academicJournals

Vol. 11(13), pp. 157-167, 16 July, 2016
DOI: 10.5897/IJPS2016.4482
Article Number: 850997559614
ISSN 1992 - 1950

http://www.academicjournals.org/IJPS

International Journal of Physical Sciences

Full Length Research Paper

Development of an identity management system for a web proxy server in a tertiary institution using anonymity technology

Fashoto Stephen Gbenga[1]*, Adekoya Adekunle[2], Owolabi Olumide[3], Ogunleye Opeyemi[2], Adediran Saseyi[2] and Tomori Rasheed[4]

[1]Department of Computer Science, Kampala International University, Kampala, Uganda.
[2]Department of Computer Science, Redeemer's University, Ede, Osun State, Nigeria.
[3]Department of Computer Science, University of Abuja, Abuja, Nigeria.
[4]Department of Computer Science, University of Ilorin, Ilorin, Nigeria.

Received 2 March, 2016; Accepted 4 July, 2016

The inability of a region to access a webpage, because of the ban being placed on users from that region as a result of its location policy, has led to this study. This problem is often solved by anonymizing web traffic by using The Onion Router (TOR). These tools, however, suffer from the problem of exposure of identity and also lack the ability to monitor web users. This study describes in detail a web proxy server service solution within the context of a tertiary institution in Nigeria and explains how this service improves the user experience. An identity management system using a web proxy server was developed to tackle these problems. The new system proxy was designed using a transparent proxy model with some additional translational features where no modification was done to the response or request of resources, other than the addition of its identification information or that of the server from which the message was recovered, and mediation of resources. Redeemer's University proxy was used as a case study in this research work. This system is also able to effectively monitor users' (staffs and students) operations on the web.

Key words: Web proxy, web anonymity, identity management, The Onion Router (TOR).

INTRODUCTION

Anonymizers allow internet users to surf the web anonymously. They also enable internet users maintain a certain amount of privacy which deter gathering of known information like internet protocol (IP) address when browsing (Li et al., 2011). These anonymity services are offered by profit-making organization propelled by subscription fees, non-profit making organizations benefitting through marketing, and home-brewed services through open source anonymous tools. Examples of Community contributed systems are the onion router

*Corresponding author E-mail. gbengafash@yahoo.com.

(TOR) (Dingledine et al., 2004), the Invisible Internet Project (I2P) and the Java Anon Proxy (JAP) (Berthold et al., 2001). Proxy servers are deployed to keep clients anonymous, to obstruct unnecessary content on a network, to save network bandwidth by supplying generally accessed data to clients, and to log and audit client usage (Squid-Cache Wiki, 2014). There are four main types of proxy servers used to gain a degree of anonymity. They are Transparent proxy, Anonymous proxy, Distorting proxy and High Anonymity proxy.

1) Transparent Proxy: It is a form of proxy server that enables the primary IP address to be accessible via the http headers. It is often used due to its capability to cache websites by providing anonymity for its users. Transparent Proxy is known for its transparency because the lP address is visible to everyone. However, it could also be tagged non transparent because its users may not be aware that they are using it.

2) Anonymous proxy: Anonymous proxy associates itself with a proxy server. It does not make its primary lP address visible. Though, this kind of proxy server is detectable, but provides a kind of security to most of its users.

3) Distorting proxy: This is another form of proxy server that allows a wrong lP address obtainable through the http headers.

4) High anonymity proxy: Lastly, this type of proxy server does not in any way relate to proxy server. It does not make its original IP address available. In this study we consider the anonymous proxy.

In the context of TOR, anonymity means preventing the dissemination of a user's internet protocol address. The anonymity set is an assemblage of the sender (source), receivers (destinations) and the servers in a communication network. Anonymity creates two types of users: unidentified users and unlinkable users. Unidentified users are those who post messages without their real name or agent name while unlinkable users are those who post messages without their real name being linked to their account or agent. Anonymity technologies can be used for legal and illegal purposes. Examples of anonymity for legal purposes are privacy, freedom of speech, anti-censorship, and so on, while protection to criminals in facilitating online crimes such as spam, piracy, identity theft, prevention of web filters from monitoring, exposing organization to malicious activities and finally abusing organization resources such as the use of YouTube are examples of anonymity for illegal purposes.

