

COMP 416

PROJECT 1

AHMET CAN TURGUT | 0040616

CAN TAYLAN ÇAPRAZ | 0036493

Instructor: Öznor Özkasap

DUE DATE: 26/10/2016

1. Purpose of the Project:

For this project, it was designed a proxy server. Proxy server gets http request from user and if there is no banned word, it forward this request to main server.

This program gives 3 option at the beginning. If you press Y, proxy server is run. If you press N, proxy server is closed. If you press A, program lists all banned words and ask you a new banned word if user wants to add.

2. Classes and Methods:

In this project, 3 class were used.

SafeInternet Class:

This method get port number as an argument. Then it asks if user wants to close program, run proxy server or manage banned words.

Y mode: If the user choice to run proxy server(Y), ProxyServer class is called. Also, thread starts, which makes user able to connect more than one website.

A Mode: If the use choice to manage banned words(A), program lists all banned words and ask user if she wants to add new one to database.txt file. It ask user a new word to ban after user press d.

N Mode: If the user choice to close program(N), program is closed.

ProxyServer Class:

This class creates a socket using the input port number. If there is no input port number, it uses 8080 as a default. After creating socket, it runs in a thread. It listens this socket until the program is closed.

ProxyThread Class:

This class is the most functional class in this project. It has **getHttp**, **readLine**, **forwardHttp** and **filter** methods. It uses these methods to get http request as an input from Client, reading this input, filtering the URL request read and streaming this request if it is not banned.

This class also keep the fallowing data in logfile.txt file:

- 1) Date
- 2) User IP
- 3) requested URL
- 4) delay if website is not banned
- 5) banned word if website is banned

Method 1: getHttp()

It gets the request data came from client, a StringBuffer and a type. It creates a ByteArrayOutputStream object and call forwardHttp to write Datas read on Client's input.

Method 2: readLine()

It gets the request data comes from the client as an input. It just read a line of Client's input and write it to StringBuffer. At the end, It transform StringBuffer to String and return this string.

Method 3: forwardHttp()

It gets the request data came from client, a StringBuffer and type and an ByteArrayOutputStream created by getHttp().

It reads all lines and look for http request, response code, host and content length. It saves this data to other variables defined before. Also, it writes all lines it read to ByteArrayOutputStream.

Method 4: Filter()

This method gets a string as an input. It opens the file database.txt to get list of banned words. Then it searches if there is a banned word in its input. If there is, it saves it in a string defined before to be able to print it in other methods. Then it returns false. Otherwise, it returns true.

RUN TIME:

Run method gets some data such as request, port number, host name, IP address, start time, end time, etc. Then, it creates a new socket to connect internet. After that, it filters the host name. If there is a banned word, it sent a message to browser using BufferedOutputStream which is created on the socket. If there is no, it connects to server. It sends request and get Response. If there is no banned word, it sends this response to client. Then, it modifies logFile.txt document and print what it does to command prompt.

3. Problems about Code

This code was written to work if and only if http request is done. However, some websites use https such as google, ku.edu.tr, Wikipedia, facebook etc. Thus, it is not possible to connect this sites if computer connected to this program. Reason of that this web pages require HTTPS. If any web page use HTTPS, with SafeInternet program user unable to connect that kind of web pages.

In that case, if any web page use HTTPS, not openable with SafeInternet proxy server.

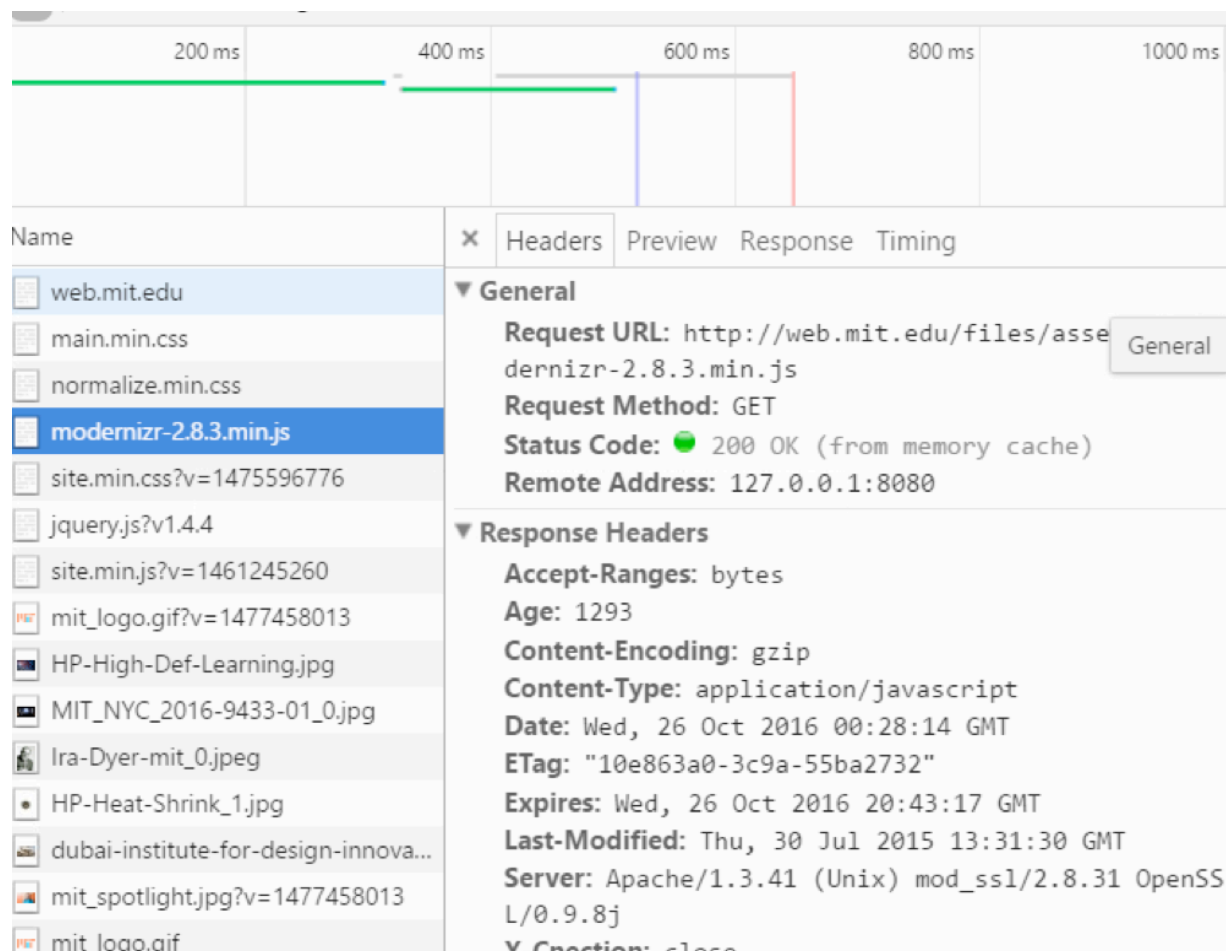
In addition, this program transfers data as a byte. It can transfer some data such as pictures and texts properly. But, for example, it may not be able to transfer javascript or video. Thus, some websites cannot be seen as it should be.

