

# **CSCE 5580: Introduction to Computer Networks**

## **Multi-Threaded Web Proxy**

**Guided by,**

**Dr.Enkh-Amgalan Evan Baatarjav**

**Team Members,**

**Group-10**

Madhuri Ponnamm-11014324

Dheeraj Reddy Kanthala-11024584

Sai Spandana Muttavarapu-11013965

## Table of Contents

Aim:.....	3
Proxy server.....	3
Our Web proxy design: .....	3
Conclusion:.....	7
References: .....	8

# REPORT

**Aim:** In this project, we will develop a web proxy. When our proxy receives an HTTP request for an object from a browser, it generates a new HTTP request for the same object and sends it to a remote server that is hosting the requested object. When the proxy receives the corresponding HTTP response with the object from the remote server, it creates a new HTTP response, including the object, and sends it to the client. The proxy will be multi-threaded, so it will be able to handle multiple requests at the same time.

**Proxy Server:** In computer networks, a **proxy server** is a server (a computer system or an application) that acts as an intermediary for requests from clients seeking resources from other servers. A client connects to the proxy server, requesting some service, such as a file, connection, web page, or other resource available from a different server and the proxy server evaluates the request as a way to simplify and control its complexity. Proxies were invented to add structure and encapsulation to distributed systems. Today, most proxies are **web proxies**, facilitating access to content on the World Wide Web and providing anonymity.

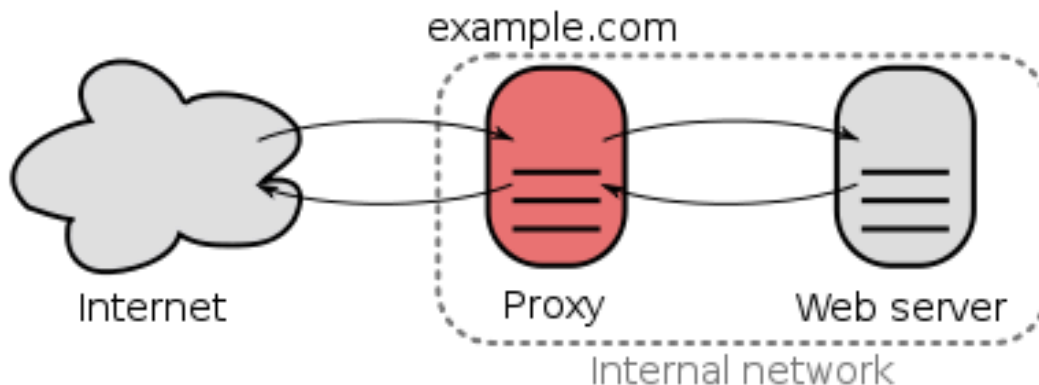


Fig 1: Proxy server

**Our Web proxy design:** In our web proxy we designed server using C language. This server is designed in such a way that it blocks the three given websites [www.youtube.com](http://www.youtube.com), [www.facebook.com](http://www.facebook.com), [www.virus.com](http://www.virus.com), and [www.hulus.com](http://www.hulus.com). The related screenshots are shown below:

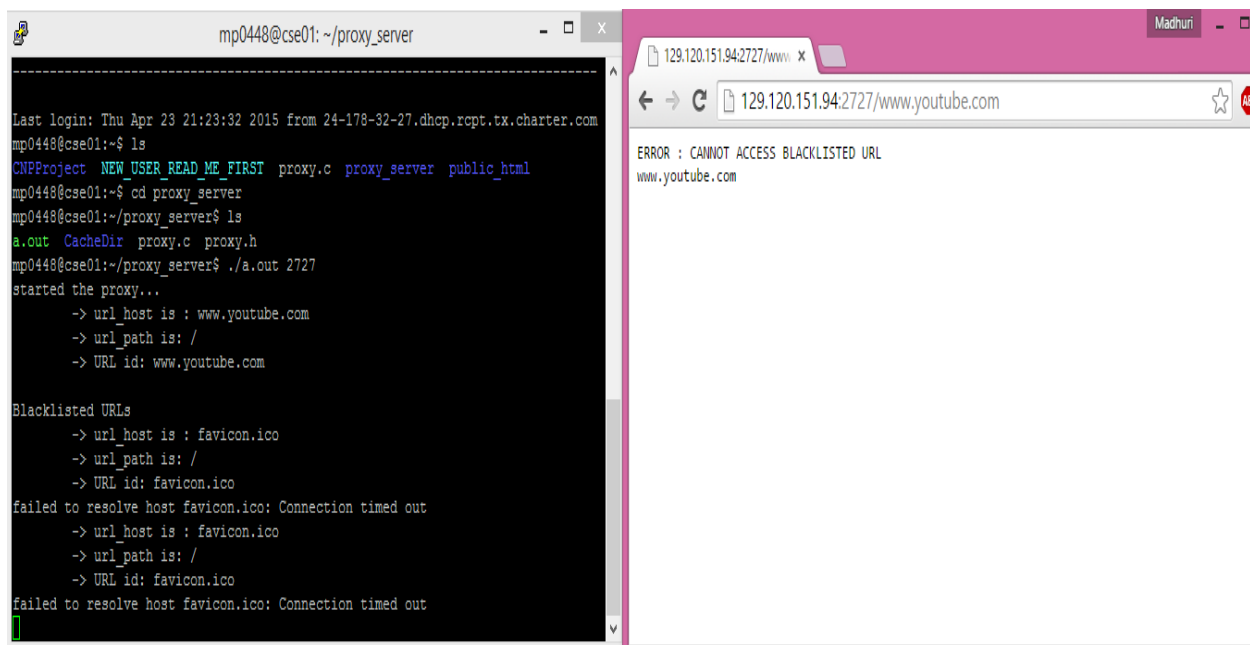


Fig 2: [www.youtube.com](https://www.youtube.com) blocked

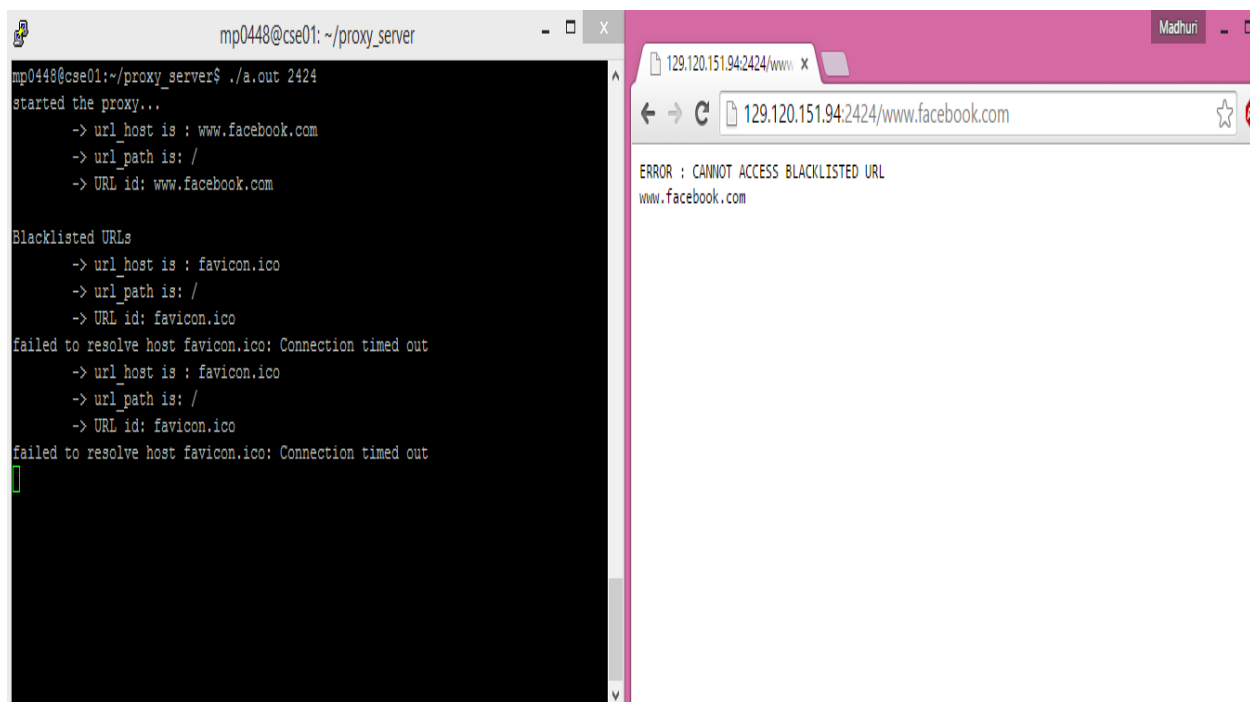


Fig 3: [www.facebook.com](https://www.facebook.com) blocked

