

# MSCS-631: Python Lab 5: Web Proxy

Oishani Ganguly

GitHub Repository: [https://github.com/Oishani/MSCS631\\_PythonLab5](https://github.com/Oishani/MSCS631_PythonLab5)

## Lab 5: Web Proxy

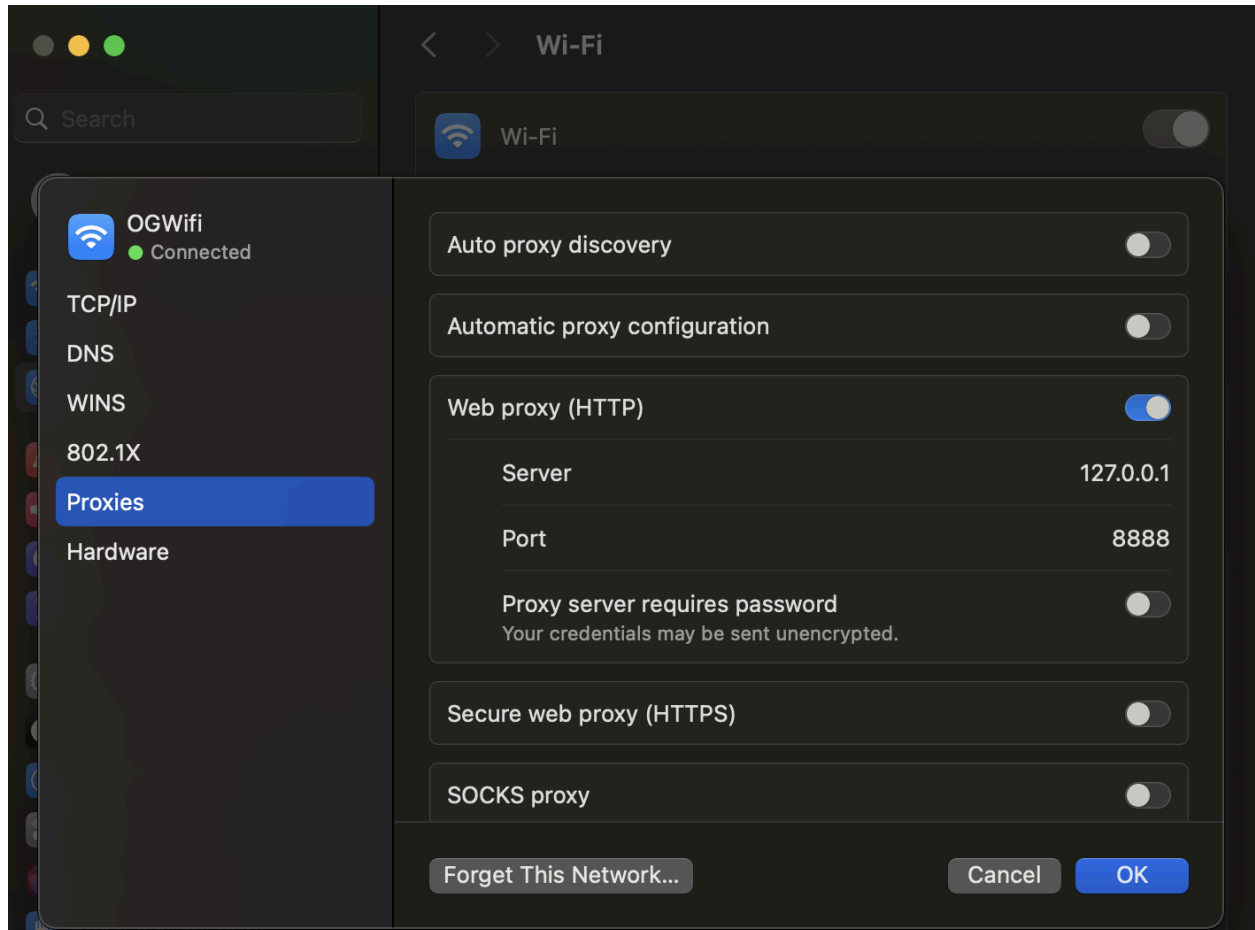
### Reflection on the Lab Experience

This lab provided valuable hands-on experience in building a basic HTTP proxy server and understanding the flow of web traffic between a browser, proxy, and web server. Setting up the proxy and configuring the browser to route traffic through it allowed me to observe real-time HTTP requests and responses, as well as see the benefits of local caching. It was exciting to see web pages load successfully through my own proxy server implementation and to verify the caching mechanism by observing terminal logs and cache files. This exercise reinforced my understanding of socket programming, HTTP protocol details, and the foundational concepts behind web proxies and content caching.

### Challenges Encountered

One of the primary challenges was adapting the skeleton code to handle the way modern browsers communicate with proxy servers. The original skeleton assumed a simplified request format, but browsers send the full URL in the HTTP GET line when using a proxy. As a result, I needed to adjust the code to reliably parse the full URL, extract the correct hostname and path, and generate safe cache file names. Additionally, I modified the file handling to use binary mode so that the proxy could correctly cache and serve not just HTML, but also images and other file types. These changes were necessary to ensure compatibility with real-world browser behavior and to fulfill the core requirements of the assignment. Without these adjustments, the proxy would have been unable to forward requests or cache responses correctly, especially for sites beyond basic text pages.

