Lab: 3
Understanding Working of HTTP Headers

Question: Understand working of HTTP headers Conditional Get: If-Modified-Since

HTTP Cookies: Cookie and Set-Cookie

Authentication: Auth-Basic

Design a web page that has one embedded page (e.g. image) and sets a cookie and enables authentication. You are required to configure the web server (e.g. apache) with authentication mechanism. Show the behavior of conditional get when embedded objects are modified and when it is not (you can just change the create date of the embedded object). Decode the BasicAuth header using Base64 mechanism as per the password setup.

Observation: Show the behavior of browser when cookie is set and when cookie is removed.
Solution: Analyzing Basic Authentication and Cookies

The three parts of experiment are: 1. Password Authentication 2. Cookie Setting 3. Conditional get

Steps of Execution (for Password Authentication) 1. Executing the below commands on the terminal.

--> Provide username and password to set authentication

sudo htpasswd -c /etc/apache2/.htpasswd netwo

--> View the authentication

sudo cat /etc/apache2/.htpasswd

Here "netwo" is the username. Also, password is entered twice.

2. To setup the authentication phase, execute the following commands. Configuring Access control within the Virtual Host Definition.

--> Opening the file for setting authentication

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

<VirtualHost*:80>
ServerAdmin webmaster@localhost
DocumentRoot /var/www/html
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined <Directory "/var/www/html">

AuthType Basic
AuthName "RESTRICTED" AuthUserFile /etc/apache2/.htpasswd Require valid-user >

</Directory> </VirtualHost>

```
<VirtualHost *:80>
        # The ServerName directive sets the request scheme, hostname and port that
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        #ServerName www.example.com

        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined
        <Directory "/var/www/html">
                Authtype Basic
                AuthName "RESTRICTED"
                AuthUserFile /etc/apache2/.htpasswd
                Require valid-user
        </Directory>
        # For most configuration files from conf-available/, which are
        # enabled or disabled at a global level, it is possible to
        # include a line for only one particular virtual host. For example the
        # following line enables the CGI configuration for this host only
        # after it has been globally disabled with "a2disconf".
        #Include conf-available/serve-cgi-bin.conf
</VirtualHost>
```

3. Password policy implementation is done by restarting the server as:

sudo service apache2 restart

4. The localhost is then accessed using the Firefox browser requiring a username and a password set during the authentication phase.

5. Wireshark is used to capture the packets sent upon the network.

6. Using the "follow TCP stream" on the HTTP message segment the password was retrieved which was encrypted by the base64 algorithm and decryption could be done with same algorithm.

Steps of Execution (Cookie Setting)

1. A PHP file to set the cookie is created which also contains an image in it (placed under the HTML directory) to be accessed once the cookie is set. The following code helped to set the cookie:

<html> <?php

setcookie("namecookie","netqwerty",time()+123); setcookie("nickname","work"); ?>

 </html>

Note: The Image added is called **1.jpeg**
2. The combined file saved with a .php extension is placed under **/var/www/html** for accessing.

3. The packets are captured using Wireshark and using the "follow TCP stream" which checks for the set-cookie field whether the cookie is set or not set.

The cookie is set as shown in the above screenshot.

Observation: Understand and work out base 64 algorithm and write in your observation. Observe various parameters associated with Cookie in the wireshark capture. **Conditional Get: If-Modified-Since Observations:**

1. Inspect the contents of the first HTTP GET request from your browser to the server. Do you see an "IF-MODIFIED-SINCE" line in the HTTP GET?
2. Inspect the contents of the server response. Did the server explicitly return the contents of the file? How can you tell?
3. Now inspect the contents of the second HTTP GET request from your browser to the server. Do you see an "IF-MODIFIED-SINCE:" line in the HTTP GET? If so, what information follows the "IF-MODIFIED-SINCE:" header?
4. What is the HTTP status code and phrase returned from the server in response to this second HTTP GET? Did the server explicitly return the contents of the file? Explain. Wireshark containing the HTTP REQUESTS AND RESPONSES

1. IF-MODIFIED-SINCE is not present in the first HTTP GET

2. The server explicitly returned the contents on the first HTTP RESPONSE
Line-Based text data is the content returned by the **server** on **first HTTP RESPONSE**

3. Yes. IF-MODIFIED-SINCE is present in the second HTTP GET

4. The HTTP status code for the second HTTP GET is **304.**
The Server does not return the file as they are already present from the time the first http get request was made.This time the contents are not modified.

