ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Information Security
www.infsec.ethz.ch

Semester Thesis

Information Security Lab: Design and Implementation of a Rewriting Forward Proxy

Contact: michael.naef@inf.ethz.ch

Introduction & Project Objectives

The Information Security Group offers a practical course on information security. The course covers various aspects of operating system and application security. It takes place in a separate laboratory environment.

The lab network is connected to ETHZ's production network via a firewall infrastructure in order to allow for basic services like backup. Apart from that, network traffic from and to the lab is reduced to a minimum.

Although security requirements for the lab environment are strict, some form of Internet access from the lab workstations can prove very convenient and often essential for students working in the lab (e.g. browsing manuals or downloading some tool).

The purpose of this project is to design, implement, and integrate a secure solution that allows basic Internet access from the lab while protecting external sites from unintended malicious activity. To that end, a *Rewriting Forward Proxy (RFP)* shall be designed and implemented.

Rewriting Forward Proxy (RFP)

The RFP works in many ways like a regular forward proxy. Web clients configure the proxy's address and port and are subsequently able to browse the WWW via the proxy. HTTP requests are delivered to the proxy, which in turn issues a new request to retrieve the requested web page from the actual (external) web server. After receiving the response, the proxy forwards the page to the client.

In addition to this functionality, the RFP does not forward any client-provided information to external sites. This is achieved by parsing all web pages in order to identify the hyper links contained. Then, all the links are replaced (rewritten) with the proxy's address and a suitable encoding. The proxy maintains a table with all the encodings along with the original links. When the client requests such an encoded link, the proxy does a lookup in the table and retrieves the original link. All unintended hacking attempts (e.g. URL hacking, HTTP header-based hacking etc.) will thus be blocked on the proxy without leaving to the Internet (or other external networks). The proxy starts up with a pre-loaded table of some commonly used portals like Google, ODP, SecurityFocus etc.

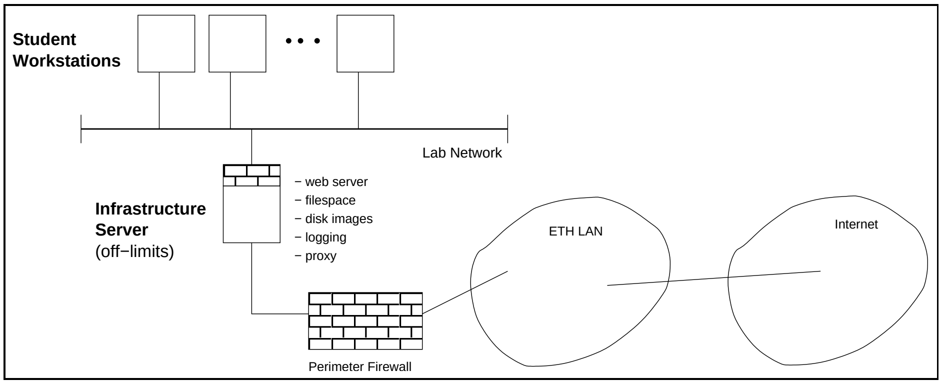
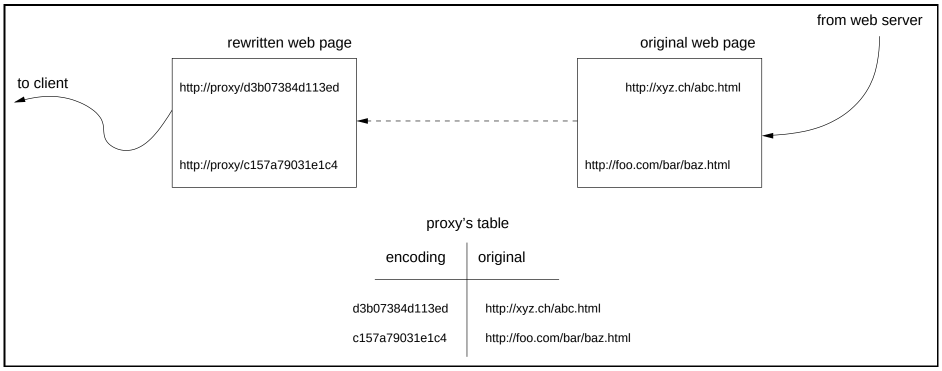


Figure 1: Simplified lab infrastructure with proxy

Figure 1 shows an overview of the lab with the proxy's location. Figure 2 shows a simplified example of how the proxy encodes and rewrites URLs in a web page and simultaneously builds the required table. That table is needed for the proxy to remap encoded URLs to the original URLs when the client clicks on one of the encoded links.



encoding	original
d3b07384d113ed	http://xyz.ch/abc.html
c157a79031e1c4	http://foo.com/bar/baz.html

Figure 2: Example showing proxy's rewriting functionality

Note: The RFP will need specialized handlers that allow to commit a search query to a search engine like Google. A typical Google query looks as follows: `http://www.google.com/search?num=100&hl=en&lr=&ie=ISO-8859-1&safe=off&q=information+security`. The RFP must rewrite Google's search form such that the query is sent to the proxy and then parse the query in a very strict manner.

Work Plan

- Develop and document technical specification for the RFP based on the sketched specification given above. This specification to be discussed with supervisor(s).
- Identify and document all deficiencies of the RFP concept (e.g. form sub-

mission not possible or session management). Analyze those deficiencies as to whether it is possible to devise a (secure) solution for them (e.g. session management capabilities). This will include the design and implementation of a simple handler framework in order to be able to commit queries to search engines. (At least one handler supporting Google must be implemented.)

- Evaluate existing proxy solutions and select a suitable solution that can be extended to provide the specified functionality. (Pertinent solutions include Apache, Squid, Tinyproxy, DeleGate, WebFilter, Muffin, FilterProxy, SquidGuard, WWWOFFLE.)
- Implement the RFP. The following characteristics are of highest priority: security (i.e. compliance with specification), stability/robustness, ease of maintenance (e.g. with regard to future releases of the extended proxy software). The following characteristics are of medium or low priority: performance, ease of administration (i.e. usability of the administrative interface).
- Devise test cases and conduct thorough testing of the implemented solution both with regard to security and stability.
- Integrate the RFP into the lab environment. This includes proper configuration and potential additional measures advisable to increase security. Document the setup and configuration.
- Devise a scheme for persistent encodings in the proxy's table to enable clients to set bookmarks to proxied web pages. (Optional)
- Implement an automated test suite for the test cases above that can be used to verify future modifications or extensions to the code or the setup. (Optional)
- Carry out a simple performance analysis. (Optional)

Prerequisites

Background in networking (especially HTTP) and information security (or the willingness to acquire the required knowledge); software development skills.

Supervision

The project is supervised by Prof. Dr. David Basin and Michael Näf, Chair of Information Security. Please contact Michael Näf (michael.naef@inf.ethz.ch / 26876) if you are interested in this project or have any inquiries related to it.