## Screenshots



Mac system network settings showing manual configuration of the HTTP web proxy to use 127.0.0.1 on port 8888. This ensures that all browser HTTP requests are routed through the local proxy server for testing and verification.

```
akshatphumbhra@Akshats-MacBook-Pro MSCS631_PythonLab5 % python3 ProxyServer.py 127.0.0.1  
Ready to serve...
```

Proxy server started successfully and is ready to accept client connections.

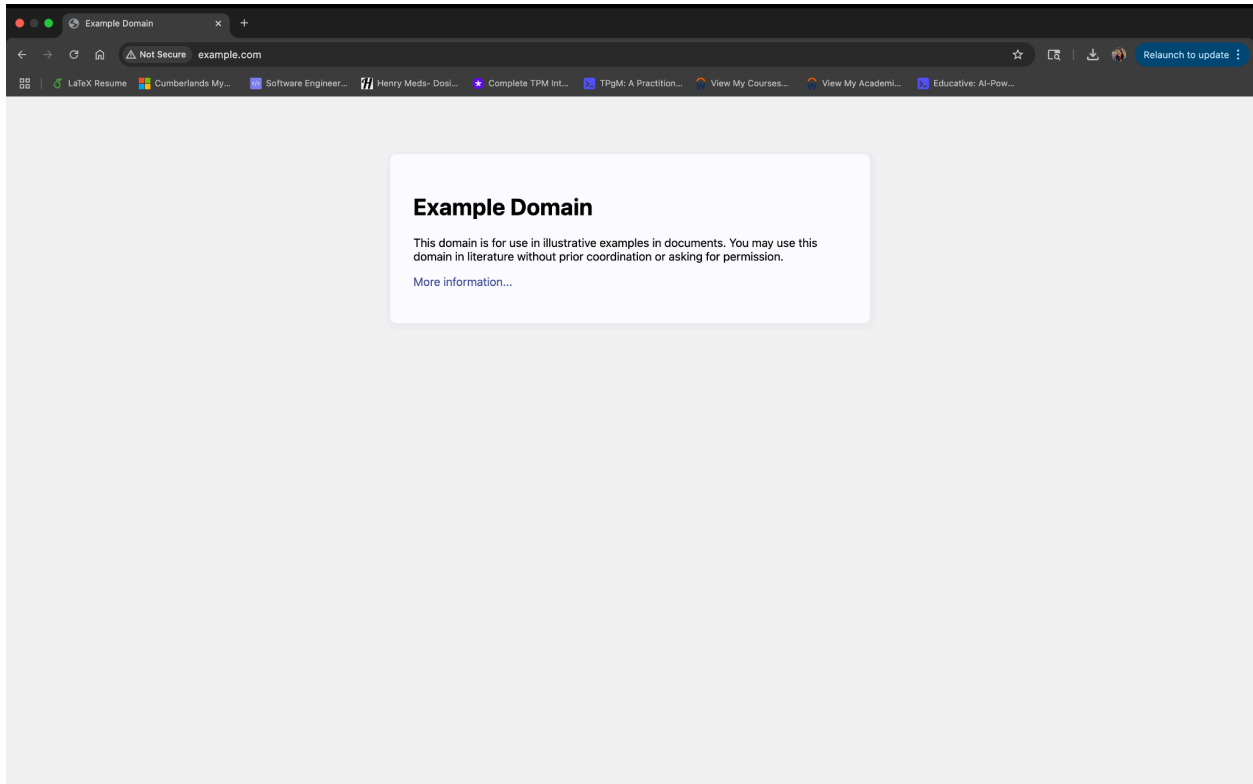
```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
Python + - [ ] [ ] ... - X

akshatphumbhra@Akshats-MacBook-Pro MSCS631_PythonLab5 % python3 ProxyServer.py 127.0.0.1
Ready to serve...
Received a connection from: ('127.0.0.1', 63995)
GET http://example.com/ HTTP/1.1
Host: example.com
Proxy-Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8

Parsed hostname: example.com
Parsed path: /
Cache filename: example.com_
Connecting to host: example.com
Ready to serve...
Received a connection from: ('127.0.0.1', 63996)
GET http://example.com/favicon.ico HTTP/1.1
Host: example.com
Proxy-Connection: keep-alive
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Referer: http://example.com/
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8

Parsed hostname: example.com
Parsed path: /favicon.ico
Cache filename: example.com_favicon.ico
Connecting to host: example.com
Ready to serve...
█
```

Terminal output showing the proxy server receiving and parsing HTTP requests from Chrome for [example.com](http://example.com), including both the main HTML page and favicon requests. The server logs show successful extraction of hostname and path, as well as attempts to connect to the target web server.



Chrome browser successfully loads "Example Domain" via the local proxy server, verifying that the proxy is correctly handling and forwarding HTTP requests.