The inability of a region to access a webpage because of the ban being placed on users from that region as a result of its location policy and, also, the effective tracking of users using a network led us to formulate this study. For example, a user trying to access a website which has barred users from certain locations from gaining access and permission to its webpage as a result of its privacy policies would be unable to access the services offered by the site. In addition, the possibility of monitoring the activities of users on a network without a monitoring system can overwhelm and limit the usage and effectiveness of the network.

An identity management system refers to a set of technologies or an information system that is used for project or cross-network identity management. It involves the organization of an individual entity's identity, authorization and authenticating, and also privileges contained in or across system boundaries with the purpose of escalating security and output while diminishing cost, downtime, and monotonous tasks (Wikipedia, 2014). This identity management system would be useful to users from states that are being barred from accessing some web pages in other regions that does not recognize that particular state in context. The introduction of a web proxy in the system for the case of the web will help user agents anonymize their identity. Proxy servers are systems positioned to act as an intermediary for clients seeking resources from other servers or clients to connect to the World Wide Web. This research work intends to design and develop an identity management system using web proxy. From previous research work done in this area, several systems have been developed using diverse models. Over the past years, researches have been conducted in the area of web proxy such as the performance analysis and optimization of web proxy servers and mirror sites by Gautam, et al. (2013) (Improving Performance on WWW using Intelligent Predictive Caching for Web Proxy Servers) which is being integrated into different identity management systems as a method for tracking visits to a web page and anonymization of clients. For instance, when a web request is relayed through the identity system proxy to a server, the server or website sees the proxy as the requesting client. More so, when the server or website tries to identify the name and location of the user from which it received a request, it finds the proxy instead of the client behind the proxy. In 2009, Jelenkovic and Radovanovic reported that web caching improves the web performance by storing web objects close to clients, which reduces the latency of delivering web objects to the end-user (Jelenkovic and Radovanovic, 2009). In addition, it makes the most of the bandwidth of the network which is considered as an important goal for network administrators and it also reduces the load on the origin servers. An additional tool can be used in attaining anonymization of clients in the protection of details about the request originator from the target server which makes such a disguise required in situations, particularly in the case of web browsing since web traffic anonymization is not part of the http specification (Sochor,

2013).

It has been observed that students and some staff of Redeemers University of Nigeria (RUN) have been facing some of these drawbacks in terms of gaining scholarly materials from regions that has not recognized or banned them for obtaining their services and also, the system was used to monitor users (staff and students) operations effectively on the web. This study developed identity management system using web proxy server to tackle these problems.

This study was based on utilizing some set of data to assist users in gaining access and also network administrators in tracking the operations of users on the network. The aim was to develop a robust system that satisfies both students and staff of RUN in all aspects of gaining scholarly materials from all parts of the Web with the effective tracking of user operations on the web by the use of a proxy.

The next section introduces us to the review of relevant works, the concept of web proxy and identity management system, while subsequent sections present the methodology, implementation, conclusion, and references, in that order.

REVIEW OF RELATED WORKS

The emergence of the connected world has brought about a wave of trends in human day to day interactions. In recent years a great deal of attention is being placed on security in the information society to the problems being faced in the management and protection of vital details and identity. Several identity management systems have been developed by various individuals and communities to suit their needs for the protection of critical details. The information gotten from these works has been used effectively in the creation and improvement of diverse identity management systems in different fields.

The growing popularity of the Web necessitated the roles of proxies on the internet. Web proxies emerged as one of the most common intermediaries in the transmission of web messages between user agents and origin servers as shown in Figure 1. Proxies began to play a vital role as early as 1994 (Balachander and Rexford, 2001).

The original purpose of a web proxy was to provide access to the web for clients who are behind an organizational firewall by ensuring the clients did not lose any functionality when requests and responses are being routed (Luotonen and Altis, 1994). Thus, the first stated design criterion for a web proxy was to ensure that clients do not lose any functionality by having their requests and responses routed through a firewall. The original web proxy was actually a version of a gateway server at the CERN laboratory, where the world's first web server was created (Luotonen and Altis, 1994). The CERN http server (the first web proxy) arranged caches hierarchically (Mahanti et al., 2000). A web proxy's role in information hiding and encapsulation is described in Shapiro (1986). Web proxies were also originally designed to allow network administrators control internet access from within the intranet (Baentsch et al., 1997). On the other hand, resulting inquiries about proxies show that they serves as intermediaries for commonly requested documents. Thus, web proxies, which started as a gateway server, have now become a vital part of web user's experience on the Web.

Web anonymity

A web proxy can be defined as an application program that accepts the retrieval of documents from some clients, and relay these requests to the suitable servers (if need be), and then send the requested documents back to the clients (Fielding et al., 1998).

