**User Manual**

Contents

* how to compile,
* Configure,
* Using the Web proxy
* Features of the web proxy
* Limitations
* Testing and
* Conclusion

**Compile**

The code is written using Java in Eclipse, and the working environment was Windows. Eclipse is automatically compiling the source code. Go to Project menu and select the Build automatically option, then it will always compile before running the code.

**Configure**

In Windows, web browsers use the proxy settings configured in Control Panel -> Internet Options -> Connections -> LAN Settings ->Proxy Server where Use a proxy server for the LAN should be checked and the Address and Port should be entered.

The proxy settings for the browser(Firefox, chrome and Opera) can be configured regardless of the proxy settings of the OS. Depending on the OS, you can find the Options window for Firefox either under Tools->Options (e.g. in Windows) or Edit->Preferences (e.g. in Solaris). In the Options window, select Advanced->Network->Settings... and choose the Manual proxy configuration option. Here you can enter the HTTP Proxy address and Port.

HTTP Proxy: 127.0.0.1

Port: 8888

**Using the web proxy**

Once you configure your web browser and control panel, run the proxy

next start your browser, and type a URL that contains banned words in the web address, this can also be done by typing in Google search engine.

It is always recommended to clean the caches before running the browser, as it may contain previously browsed history.

**Features that the proxy supports**

1. The proxy support both HTTP/1.0 and HTTP/1.1.
2. Handles simple HTTP GET interactions between client and server
3. Blocks requests for undesirable URLs, using HTTP redirection to by displaying the following error page

Sorry, but the Web page that you were trying to access is inappropriate for you, based on the URL. The page has been locked to avoid insulting your intelligence.

Net Ninny

1. Detects inappropriate content bytes within a Web page before it is returned to the user, and redirecting the above error page
2. Is compatible with all major browsers (e.g. Internet Explorer, Mozilla Firefox, Google Chrome, etc.)
3. Is smart in selection of what HTTP content should be searched for the forbidden keywords. For example, you probably agree that it is not wise to search inside compressed or other non-text-based HTTP content such as graphic files, etc.

**Limitations:**

The proxy cannot scan bad contents coming in the form of images, videos and audios from the web client to the web browser.

**Testing:**

The testing was done both at home and the university and both are successful.

The web proxy has been tested and works perfectly for the following samples;

Download a simple text file such as the

<http://www.ida.liu.se/~TDTS04/labs/2011/ass2/goodtest1.txt>

Download a simple HTML file such as the

<http://www.ida.liu.se/~TDTS04/labs/2011/ass2/goodtest2.html>

Download an HTML file with a bad name such as the

<http://www.ida.liu.se/~TDTS04/labs/2011/ass2/SpongeBob.html>

Download an HTML file with a good name but bad content such as the

<http://www.ida.liu.se/~TDTS04/labs/2011/ass2/badtest1.html>

Download various pages that you would expect a regular user accessing

[www.google.se](www.google.se%20)

[www.google.com](www.google.com%20)

[www.youtube.com](www.youtube.com%20)

<www.wikipedia.org>

**Conclusion**

The proxy server receive an HTTP request from browsers(client) such as Mozilla Firefox, Google chrome and Opera. It scans if the URL in the request has bad words in that it return an error page. Otherwise it passes the request to the server side if the URL does not contain any of those bad words. The proxy then passes the request to the web server and receives content from the web server. Next the proxy will scan the content of the reply for bad words in that it will forward an error page to the browser, Otherwise the proxy will passes the web content to the server which in turn sends the content to the browser.

**Figure 1. Request and response flow**