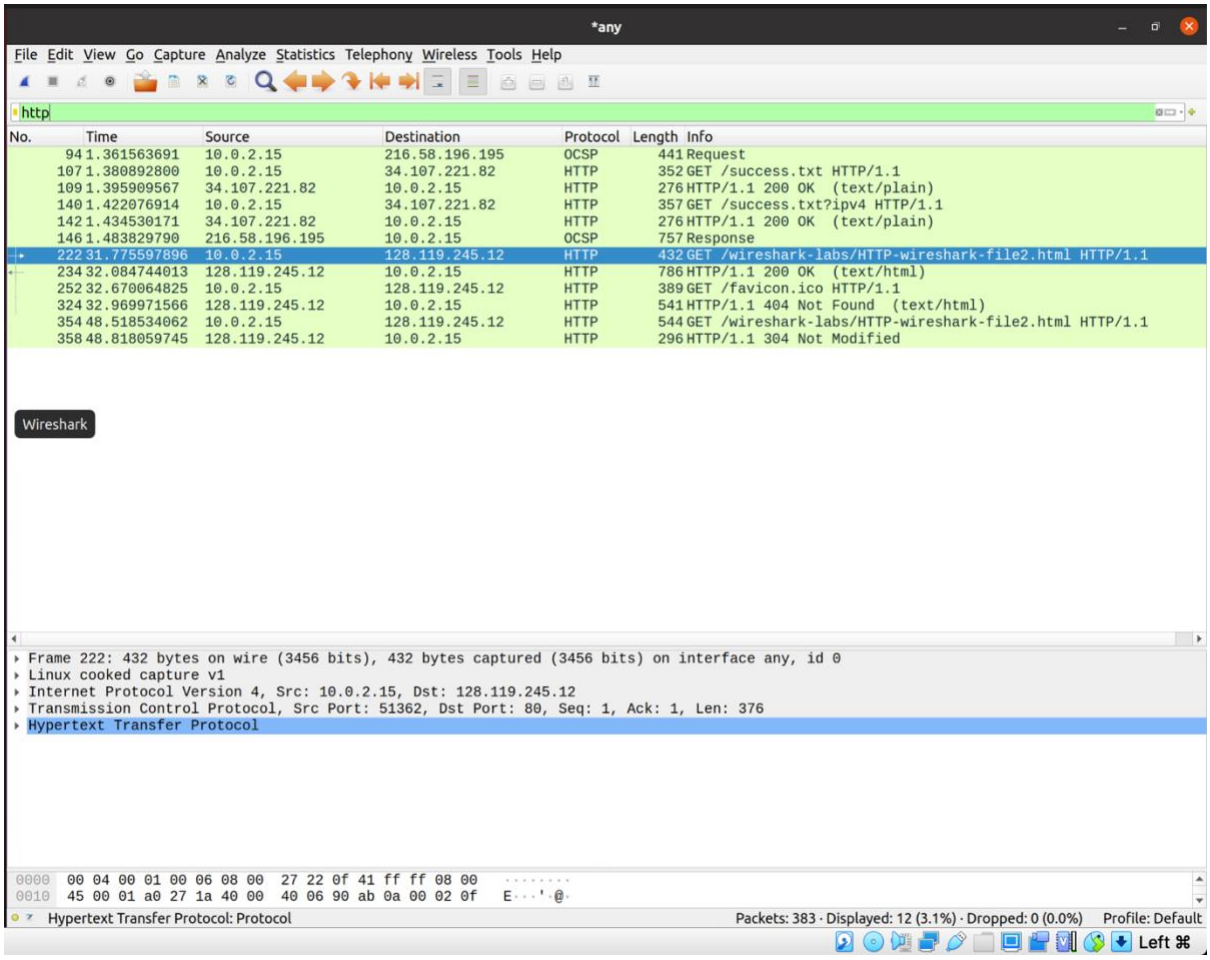
III. Conditional get

Conditional GET response can be implemented by using the **If-Modified_Since** which is checked by the server and then the resource that is requested is resent only if the resource has been modified since the timestamp in the header.

304 Not Modified status code is sent back if the resource has not been modidfied.

This is called conditional HTTP response because it only resends the resource if it has been modified since the last GET request by the client.

