CS 421 – Computer Networks

Programming Assignment 1

*ProxyDownloader*

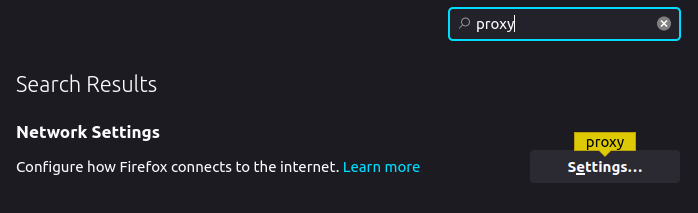
**Due: March 31, 2023 at 11:59PM**

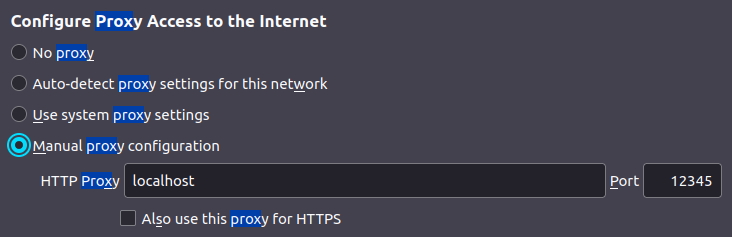
In recent years, Internet users have been increasingly preferring to route their traffic over third parties — whether to bypass geographical content restrictions applied by media streaming services, or to bypass firewall restrictions enforced by Internet Service Providers (ISPs), or to prevent privacy-violating user tracking practices of ad businesses — using standard technologies, such as [Virtual Private Networks](https://en.wikipedia.org/wiki/Virtual_private_network) (VPN) and HTTP proxies, or proprietary ones, such as [Apple’s iCloud Private Relay](https://support.apple.com/en-us/HT212614). This programming assignment is about HTTP proxies.

In this programming assignment, you are asked to implement a program in either Java or Python. This program is supposed to download a file using HTTP commands while getting the necessary information from your implemented Proxy server. For this programming assignment, you are not allowed to use any third-party HTTP client libraries or the HTTP-specific core or non-core APIs. The goal of this assignment is to make you familiar with the internals of the HTTP protocol and proxy servers, so using any (third party, core, or non-core) API providing any level of abstraction specific to the HTTP protocol is not allowed. You must implement your program using either the Java Socket API of the JDK or the socket module in the Python distribution. For Python, the use of any class/function from http or the requests modules is prohibited. If you have any doubts about what to use or not to use, please contact [**hamza.pehlivan@bilkent.edu.tr**](mailto:hamza.pehlivan@bilkent.edu.tr)and [**sarp.yenicesu@bilkent.edu.tr**](mailto:sarp.yenicesu@bilkent.edu.tr)**.**

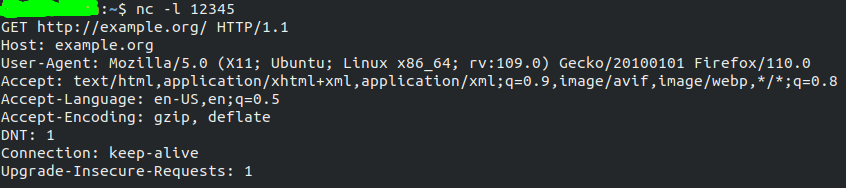
This assignment consists of two parts: Setting up Firefox for redirecting requests, and coding the proxy server.

1-) Firstly, download [Firefox Web browser](https://www.mozilla.org/en-US/firefox/new/) and change its proxy settings to manual proxy configuration with hostname as ‘localhost’ and port number to any free port. In this example, we use 12345. This process allows Firefox to redirect any HTTP request to our proxy server code.





Using Netcat networking tool, observe what information Firefox is sending. Use -l option to specify the port number. An example is given below when we try to connect at ‘example.org’, which is an [IANA-managed reserved domain name](https://www.iana.org/domains/reserved), with Firefox:



Include 3 HTTP website results using Netcat to your report. You can find many example non-HTTPS URLs from the following website: <https://whynohttps.com/>. Also notice that if you have visited a website with HTTPS before, Firefox may convert the default protocol to HTTPS, even if you try connecting with HTTP! HTTPS websites will not appear in Netcat output, because we have not set a proxy for HTTPS.