Web proxy is relatively efficient and fast. It aids the anonymization of clients behind it, monitors and helps the administrator to effectively track its users and performs many other functions.

Squid-Cache Wiki website in 2013 defined a proxy as a popular and useful intermediary on the web that enables a large number of clients behind it to share access. Proxy servers are systems deployed to act as a mediator for customers looking for assets from different servers or clients to connect to the World Wide Web. Proxy servers are deployed to keep clients anonymous, to block unwanted content on a network, to save network bandwidth by supplying commonly accessed data to clients, and to log and audit client usage. A proxy basically goes about as a system 'go between'. As opposed to asking for a web object straightforwardly from a remote host, the client may ask for it from an intermediary, which thus acquires the article itself and advances the response to the requesting client (Piatek, 2004).

Identity management

Identity has become a new focal point in today's global world; likewise, Identity management has become a significant factor in this era due to its potential value as it aids the proper handling of sensitive data (Gaurav and Pruthi, 2012). The management of different identities is a key element of information system safety. It involves the protection of various individuals' digital identities. Digital identity alludes to ascribed qualities credited to a person, which are instantly available by specialized means (Müller and Böhm, 2011). Alternatively, digital identity refers to a situated set of qualities and properties around an individual that are related together and accessible in an electronic structure to build trusted digital certifications (Al-Khouri, 2012). The identity of an individual consists of

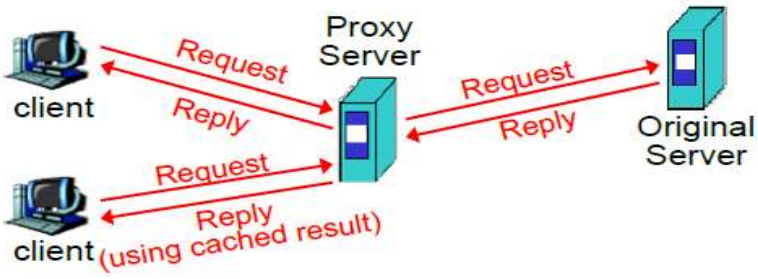

Figure 1. A Web Proxy server (NCTU-Education, 2007).

a vast number of private information with respect to the person. Identities of individuals are exposed in different contexts; that is why an identity management system is used in managing various partial identities.

Identity management systems furnish the administrator with the apparatuses and tools expected to change a user's part, monitor client exercises and to implement policies on client premises. These frameworks are intended to give a method for regulating client accesses over a whole venture and to guarantee consistence with approaches and government regulations. The objective of identity management is to offer extensive entities with the capacity to utilize a concentrated database of client characters to streamline work process and computerize undertakings managing client confirmation, access rights, arrangement authorization, and provisioning of both physical and electronic assets (SANS-Institute, 2003).

Web privacy has been a noteworthy topic amongst the World Wide Web community. Anonymity is being applied in diverse ways across the web due to the privacy concerns raised. Ruiz-Martinez, in his work “A survey on solutions and main free tools for privacy enhancing Web communications”, emphasized the need for users to perform anonymous communications when surfing the internet in other to protect their ever increasing digital identity. This significance is also mutual to both the users and the research community (Gross and Rosson, 2007). Gross and Rosson expatiated on different privacy enhancing tools, among which is the web proxy. An anonymous Web proxy(also known as anonymizer) behaves as a TCP proxy and gets rid of headers with client’s information (or fake them), rephrases HTML pages so that when the client connects to a link on that webpage, the request is requested through the proxy and hides the client’s identity on the network (Shubina and Smith, 2003).

In general, these systems also manage cookies for the client but, at the end, the demand originates from the same IP address and as a result, the IP address of the client can be known. Furthermore, if the intermediary is a third party, then, the address of the client on the network cannot be known (Edman and Yener, 2009; Li et al., 2011). The advantages are the towering effectiveness they proffer, simple to contact and to use and ease. Web proxies do notrequire extra components (Edman and Yener, 2009). The principal shortcoming of using web proxies is that a straightforward anonymizerdoes not guard against traffic examination even though aSSL/TLS link is being used as researched by several researchers in the web community.