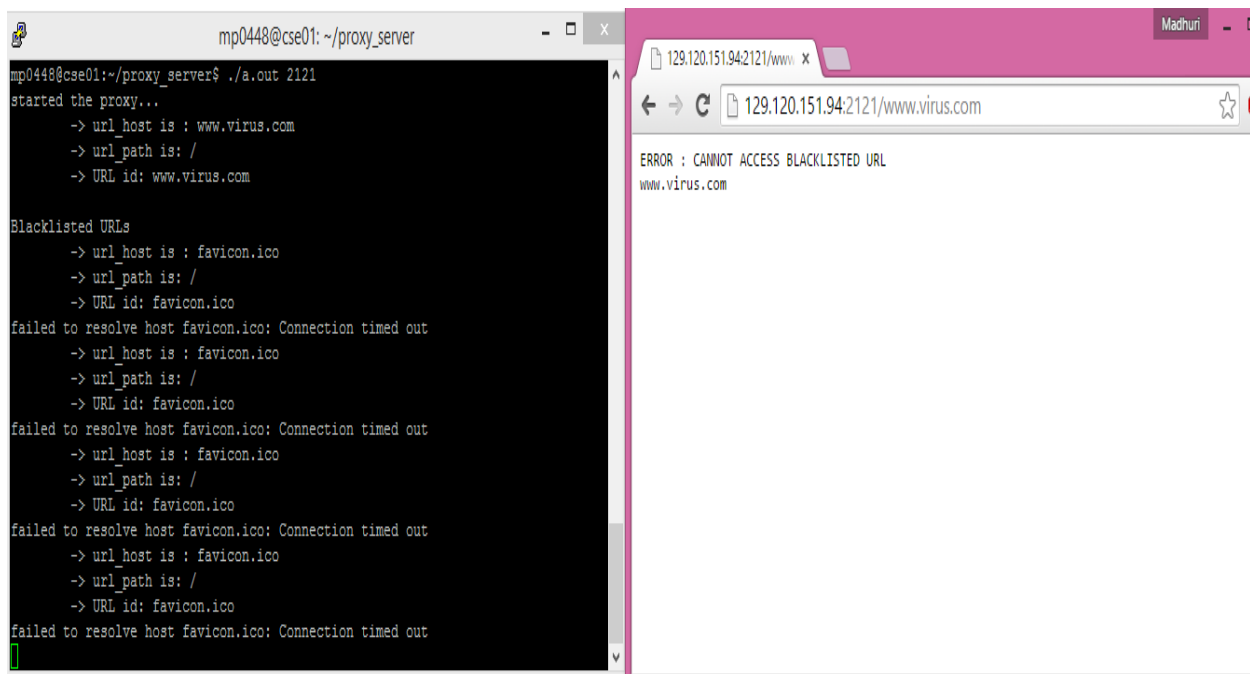


Fig 4: [www.virus.com](http://www.virus.com) blocked

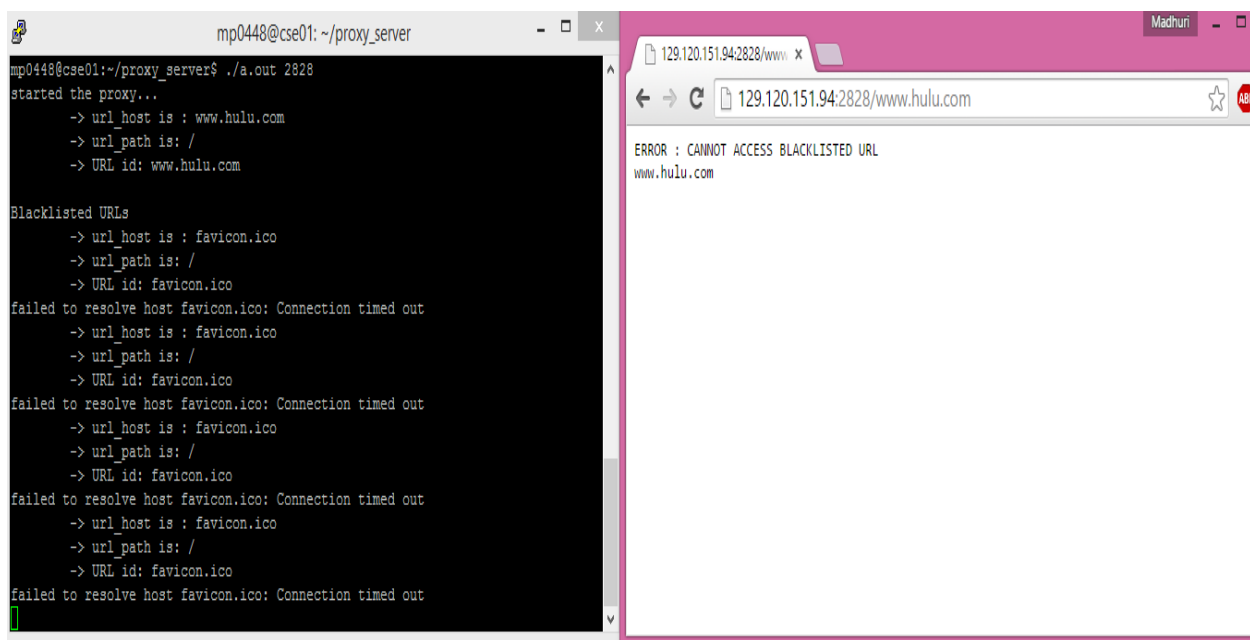


Fig 5: [www.hulu.com](http://www.hulu.com) blocked