2-) As the second part of the assignment, you should code your proxy server, which will read requests coming from Firefox, just like Netcat does. You should also print the requests on the console, again, like Netcat.

Your program must be a console application (no graphical user interface (GUI) is required) and should be named as ProxyDownloader (i.e., the name of the class that includes the main method for Java should be ProxyDownloader). Your program should run with the

java ProxyDownloader <port>

or

python ProxyDownloader.py <port>

command, where <port> is a command line argument.

The details of the command-line arguments are as follows:

* <port>: [Required] The Proxy server port that your program will be listening to.

When a user runs the command above, your program will start listening to the designated port, and once you visit a website that uses HTTP from your browser, the web proxy server will get the necessary information. Using this information (like server name), you should send an HTTP GET request to download the file. If the file is not found, i.e., the response is a message other than 200 OK, your program should print an error message to the command line. If the file is found, i.e., the response is a 200 OK message, your program should continue and follow the URLs and download the complete file. Your program should print the status code in either case.

If your program successfully obtains the file or a part of the file, it saves the content under the directory in which your program runs. The file name should be parsed from given URL. Your program should continue to run as long as the user tries to access the websites. You can use the following URLs to test your implementation (our tests are not limited to these files): [www.cs.bilkent.edu.tr/~cs421/spring22/project1/bilkent.txt](http://www.cs.bilkent.edu.tr/~cs421/spring22/project1/bilkent.txt)

[www.cs.bilkent.edu.tr/~cs421/spring22/project1/decrypted\_file\_1.txt](http://www.cs.bilkent.edu.tr/~cs421/spring22/project1/decrypted_file_1.txt)

[www.cs.bilkent.edu.tr/~cs421/spring22/project1/nums.txt](http://www.cs.bilkent.edu.tr/~cs421/spring22/project1/nums.txt) (Handle 404 error)

[www.cs.bilkent.edu.tr/~cs421/spring22/project1/lorem.txt](http://www.cs.bilkent.edu.tr/~cs421/spring22/project1/lorem.txt) (Large file, around 4 MB)

Basically, you should do the following:

1. Adjust proxy settings of your Firefox web browser so that Firefox can direct requests to your code.
2. Parse the incoming request to extract various information such as server name and file name. You **must** print the request on the console.
3. Send an HTTP request to the server you found and retrieve its content. Beware that the file content can be large, which might require additional logic in your code. You **must** print the returned status code on the console.
4. Save the downloaded file with the correct name.
5. Repeat the same procedure starting from step 2, as long as the user keeps connecting to the websites.

As a **bonus assignment**, you can add additional features and mechanisms to ProxyDownloader. We will evaluate these features based on creativity and implementational complexity. This part's grading is separate from the original scheme and is not mandatory. The bonus points that you will get from this part will be added to the original grade.

Lastly, write a **report** (PDF file) in which you explain the important parts of your code, together with Netcat results. We should be able to navigate the source code just from the report. The number of pages should not exceed 5, otherwise, you may **lose** points.

