**Lab - 02**

Name: Raphael Obu | Class: Comp120 Web Programming & Engineering

Release Date: 05/26/2021 | Due Date: 06/02/2021

**LAB DETAILS**

**Lab Format;**

The lab is provided as a google survey form link.

**Lab Task;**

The task is to complete the google survey form using all instructions available within the form.

**Lab Objectives;**

* Practice using Developer Tools in Google Chrome or Firefox.
* Understand how HTTP and URIs work.
* Dissect HTTP responses and request headers.
* Learn how to use curl.

**Lab Overview;**

Developer Tools in Google Chrome or Firefox "allows web developers and programmers deep access into the internals of the browser and their web application." Throughout this course, you will be using Developer Tools heavily, including debugging and finding bugs, inspecting assets and resources for a web page or web application, and measuring performance. In this lab, you will practice using Developer Tools to understand how HTTP works.

**LAB BODY**

**Instructions;**

Please clear all browsing history and cookies first. In Google Chrome or Firefox, open a new browser window and open Developer Tools.

For questions 1 to 6 (inclusive), visit <http://www.cs.tufts.edu/comp/120/http_lab.html> . Cut-and-paste the URL into the URL bar in the browser window with Developer Tools opened, enabled. Then press the return key on your keyboard. Click on the "Network" panel in Developer Tools. Answer the questions below:

**Questions;**

1. How many requests were made for the page? (For page <http://www.cs.tufts.edu/comp/120/http_lab.html>.)
2. How many kilobytes of data were transferred in total?
3. What is the status code for each of the requests? Are there any HTTP errors?
4. In the "Network" tab in Developer Tools, click on the thisisfine.jpg image on the left and view the HTTP response header details under the "Headers" tab. What is the web server software, and version, used by the Tufts Department of Computer Science? (This information can be found in the HTTP \*response\* header)
5. What was a significant amount of time being spent on prior to receiving the images? HINT: see the time line (NOTE: There is a finer point to this question, to be discussed more in this course.)
6. Reload the page on your web browser. What is the status code for each of the requests? If the status code is different than the one in question 3, explain why.
7. How many requests were made for the page? (For the home page of WIRED, <https://www.wired.com/>)
8. How many megabytes of data were transferred in total? (For the home page of WIRED, <https://www.wired.com/>)
9. Visit https://old.reddit.com/. Cut-and-paste the URL into the URL bar in the window with Developer Tools enabled. Then press the return key on your keyboard. Click on the "Network" panel of Developer Tools. DON'T DO A THING, STAY ON THE PAGE (for a minute or two). Notice the number of requests. What content or file that is constantly being retrieved? What is the name of the content or file?
10. Go to the page http://www.cs.tufts.edu/comp/120/hackme.php. You will see the message "...But sorry, no phrase that pays for you." Your job is to reveal a "phrase that pays." To do this, you will need to modify the URL, add a \*query string\*. What is the phrase that pays? (BIG HINTS: the necessary key in the query string is "token" (without the double quotes). You may need to submit a few times too!)

**About curl;**

curl is an open source command line tool and library for transferring data with URL syntax, supporting DICT, FILE, FTP, FTPS, Gopher, HTTP, HTTPS, IMAP, IMAPS, LDAP, LDAPS, POP3, POP3S, RTMP, RTSP, SCP, SFTP, SMB, SMTP, SMTPS, Telnet and TFTP. curl supports SSL certificates, HTTP POST, HTTP PUT, FTP uploading, HTTP form based upload, proxies, HTTP/2, cookies, user+password authentication (Basic, Plain, Digest, CRAM-MD5, NTLM, Negotiate and Kerberos), file transfer resume, proxy tunneling and more.

Please run `man curl` for the manual page on the curl tool. Examples on using curl can be found at <https://curl.haxx.se/docs/httpscripting.html> , <http://www.thegeekstuff.com/2012/04/curl-examples/>

**Questions Continued;**

1. Go to the page https://hls.ted.com/satisfaction using curl (no flags necessary). What is the response? From this, what is the default HTTP command (or verb) used when no flags are specified in curl? (HTTP joke of the day)
2. Run `curl -I https://www.tufts.edu/` (that flag is a capital letter "i"). What is the output? What is the purpose of the "-I" flag in curl?

**To Ponder;**

For your intellectual curiosity, use Developer Tools on Gmail or on Facebook to see what's going on behind the scenes.

**LAB WORK**

**Responses;**

1. 5 Requests
2. 60.4kB of data
3. Status codes for each of the requests are;
   1. Name: http\_lab.html Status: 200 OK
   2. Name: manchester.jpg Status: 403 Forbidden
   3. Name: college.jpg Status: 404 Not Found
   4. Name: thisisfine.jpg Status: 200 OK
   5. Name: secret.jpg Status: 200 OK
4. Apache/2.4.6 (Red Hat Enterprise Linux) OpenSSL/1.0.2k-fips Phusion\_Passenger/5.3.4 PHP/5.4.16 mod\_wsgi/3.4 Python/2.7.5
5. Prior to receiving the images a significant amount of time was spent loading the http\_lab.html file.
6. After reloading the status codes are now the following;
   1. Name: http\_lab.html Status: 304 Not Modified
   2. Name: manchester.jpg Status: 403 Forbidden
   3. Name: college.jpg Status: 404 Not Found
   4. Name: thisisfine.jpg Status: 200 OK
   5. Name: secret.jpg Status: 200 OK

The only code to have changed is the http\_lab.html one that has changed from ‘200 OK’ to ‘304 Not Modified’ which makes sense as nothing has changed.

1. 613 requests were made (Over time this would periodically increases in waves of about 50)
2. 6.7MB of data were transferred (Over time this would periodically increase in waves however)
3. It appears a gif named i.gif?z=gAAAAABgr… is constantly loading.
4. It was the best of times, it was the worst of times.
5. Output html file contents to terminal.
6. The output is information about the page. The purpose of the -I flag is to ‘show document info only’.

**Notes;**

* Must be connected to Tufts network, either by being at Tufts or using the VPN to access the Tufts network. This also happens to be useful for accessing other services that are only available in the US!

**REFERENCES**

* <https://developer.chrome.com/docs/devtools/>
* <https://developer.mozilla.org/en-US/docs/Tools/Network_Monitor>
* <https://curl.se/>