For a server with images (used the website www.zoomify.com)

1. IF-MODIFIED-SINCE is not present in the first HTTP GET

2. Theserverreturnsthefileexplicitly.

3. Yes. IF-MODIFIED-SINCE is present in the second HTTP GET

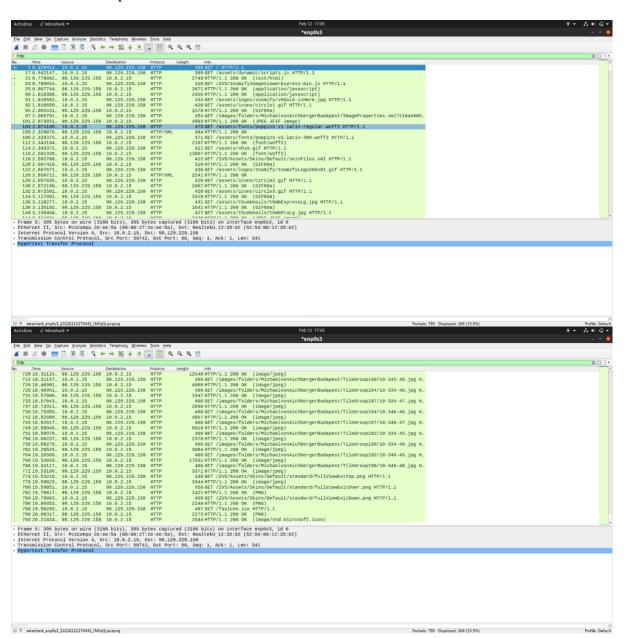4. The HTTP status code for the second HTTP GET is **304.**
The Server does not return the file as they are already present from the time the first http get request was made.
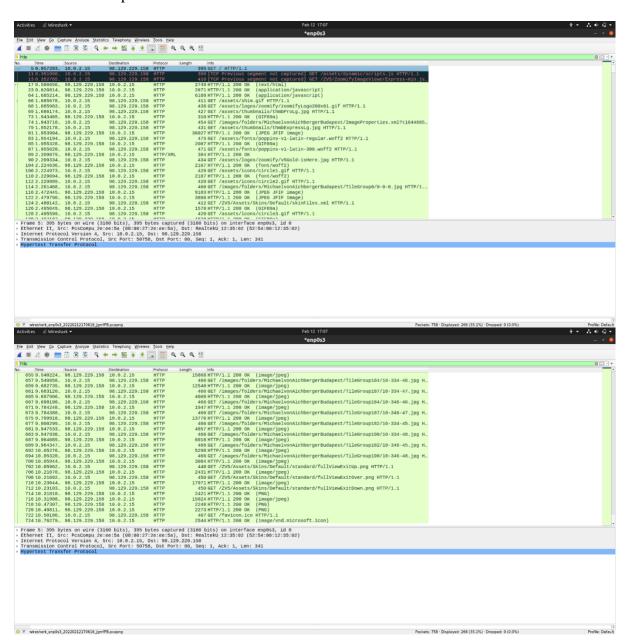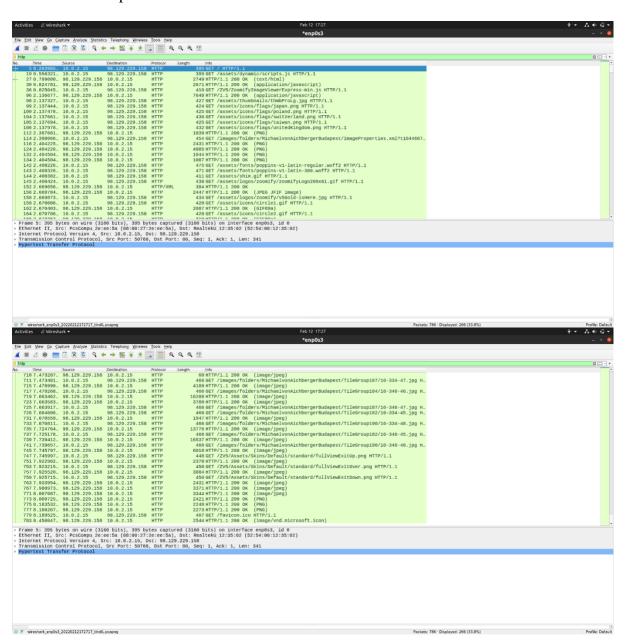