Accessing the HTML page : **http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html** for the first time :-



Wireshark · Follow TCP Stream (tcp.stream eq 8) · any

```
GET /wireshark-labs/HTTP-wireshark-file2.html HTTP/1.1
Host: gaia.cs.umass.edu
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:84.0) Gecko/20100101 Firefox/84.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1

HTTP/1.1 200 OK
Date: Mon, 08 Feb 2021 12:18:25 GMT
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.14 mod_perl/2.0.11 Perl/v5.16.3
Last-Modified: Mon, 08 Feb 2021 06:59:01 GMT
ETag: "173-5bacdb4665369"
Accept-Ranges: bytes
Content-Length: 371
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8


<html>

Congratulations again!  Now you've downloaded the file lab2-2.html. <br>
This file's last modification date will not change.  <p>
Thus  if you download this multiple times on your browser, a complete copy <br>
will only be sent once by the server due to the inclusion of the IN-MODIFIED-SINCE<br>
field in your browser's HTTP GET request to the server.

</html>
```

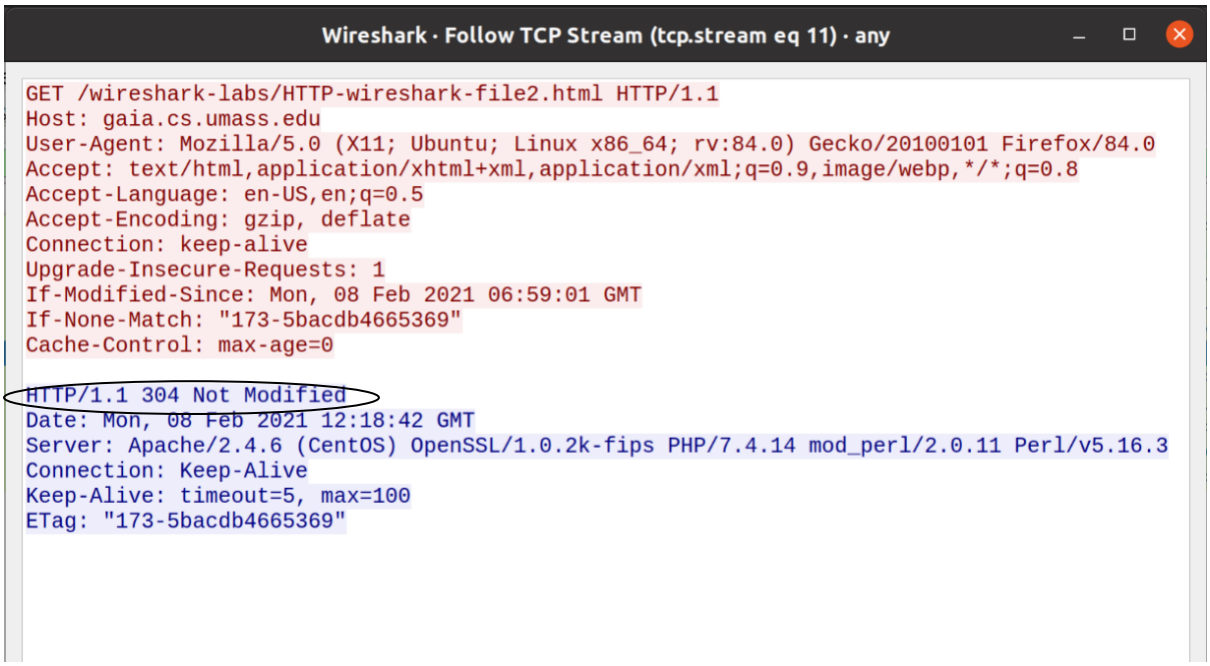
1 client pkt, 1 server pkt, 1 turn.