Sochor (2013) reported in originator from the intended server. In his study, the focus was on the anonymization tool which is known as TOR (The Onion Routing). TOR can be said to be an extra tool since web traffic anonymization is not a component of the http specification, to create a disguise especially in the case of web browsing. Noteworthy deceleration of anonymized traffic contrasted with ordinary activity is inescapable yet it can be controlled now and again as this study proposes. The outcomes exhibited in the study concentrates on measuring the parameters of such deceleration regarding response time, transmission speed and latency and proposing routines on the most proficient method to control it. The study concentrates on TOR mainly in light of the fact that recent studies by (Liska et al., 2010) and (Sochor, 2012) have reasoned that different tools (like I2P and JAP) give worse services. Sets of 14 record areas and 30 web pages have been shaped and the dormancy, reaction time and transmission rate amid the page or document download were measured over and over both with TOR dynamic in different designs and without TOR. The major result offered includes various ways on the most proficient method to enhance the TOR anonymization proficiency and the proposition for its programmed control. Disregarding the way that productivity still remains too low when compared to ordinary web activity for standard use, its programmed control could make TOR a functional instrument in uncommon cases. The study also conducted a deeper analysis of TOR behavior, especially with respect to the possibility of improving the TOR behavior and efficiency. Finally, the study reported that in order to achieve and sustain the best results for TOR, it must ordinarily be used in the environment of permanently changing www communication. Automatic control using a fuzzy controller or an artificial neural network was proposed with the expectation that this will allow for the automatic tuning of anonymization parameters.

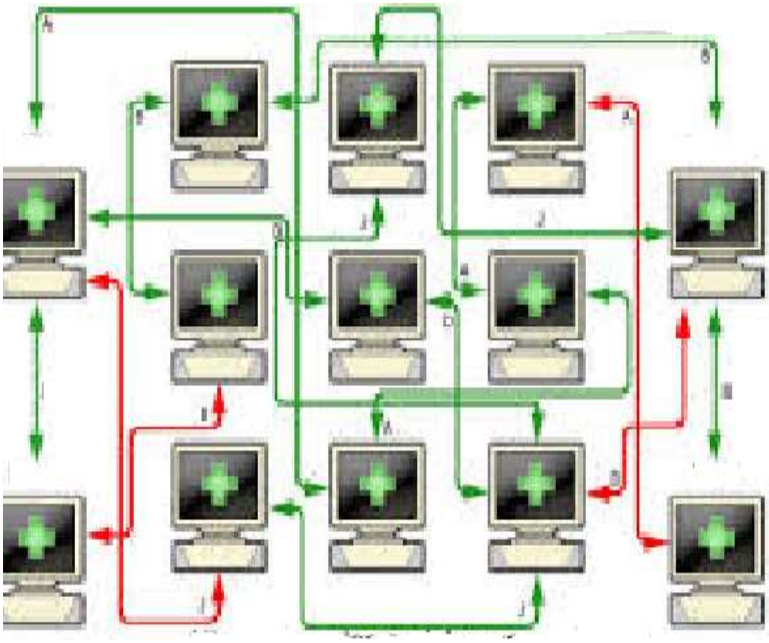

Figure 2. Architecture of The Onion Router (TOR).

METHODOLOGY

To achieve our solution we employed the Rapid Application Development (RAD) methodology for this study. It provides the ability to change the system design as demanded by the user. In addition, the Rapid Application Methodology has a focused scope, such that the purpose is well defined and narrow, providing a comprehensive functionality of the system which are clearly noticeable at the users interface. MySQL was used for the database (the back-end design) while Java Development Kit (JDK) using *Netbeans 6.8 IDE* and PHP was used for the front-end design.

Description of current system: The Onion Router (TOR)

TOR is a popular free software with a network of servers that allows web users perform anonymous communication. The Onion Router is known to be one of the most accepted overlay networks for anonymizing TCP traffic and also for safe and secure online browsing (Bauer et al., 2007). It is noteworthy to mention that TOR does not provide internet security but it offers a wide range of anonymity. It is perceived tough anonymity properties and its moderately low latency service makes it very useful as an anonymizing tool. Low latency is reached through TOR's ability to poise the traffic load by bettering the TOR router selection to probabilistically support routers with high-bandwidth potentials. It directs internet traffic through a liberated, worldwide, volunteer network consisting of over six thousand relays to hide a user's location and usage from any person conducting network observation or traffic analysis. The onion routing design is simply wrapping traffic in encrypted layers in order to guard the contents of the data as well as the anonymity of the sender and recipient.

In the TOR architecture, shown in Figure 2, several basic models are defined as follows: A TOR proxy is the client component of the network that places the user's traffic into the network of TOR routers. A TOR router is the server section of the network that is responsible for promoting traffic within the central fraction of the network. We can analyze the Tor proxy as a service that runs on the user's computer.