Persistent Non-persistent Connections
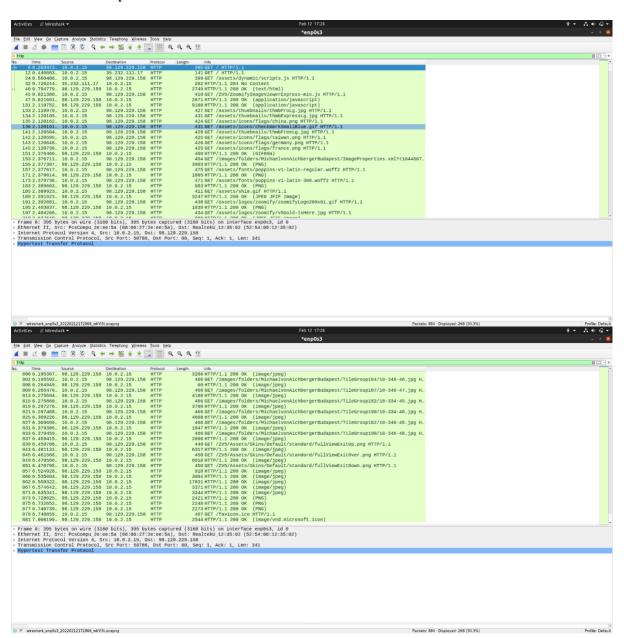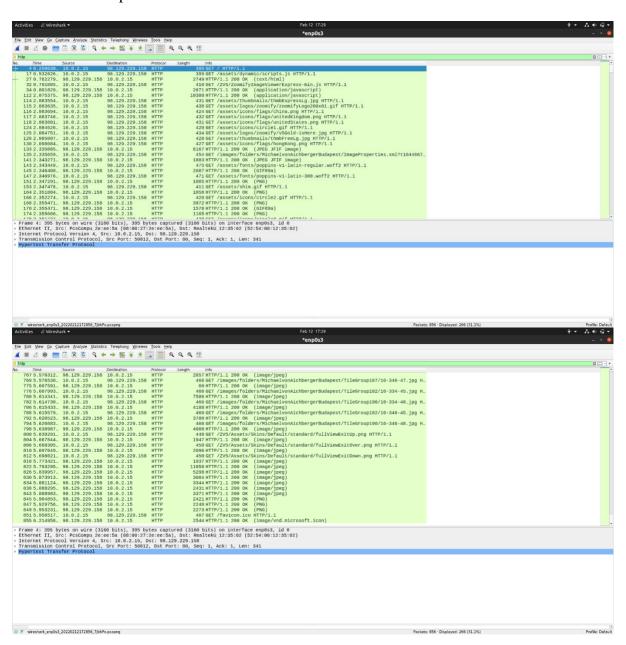
Non- Persistent

38.2236-0.2669=37.9567

Max Number of persistent Connections 2

20.238-0.2594=19.9786

Max Number of persistent Connections 4

10.762-0.057=10.705

Max Number of persistent Connections 6

8.45-0.263=8.187

Max Number of persistent Connections 8

7.006-0.263=6.743

Max Number of persistent Connections 10

6.214-0.250=5.964