Uncorrected Observations

Some webpages load completely after applying interruption on the running program (ctrl + c).

4. Results and Conclusion:

This SafeInternet program was tasted using Chrome, Firefox and Edge. It was observed that it works properly for all http website. There are only two problems which was explained in the part “3) Problems about Code”. If the website does not contain any banned word, proxy server stream the URL request and response. If it contains any banned word, “This Webpage Is Forbidden To Access” error is written to browser. All requests are saved to logfile.txt.



The screenshot shows a web browser's Network tab. The top bar includes tabs for Elements, Console, Sources, Network, Timeline, and Profiles. The Network tab is active, displaying a list of requests. The 'analytics.js' request is selected, and its details are shown on the right. The details include the Request URL, Request Headers, and a warning about provisional headers.

Filter: ☐ Regex ☐ Hide data URLs

All | XHR | JS | CSS | Img | Media | Font | Doc | WS | Manifest | Other

Name

- web.mit.edu
- main.min.css
- normalize.min.css
- site.min.css?v=1475596776
- modernizr-2.8.3.min.js
- jquery.js?v1.4.4
- site.min.js?v=1461245260
- mit_logo.gif?v=1477458013
- HP-High-Def-Learning.jpg
- MIT_NYC_2016-9433-01_0.jpg
- Ira-Dyer-mit_0.jpeg
- HP-Heat-Shrink_1.jpg
- dubai-institute-for-design-innova...
- mit_spotlight.jpg?v=1477458013
- mit_logo.gif
- analytics.js**
- logo-mit-print.gif
- icon-search-hd.png
- bg_transparent.png
- icon-arrow-go-hd.png
- sprites-social-hd.png
- sprites-social-print.png
- icon-arrow-down-hd.png

23 requests | 378 B transferred | ...

Request URL: <https://www.google-analytics.com/analytics.js>

Request Headers

Provisional headers are shown

Referer: <http://web.mit.edu/>

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/54.0.2840.71 Safari/537.36

Session list

- Cans-MacBook-Pro.l

Elements Console Sources Network Timeline Profiles 3

View: [Icons] [] Preserve log [] Disable cache [] Offline No throttling

Filter [] Regex [] Hide data URLs

All XHR JS CSS Img Media Font Doc WS Manifest Other

5000 ms 10000 ms 15000 ms 20000 ms 25000 ms 30000 ms 35000 ms

Name

www.google.com.tr

www.google.com.tr

Headers Preview Response Timing

General

Request URL: <https://www.google.com.tr/>

Request Headers

⚠ Provisional headers are shown

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/54.0.2840.71 Safari/537.36

2 requests | 0 B transferred

TeamViewer Free license (no use only)

Session list

Cans-MacBook-Pro.local

The screenshot shows the Network tab of a web browser's developer tools. The top toolbar includes icons for navigation, a filter input, and checkboxes for 'Preserve log', 'Disable cache', 'Offline', and 'No throttling'. The filter input is empty, and the 'All' filter is selected. Below the filter, a list of resource types (XHR, JS, CSS, Img, Media, Font, Doc, WS, Manifest, Other) is shown. A timeline at the top of the network list shows a request starting at 200 ms and ending at 400 ms. The request is highlighted in blue. The details pane on the right shows the following information:

| Name | Headers | Preview | Response | Cookies | Timing |
|--------------|--|---------|----------|---------|--------|
| www.bing.com | <p>General</p> <p>Request URL: http://www.bing.com/ Request Method: GET Status Code: 500 Remote Address: 127.0.0.1:8080</p> <p>Request Headers view source</p> <p>Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8 Accept-Encoding: gzip, deflate, sdch Accept-Language: tr-TR,tr;q=0.8,en-US;q=0.6,en;q=0.4 Cache-Control: no-cache Cookie: MUID=3D6F75CA72FC62470D9E7C7E76FC61A0; SRCHD=AF=NOFORM; SRCHUID=V=2&GUID=9F77F313BE0C4CCCAFDC283947024042; SRCHUSR=DOB=20161020; MUIDB=3D6F75CA72FC62470D9E7C7E76FC61A0; _ITAB=STAB=FAV; SRCHPGUSR=CW=1536&CH=759&DPR=1.25&UTC=180 Host: www.bing.com</p> | | | | |