We also blocked the given bad words list and the related screenshot is shown below:

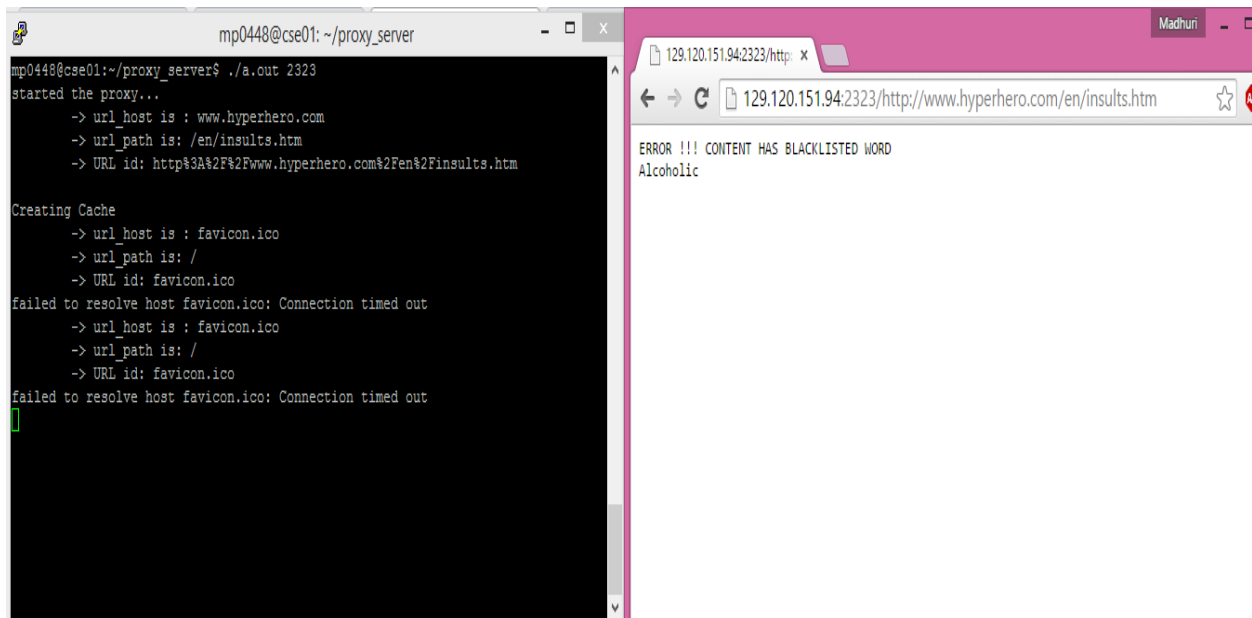


Fig 6: Bad words list blocked

We also designed our server with multi-thread facility which is show below:

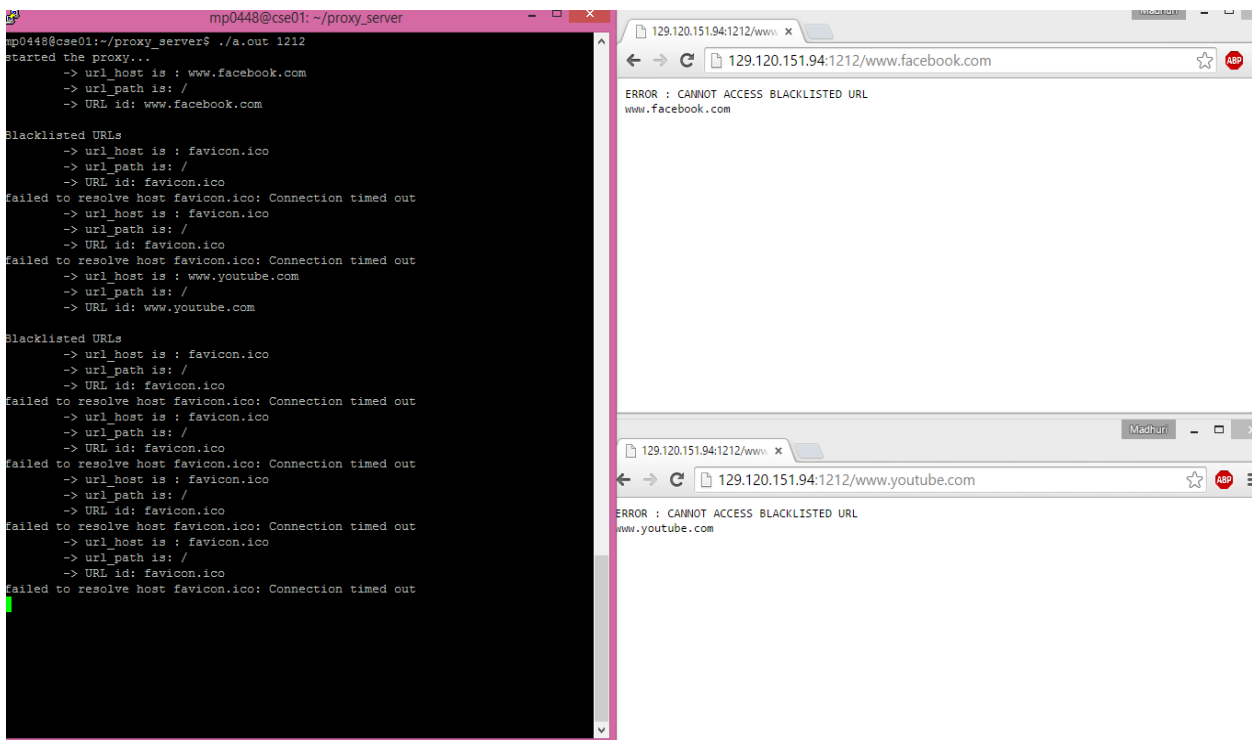


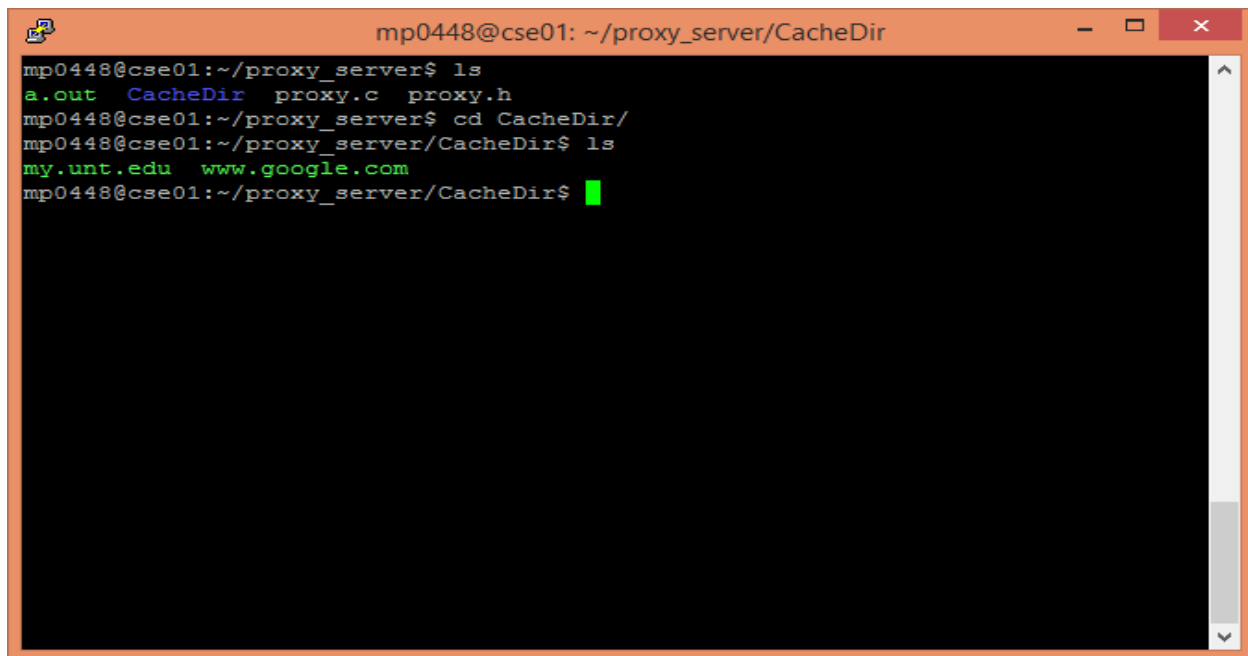
Fig 7: Multi-thread implementation

Below are the screen shots which show the files present in the proxy server and the cache directory of the server:

A terminal window titled 'mp0448@cse01: ~/proxy\_server'. The prompt is 'mp0448@cse01:~/proxy\_server\$'. The command 'ls' has been executed, and the output is 'a.out CacheDir proxy.c proxy.h'. The cursor is on the next line.

```
mp0448@cse01: ~/proxy_server
mp0448@cse01:~/proxy_server$ ls
a.out  CacheDir  proxy.c  proxy.h
mp0448@cse01:~/proxy_server$
```

Fig 8: files present in the proxy server

A terminal window titled 'mp0448@cse01: ~/proxy\_server/CacheDir'. The prompt is 'mp0448@cse01:~/proxy\_server\$'. The command 'ls' has been executed, and the output is 'a.out CacheDir proxy.c proxy.h'. The prompt is 'mp0448@cse01:~/proxy\_server/CacheDir\$'. The command 'cd CacheDir/' has been executed. The prompt is 'mp0448@cse01:~/proxy\_server/CacheDir\$'. The command 'ls' has been executed, and the output is 'my.unt.edu www.google.com'. The cursor is on the next line.

```
mp0448@cse01: ~/proxy_server/CacheDir
mp0448@cse01:~/proxy_server$ ls
a.out  CacheDir  proxy.c  proxy.h
mp0448@cse01:~/proxy_server$ cd CacheDir/
mp0448@cse01:~/proxy_server/CacheDir$ ls
my.unt.edu  www.google.com
mp0448@cse01:~/proxy_server/CacheDir$
```

Fig 9: Cache directory in the proxy server

**Conclusion:** Thus, our design approach of proxy server for blocking some of the websites and blocking the bad words is successful and the related results are shown.

## References:

- [1] Computer Networking: A Top-Down Approach featuring Internet 6th edition, Kurose and Ross, Addison Wesley.
- [2] UNIX Network Programming 3rd edition, Stevens, Fenner & Rudoff, Addison Wesley.
- [3] [https://www.cs.princeton.edu/courses/archive/spring07/cos461/web\\_proxy.html](https://www.cs.princeton.edu/courses/archive/spring07/cos461/web_proxy.html)
- [4] Rabinovich, Michael, and Oliver Spatscheck. Web caching and replication. Boston, USA: Addison-Wesley, 2002