Simulation of a Proxy Server

Augusto Lucas, Almiqdad Elzein, Selmane Tabet

Abstract

One of the main issues that institutions face involve hacking and breaching of important data. Proxy servers can help reduce and alleviate the odds of data breach by encrypting the communication between itself and the client. It also helps reduce network congestion of the institution/company as proxy servers cache recently-requested web pages, making it possible to serve multiple clients after making a single request to a web server. In additional to performance and security-related improvements that proxy servers provide for users in general, it also gives the ability for network administrators to be in control of the internet resources and domains accessible by the users within the network. The aim of this project will be to build an efficient, secure, multi-threaded, web accessibility managing proxy server which manages a network of multiple clients. For this project, the FreeProxy Internet Suite software will be used.

Keywords

Ban Lists, GUI Setup, Packet Inspection, Proxy Server, URL Redirection, Web Caching,

INTRODUCTION

The Advent of technological development and the rise of the information age has brought about numerous ways to manage valuable data and information. Consequently, many of the companies today employ the use of network infrastructures and connectivity with the internet. The development of the internet has also led to heinous methods of stealing data and money from the companies. Specifically, institutions face threats such as hacking, breaching and denial of service attacks. Proxy servers can help reduce and alleviate the odds of data breach by encrypting the communication between the server and Client.

Proxy servers are servers which make web requests on behalf of the machines in a given network. When a machine in a network intends to send a request to some web server, it contacts its local proxy server and asks it to retrieve that desired web page from a specified web server.

In order to gain a better perception of the intricacies of proxy servers, the team used a software entitled the Free Proxy Suite to simulate key characteristics. The team focused on features pertaining to restriction, URL redirection, web caching and packet inspection.

METHODOLOGY

I. Research Phase

The team started the project by looking at current implementation of proxy servers. The team focused on two particular solutions. The first one was to utilize our programming background to simulate the process of the Proxy. The team viewed several source codes from GitHub and deemed the process tedious for a simple simulation. Fortunately, the team discovered a GUI-based setup wizard called the Free Proxy Suite. The Free Proxy Suite is an efficient tool for learning key characteristics of a proxy server.

The FreeProxy Internet suite was developed by Hand-Crafted Software as a medium of internet connection sharing. Since then, it has been expanded to offer a plethora of services and features that define a typical Proxy Server. The general features include Ban Lists, URL filtering and since it is a transparent Proxy, it includes web caching.

Transparent Proxy is defined as a proxy server that is situated between a host computer and the internet. It redirects your requests without modifications. When a user makes a client request to the web server, the transparent proxy processes several interesting responses such as URL restriction and redirection. One of the key features of a transparent proxy is to have web caching.

II. IMPLEMENTATION

The team conducted several trials to highlight the important points of the software. The testing phase revolved around simulation of restricted sites, URL redirection, Web Caching and packet inspection.

1. Initialization Process

The initial part of the setup is to define users and groups for the use of the HTTP Proxy Server. This was supplemented by including a password for authentication purposes. This process is important as it will be later be used for granting permissions to different groups. For the testing of this feature, two groups were created, Admin and Limited. The Admin group was given more freedom to access different web sites which the Limited group was prohibited from accessing. Figure 1.0 shows the initial setup.

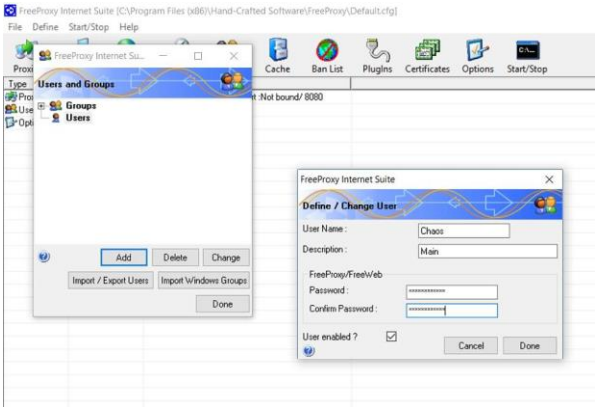

Figure 1.0: Initialization stage

2. Creation of Ban Lists and Redirects

The second portion of the simulation is to create two lists pertaining to ban lists and URL redirection. The Ban Lists feature is used to restrict certain websites to a defined group or user. This is a common feature that is employed in companies and Universities. The second list to be considered is the URL Redirection Lists. This List is similar to the ban list in the fact that they both will not allow the user to access some websites. The difference is that the URL redirection maps the selected website to a different URL instead of an error page. Figure 2.0 illustrates the restriction feature while Figure 3.0 shows the URL redirection feature.