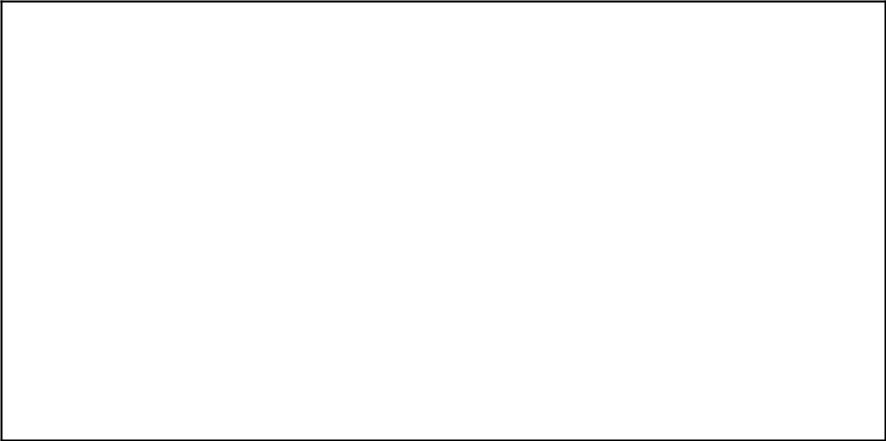
**Example**

Let us assume we set up Firefox to use port 12345 and we are trying to access [www.cs.bilkent.edu.tr/~cs421/spring22/project1/bilkent.txt](http://www.cs.bilkent.edu.tr/~cs421/spring22/project1/bilkent.txt) with Firefox.

**Example run**  Let your program start with the command:

java ProxyDownloader 12345 or

python ProxyDownloader.py 12345



Command-­‐line:

Retrieved request from Firefox:

GET http://www.cs.bilkent.edu.tr/~cs421/spring22/project1/bilkent.txt HTTP/1.1

Host: www.cs.bilkent.edu.tr

User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86\_64; rv:109.0) Gecko/20100101 Firefox/110.0

Accept:text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,\*/\*;q=0.8

Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate

DNT: 1

Connection: keep-alive

Upgrade-Insecure-Requests: 1

Downloading file ‘bilkent.txt’...

Retrieved: 200 OK

Saving file…

(Continue with next website)

Submission rules

You need to apply all the following rules in your submission. **You will lose points if you do**

**not obey the submission rules below or your program does not run as described in the**

**assignment above.**

* The assignment should be uploaded to the Moodle in a zip file. Any other methods, e.g., e-mail, flash disk, CD, DVD, for submission will not be accepted.
* The name of the submission should start with [CS421\_PA1], and include your name and student ID. For example, the name must be

[CS421\_PA1]AliVelioglu20111222

if your name and ID are Ali Velioglu and 20111222. **You need to do and submit this assignment individually. You cannot cooperate with anybody while doing the assignment.**

* All the files must be submitted in a **zip** file whose name is described above. The file must be a .zip file, not a .rar file or any other compressed file.
* All of the files must be in the root of the zip file; directory structures are not allowed. Please note that this also disallows organizing your code into Java packages. The archive should not contain:

– Any class files or other executables,

– Any third-party library archives (i.e. jar files),

– Any text files,

– Project files used by IDEs (e.g., JCreator, JBuilder, SunOne, Eclipse, Idea or NetBeans etc.). You may, and are encouraged to, use these programs while developing, but the result must be a clean, IDE-independent program.

* The standard rules for plagiarism and academic honesty apply; if in doubt refer to [Academic Integrity Guidelines for Students](https://w3.bilkent.edu.tr/web/provost/SAIC_Students.pdf) and [Academic Integrity, Plagiarism & Cheating](http://ascu.bilkent.edu.tr/Academic_Honesty.pdf).

**Important Notes**

**General Submission Rules:**

**1- )** Your submission should includesource code(s) (.java/.py).

**2- )** The format of the report should be **PDF**. (Do not upload .doc, .docx or any other types). The number of pages should not exceed 5.

**4- )** Your submission should not contain any other files other than the source code(s) (.java/.py), report (.pdf) and optional README. No .txt files, no folders, no IDE related files should be included.

**5-)** Compress these files with **.zip** format. (.rar, .7z or any other compressing types will not be accepted.)

**6-)** Make sure to follow rules in the “Submission Rules” section like name of the zip file, method of the submission etc.

**For Python Submissions:**

**1-)** Thecode should run with the “python3 ProxyDownloader.py <port>” command.

**2-)** Python version should be **3.6 or higher**. Other versions (like Python 2) are **not accepted**.

**For Java Submissions:**

**1-)** The code should run with the following commands:

Compile: “javac \*.java”

Run: “java ProxyDownloader <port>” command.

**2-)** Java version should be **8 or higher.**

**3-)** The JDK should be Oracle JDK (**not** OpenJDK).