Entire conversation (1,106 bytes) | Show data as ASCII | Stream 8

Find: | Find Next

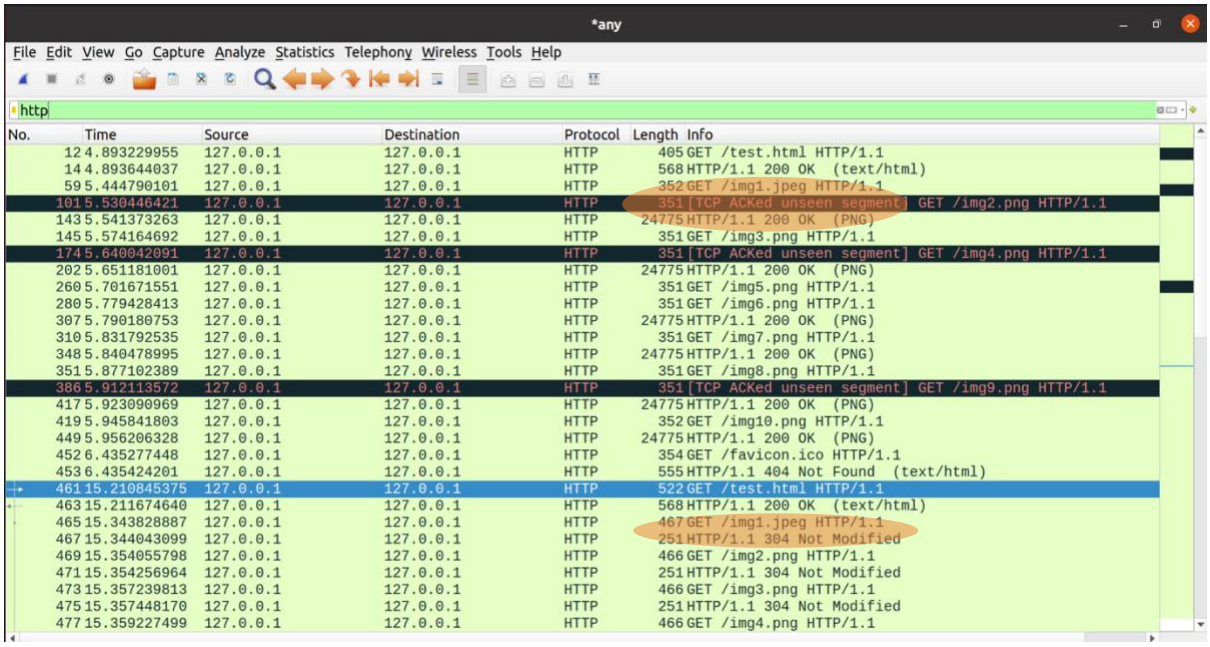
Help | Filter Out This Stream | Print | Save as… | Back | Close

Accessing the HTML page : **http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html** for the second time :-



<u>Accessing a local host file :-</u> An HTML file with ten images was created and saved in the localhost home directory. The screenshots below demonstrate the conditional get requests and responses.

Wireshark packet capture =>



Wireshark · Follow TCP Stream (tcp.stream eq 3) · any

```
GET /test.html HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:84.0) Gecko/20100101 Firefox/84.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1
If-Modified-Since: Tue, 02 Feb 2021 05:14:36 GMT
If-None-Match: "16b-5ba538bea9cc5-gzip"
Cache-Control: max-age=0

HTTP/1.1 200 OK
Date: Mon, 08 Feb 2021 12:26:20 GMT
Server: Apache/2.4.41 (Ubuntu)
Last-Modified: Tue, 02 Feb 2021 05:14:36 GMT
ETag: "16b-5ba538bea9cc5-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 163
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html

..........u.=..0...=.0=@C.G
Y.......I..R.,.=.ZX....=/...q[.O{p..F...K.F.......!..<...
-...Y...2P8
. :.....~
...Bx.`.YZU.h.f.'..M.......3.`|....5.j.
......
....k...GET /img1.jpeg HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:84.0) Gecko/20100101 Firefox/84.0
Accept: image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://localhost/test.html
If-Modified-Since: Mon, 01 Feb 2021 15:25:37 GMT
If-None-Match: "165ad2-5ba47f73f23d7"
Cache-Control: max-age=0

HTTP/1.1 304 Not Modified
Date: Mon, 08 Feb 2021 12:26:21 GMT
```

11 client pkts, 11 server pkts, 21 turns.

Entire conversation (6,766 bytes) | Show data as ASCII | Stream 3

Find: | Find Next

Help | Filter Out This Stream | Print | Save as… | Back | Close

It can be observed that when the file is reloaded the images are not resent by the HTTP client. This is because the files were not modified since the last time that we requested them. (img1.jpeg is highlighted in the screenshots as an example)