Solution provided by the new system

This study is focused on improving and proffering solutions to the problem faced by the current system. The current system is faced with the problem of exposure of identity and it also lacks the ability to monitor web users. The technical details in solving the problem are discussed below.

Firstly, the solution to the problem above is to create a web proxy that provides and guarantees the security of users surfing anonymously on World Wide Web.

The second aspect to the solution is to identify the best possible way to browse anonymously by developing a transparent proxy. The virtualisation of the proxy will allow as many users as possible to surf the internet securely through a more effective transmission speed and make anonymous surfing easier unlike TOR which will only allow a limited amount of security and reduce the transmission speed.

Overview of the new system

The new system, RUN Proxy, was developed basically to implement identity management with features to aid the anonymization of users to enable them surf the internet anonymously without being barred. An added advantage of the system is that it can track and monitor the operations of the user.

It was designed and developed essentially to interface with the client system (that is, web browser). The RUN Proxy was designed using a transparent proxy model with some additional translational features where no modification was done to the response or request of resources, other than the addition of its identification information or that of the server from which the message was recovered and mediation of resource. The RUN Proxy would certify that the length of the message remains unchanged also.

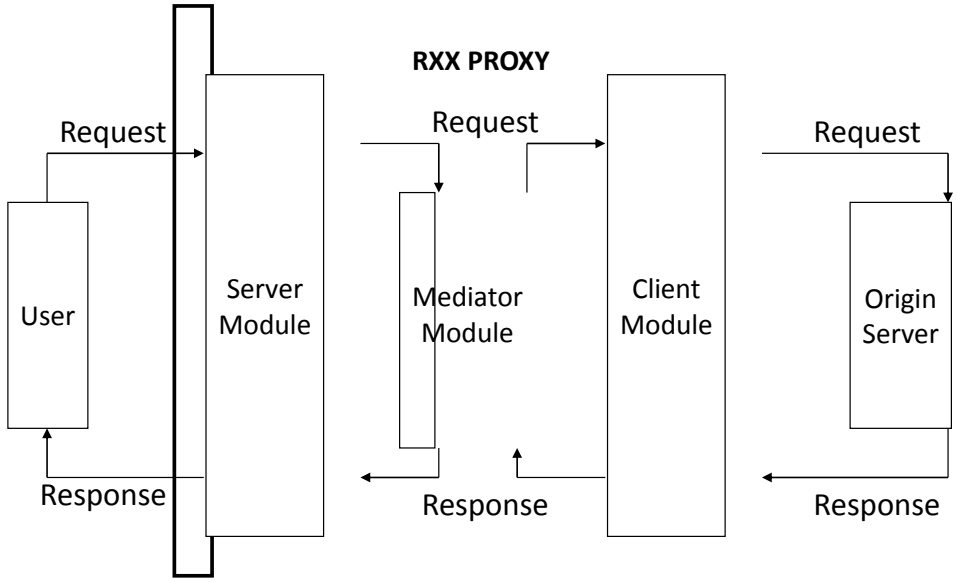

Figure 3. RUN Proxy Architecture Dataflow diagram.

Description of the proposed system

The RUN Proxy, shown in Figure 3, was designed and implemented with some features of HTTP 1.1 and performs the same function as the HTTP Proxy Server. The HTTP protocol is a request/response protocol communicating between the client and server. The client launches a request to the server in the form of a request method, followed by a message containing request modifiers, client information, and possible body content over a connection with a server. The server responds with a response line, including the message's protocol version and a success or error code, trailed by a message having the information about the server, and possible entity-body content. The RUN proxy will act as an intermediary between the client and the server, acting as a server to the client and as client to the server.

