Group 6
CS 433: Computer Networks
IIT Gandhinagar

Adit Kaushik, 21110010
Anish Karnik, 21110098
Ayush Modi, 21110039
Rutwik More, 21110133

Proxy Server Project Proposal

1. Nature of Work

We propose to develop a smart proxy server from scratch with the capability to efficiently handle various types of network traffic, provide advanced features, and adapt to changing network conditions. The smart proxy server will be designed to serve as both a forward and reverse proxy, enhancing network security, performance, and flexibility.

2. Summary of Intended Work

2.1. Initial Work

- Create a basic server framework that listens for incoming requests on a specified port.
- Accept and parse incoming client requests. Determine the target server to forward the request to. The proxy server should be able to handle HTTP and HTTPS requests. Forward client requests to the target server.
- Add a caching mechanism to store and serve frequently requested resources locally, reducing latency and load on backend servers.

2.2. Additional Features

- Implement rate limiting to prevent abuse of the proxy server and protect backend resources.
- Implement security measures to mitigate common threats.
- Implement an anonymous proxy.

3. Tools to be Used for the Project

- Programming Language: Python
- Postman
- Network Socket Programming Libraries: socket, asyncio
- SSL/TLS Handling: OpenSSL or Python’s SSL module
- Caching: cachetool library
- Version Control: Git

4. Roles of Individual Members

- Adit Kaushik: [Basic Server Framework, Handling HTTPS requests, parsing and forwarding client request]
- Anish Karnik: [Basic Server Framework, Handling HTTP requests, Caching Mechanism]
- Ayush Modi: [Basic Server Framework, Handling HTTP requests, Caching Mechanism]
- Rutwik More: [Basic Server Framework, Handling HTTPS requests, parsing and forwarding client request]

5. Timeline

- Week 0: Project planning and research.
- Week 1-2: Core proxy server development which listens for incoming requests on a specified port and parses incoming client requests.
- Week 3: Basic performance optimization and caching.
- Week 4: Minimal security implementation.
- Week 5: Final testing, bug fixing, and deployment.
- Week 6: Documentation and report writing.

6. Expected Outcome of the Project

The expected outcome of the project is the development of a smart proxy server with the initial capabilities mentioned in the summary. The server will be able to efficiently handle various types of network traffic, provide features like caching, and enhance network security, performance, and flexibility. Additional features and adaptability may be implemented as the project progresses beyond the initial work phase.