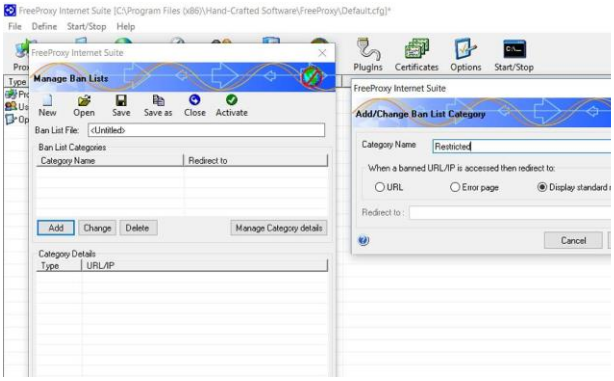

Figure 2.0 Ban List

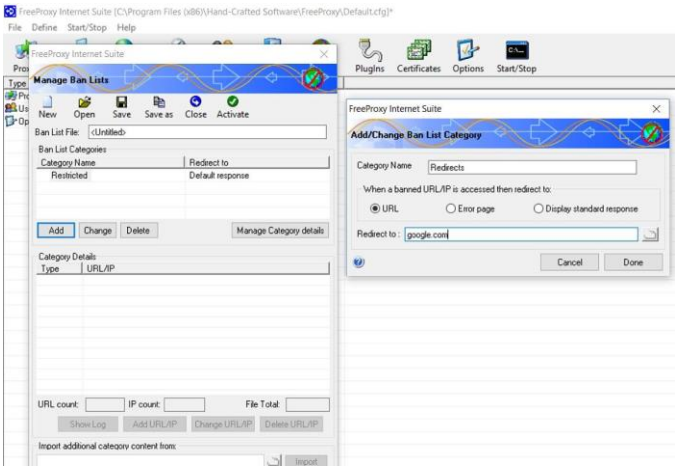

Figure 3.0 Redirection Process

3. Linking Permissions

The software also allowed the feature of setting permissions for the users. For instance, we can choose to limit the access of specific groups of users (or privilege levels) by applying a ban list to a certain user group. The example group shown in figure 4.0 named "Limited" is an example of a group that would have limited access privileges.

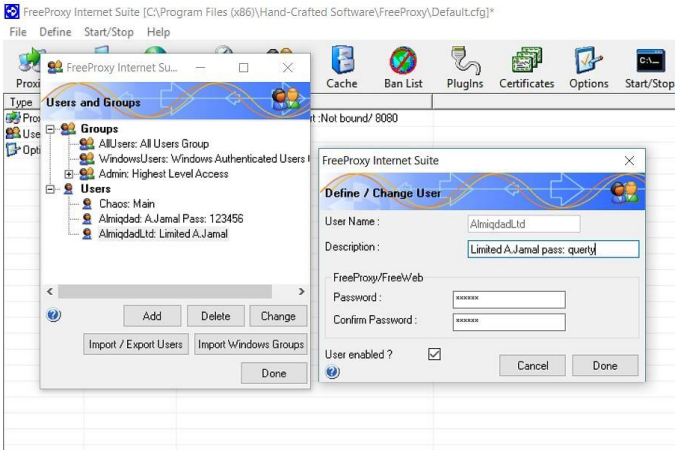

Figure 4.0: Permission allocation

4. Web Caching

One of the benefits of the transparent server is to enable caching. Figure 5.0 shows the process of web caching