Algorithm for the RUN Proxy

1. Start
2. Proxy listens to connections from clients (web browser)
3. Proxy accepts the connection;
If connection is found, then proxy accepts the connection
4. Proxy creates a new thread for the connection
5. Proxy gets a request from the client after the connection has been established
6. Proxy creates a default url "" and default port "80"
7. Proxy tests the connection by performing the following;
a. Proxy checks if the first line is "connect",
If connect exist, proxy closes the connection
b. Else, proxy checks if the first line is a "request line"
c. Else, proxy prints "unknown first line"
8. The proxy processes the request line
9. Proxy enters a loop in other to test the request message
10. Proxy tries to fork a connection with the origin server
11. Origin server creates a new thread for connection between proxy and itself if connection is accepted
12. Proxy relays the request message to the origin server via the created connection
13. The origin server processes the request message from the proxy
14. Origin server sends a response message containing the message header and message body
15. Proxy deciphers the message header from the message body via the last line which is "<cr><lf>"
16. Proxy processes the response header from the origin server by reading each line of the header field until it reaches an empty line
17. Proxy processes the response body by determining whether the message body is chunked or not from the content length read in the header.
18. Proxy processes the response body from the server. It performs the following;
a. If message body is not chunked, the proxy processes the body
b. Else, if the message proxy is chunked, the proxy processes it until a "0" is found.
19. Proxy sends response message to the client
20. Proxy requests from client if the connection should be closed or not;
If connection should be closed, proxy closes the connection
Else proxy maintains the connection if the following are true;
a. If server connection is lost
b. If clients does not respond to proxy in 10 seconds
21. Stop

Flowchart for the RUN Proxy

The RUN Proxy flowchart is a form of diagram that represents the steps in the algorithm showing a diagrammatic illustration of the steps taken by the proxy in relaying the request message from the client (web browser) to the origin server, and also the sending of the response message from the origin server back to the client. The RUN proxy flow chart is shown in Figure 4.

