CSE471: DATA COMMUNICATIONS AND COMPUTER NETWORKS

YEDITEPE UNIVERSITY

FALL 2021

TERM PROJECT – DUE DATE DECEMBER 29TH, 2021

As a term project you are expected to develop a full-featured HTTP Proxy. In basic terms, a proxy relays HTTP requests and responses back and forth between a client and a web server – see Figure 1. Using your prior experience developing an HTTP proxy, you should implement the following main capabilities into your project application:

i. Support for HEAD and POST HTTP methods
ii. HTTPS relaying via HTTP CONNECT method
iii. Local caching for appropriate resources

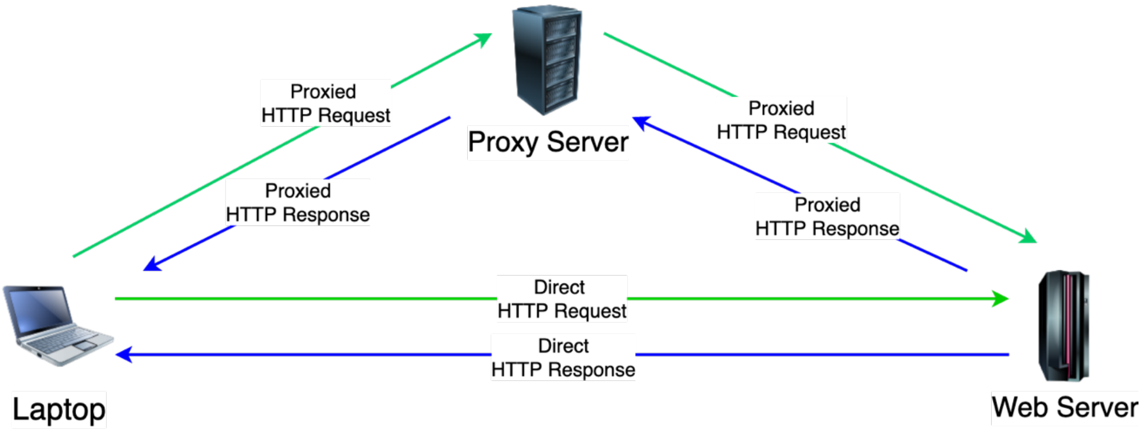


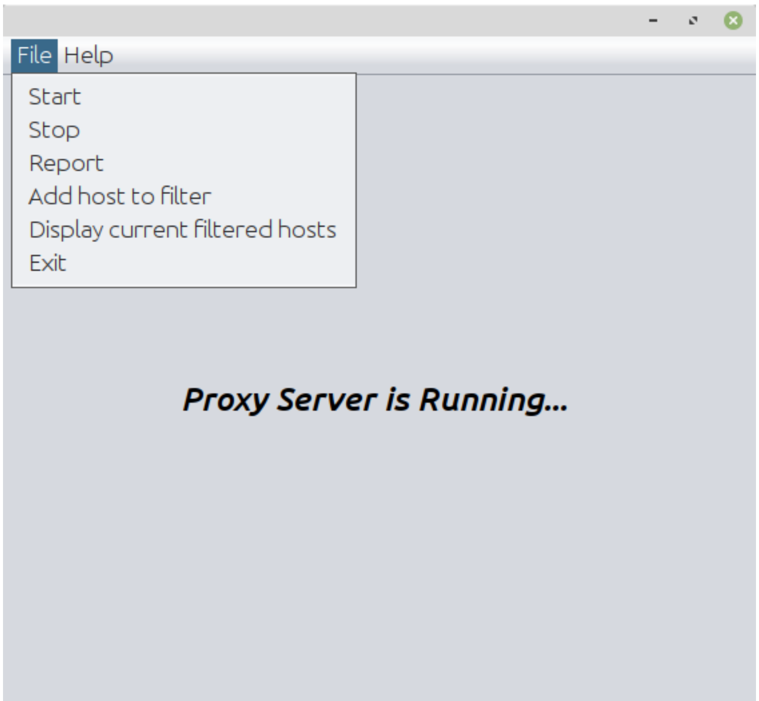
Figure 1: HTTP Proxy Basic Principle

Your HTTP Proxy application should satisfy the following requirements:

- For usability purposes, you are asked to implement a graphical user interface (GUI) for your HTTP Proxy – see Figure 2.
- To start the proxy application, you should select Start from the File menu (File->Start)
- To stop the proxy application, you should select Stop from the File menu (File->Stop)
- A client's history log reports should be accessible when Report from the File menu (File->Report) is selected. Selecting this menu item should trigger a pop-up menu, which contains a text box that the user can input a specific client's IP address. The report should be saved to a TXT file with each line

consisting of the date, client IP, requested domain, resource path (if visible), HTTP method (if visible) and the response status code (if visible).

- The proxy should have functionality where it denies relay if a domain/host is in a relay-ban list (filter list). The administrator should have the option to add hosts to the filter list by selecting “Add host to filter” in the File menu (File->Add host to filter). The filter list should be displayed on the screen by selecting “Display current filtered hosts” in the File menu (File->Display current filtered hosts).
- To exit the application, you should select Exit from the File menu (File->Exit).
- The Help menu should contain information about the developer of the application.



Proxy Server is Running...

Figure 2: HTTP Proxy Application Screenshot

From the application functionality point of view, your HTTP Proxy entail the following:

- The proxy should support the following HTTP methods; GET, HEAD, POST, and CONNECT.
- If the proxy encounters any other requested method from the client, it should return a *“405 Method Not Allowed”* response without contacting to the requested domain.
- If the proxy encounters a GET, HEAD, POST or CONNECT request with a filtered domain, it should return a *“401 Unauthorized”* response without contacting to the requested domain.

- HTTP proxy should support the relay of HTTPS connections between a client and a web server. This functionality can be achieved by integrating the HTTP CONNECT method. The details of an HTTP request using CONNECT method and its subsequent response can be examined using the following links:
    - https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods/CONNECT
    - https://en.wikipedia.org/wiki/HTTP_tunnel
- Your proxy server should be able to cache proper resources when clients access websites. The most common way to determine if a resource is cacheable is to check if the HTTP response from the server contains a "*Last-Modified*" field in the header. The proxy server should save these resources along with their URL addresses and last modification dates.

  When a client requests the same resource again in the future, the proxy server should query the web server with HTTP request containing *"If-Modified-Since"* header to check whether the cached resource is not expired.

  If the resource is not expired, then the proxy server can serve the resource back to the client using its cache. Otherwise, the proxy server should serve the updated resource pulled from the server back to the client while updating its cache to reflect the modified resource.
- [BONUS] Your HTTP Proxy acts also as SSH relay for clients behind corporate firewall – such as Yeditepe University's FortiGate Firewall.

Tip: Due to HTTPS upgrade policies of certain websites, some resources can automatically upgrade to HTTPS connections thus bypassing your proxy server. To prevent this, use Firefox browser and type *"about:config"* in the navigation bar. Look for *"network.stricttransportsecurity.preloadlist"* and *"browser.fixup.fallback-to-https"* fields and set them to false. You may need to delete your entire browsing history and restart Firefox so that your proxy server works as expected.

Submit your project source code in a zip file, which has your student number as name, using YULEARN by the end of Wednesday, December 29th, 2021. All submitted source files will be check for plagiarism among classmates and with any existing open source code available on the Internet. Furthermore, all students will be required to demonstrate their work for 15 minutes. DO NOT submit somebody else's work.