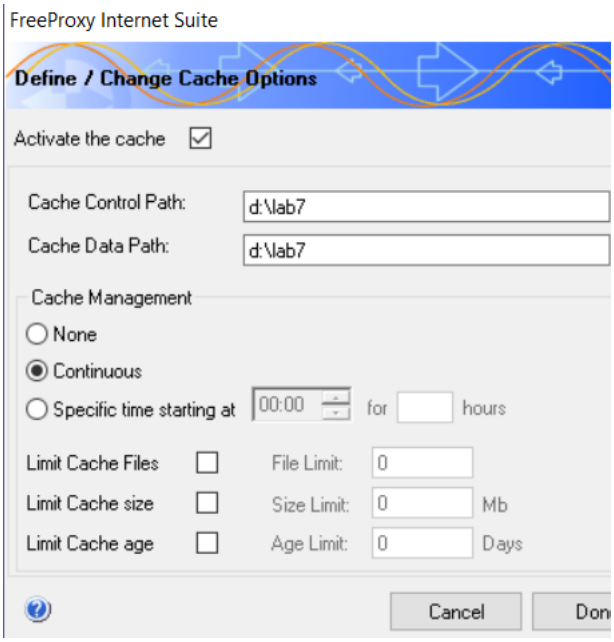

Figure 5.0 Web Caching

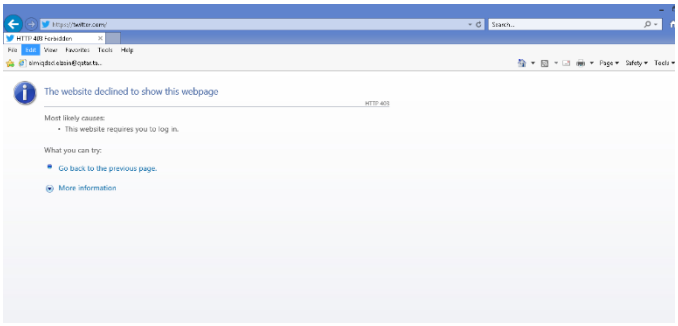

Figure 6.0: HTTP 403

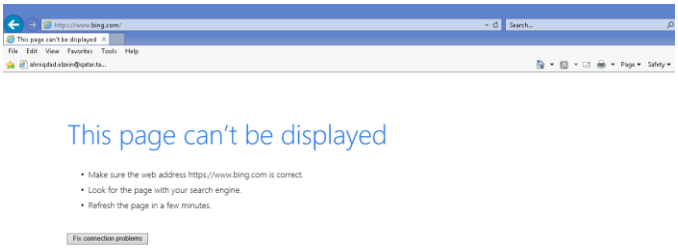

Figure 7.0 Redirection

Results

In testing the proxy server built, the team created a ban list which included a set of restricted URLs, such as twitter.com and bing.com. For twitter.com, the proxy server was configured to block, displaying an error webpage. On the other hand, for bing.com, the proxy server was configured to redirect onto google.com. In other words, visiting twitter.com results in the browser displaying an error message that accessing the site is forbidden and visiting bing.com would result in the browser serving google's homepage instead of bing's. Implementing this, however, proved problematic for sites using HTTPS instead of HTTP. As can be seen from Figure 6.0, accessing twitter.com did not result in displaying the desired error message, which is a built-in webpage that comes with the FreeProxy server. Moreover, Figure 7.0 shows that visiting bing.com did not result in displaying the webpage google.com. It was later discovered that sites which use HTTPS instead of HTTP can be blocked but without displaying the desired webpages. Also, sites which use HTTPS instead of HTTP cannot be redirected from.This is because HTTPS websites use TLS (or SSL) protocols, and thus their traffic is encrypted, making it impossible for their traffic to be interpreted by a proxy server. For HTTP websites, Figure 8.0 shows the error webpage displayed when bbc.com, a restricted website which uses HTTP, is accessed.

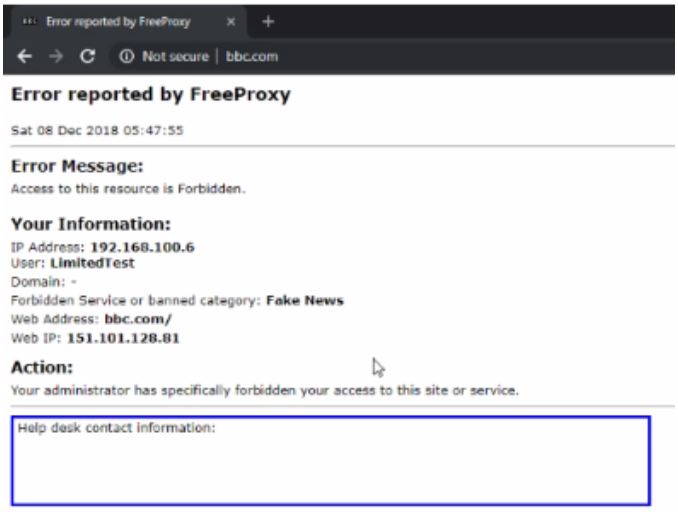

Figure 8.0: Forbidden Website

Discussion

To resolve the encryption problem with HTTPS, it would be required that the certificates of the HTTPS webpages be obtained. This will allow the proxy server to read the websites' traffic and thus be able to block (with displaying the desired error webpage) and redirect from webpages that use HTTPS.

Conclusion

This project successfully simulated the intricacies and key features of a proxy server. This include URL redirection and Restriction, Web Caching and appropriate displayed responses. Furthermore, this project reinforced the fundamental concepts that were discussed in CPEG 460 Networks Class. The concepts include encryption, HTTP messages, Web Caches, Certificates and public keys.