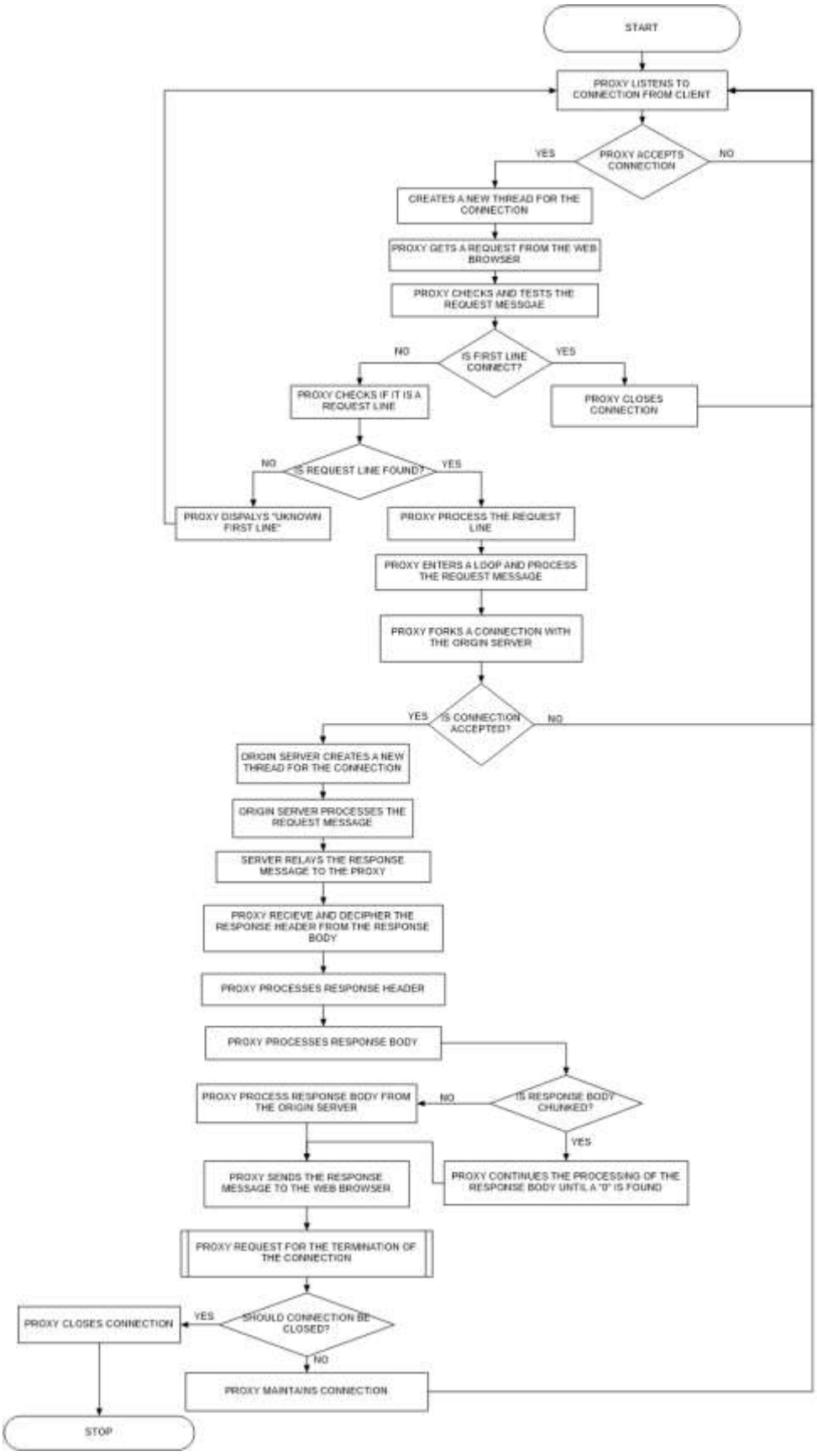

Figure 4. RUN Proxy flowchart.

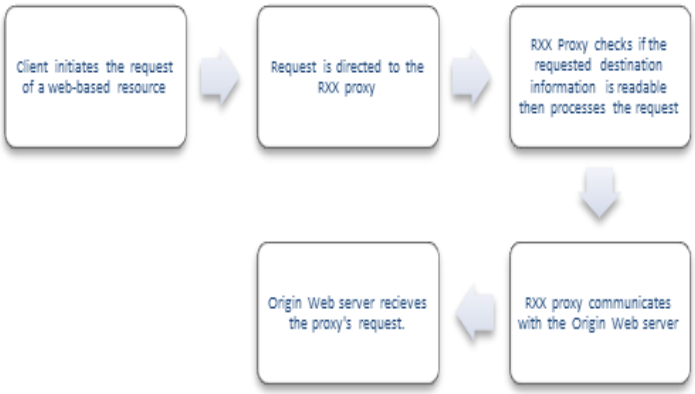

Figure 5. Client (Web brower) requesting for a resource through the RUN proxy.

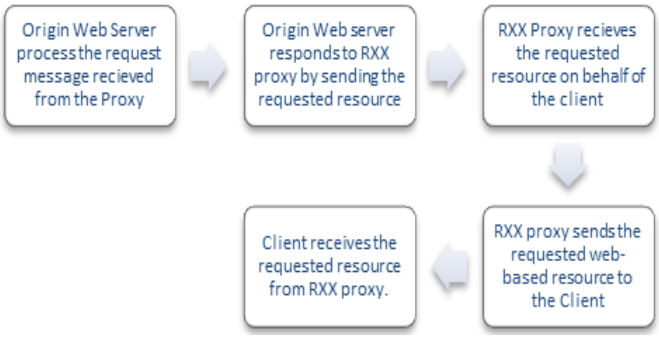

Figure 6. Server responding to the resource requested by the Client via RUN proxy.

Process flow diagrams

User requirements

The user requirement for the RUN Proxy is that the user must ensure its web browser is configured to use the proxy (Figure 5). This is done by setting the browser to accept HTTP traffic on the proxy's port and IP address (Figure 6).

FINDINGS AND DISCUSSION OF SYSTEM TESTING RESULTS

Here describes the results obtained from the tests carried out on the system, and it gives a detail report on the tests and also the interface which displays the usage.

The RUN Proxy interface displays how it listens to connections. Figure 7 illustrates how RUN Proxy behaves when no connection has been established. It displays an empty screen because there is no connection of any sort with the proxy at the current time. To establish a connection, a web client (browser) must initiate a connection with the proxy. The configuration of a web proxy can be done by setting the LAN setting of the computer system to the proxy's IP address and port number.

Figure 8 displays the RUN Proxy's functionalities when the web browser whose connection was routed through the proxy establishes a connection. It shows how a resource is requested by the client to an origin server via the RUN Proxy. The URL accessed by the proxy for mediation is (http://www.greens.org/about/software/editor.txt). It also shows the information of both the client and the origin server which aids the effective tracking of client's web patterns and activities.

The highlighted portion in Figure 9 is responsible for the creation of a socket which is an abstraction of a client–server communication. The portion presents how the RUN proxy checks if the request line was found.

Figure 10 explains the process on how the RUN Proxy detects the request line in a request message. The source code illustrates the method it uses to check for some HTTP methods to determine the presence of a request line.

Figure 11 shows how the RUN proxy displays the processing of resources that are chunked. The URL for

Figure 7. RUN Proxy waiting to establish a connection.

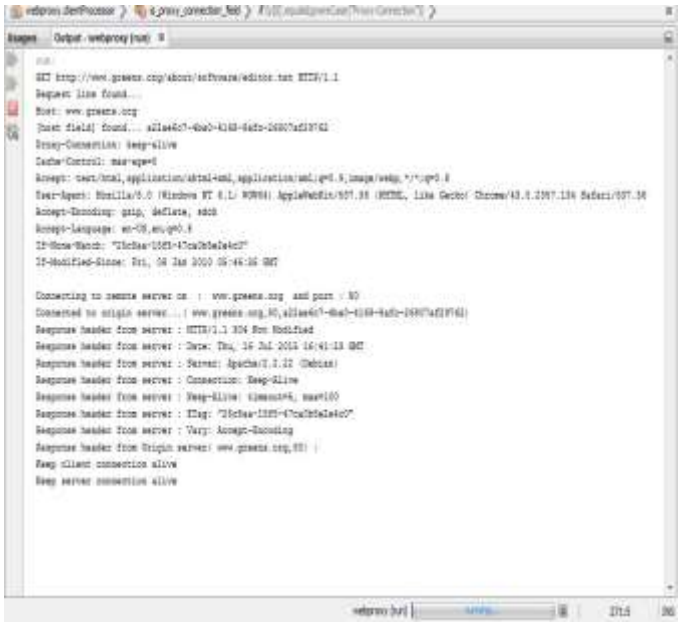

Figure 8. RUN proxy functionalities.

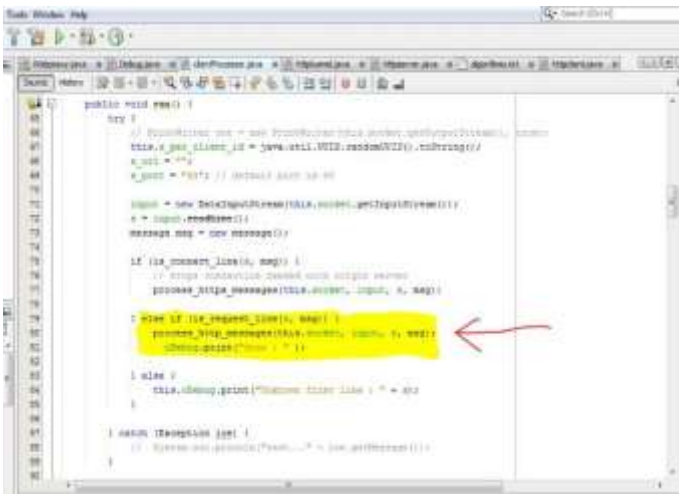

Figure 9. Source code for the creation of a socket.

Figure 10. Source code portion on how the request line is detected.

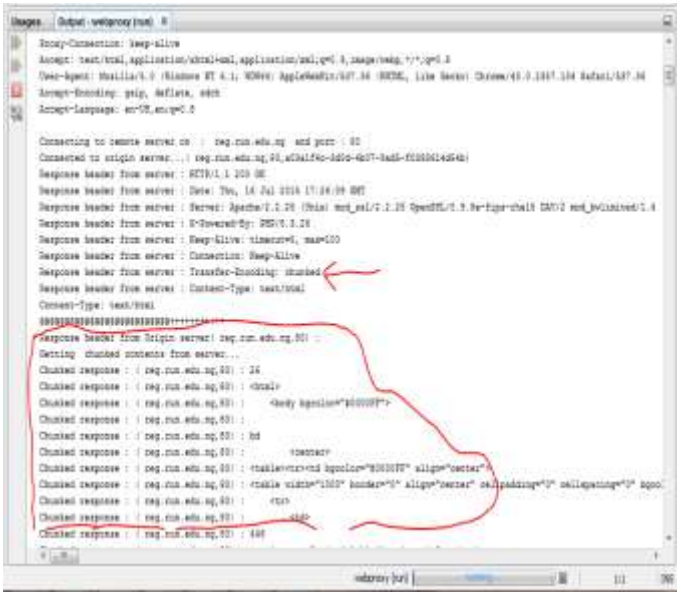

Figure 11. RUN Proxy displaying the chunk process.

this testing is (http://reg.run.edu.ng/FP_growth/FP_growth.php?mat_no=run-300/kunle).

Figure 12 is the portion of the source code that handles the chunked messages. Firstly, the proxy detects if the response message is chunked from the content length received from the origin server. After ascertaining that the response message received from the origin server is chunked, it processes it by reading the input DataStream line by line until a “0” is found which lets the proxy know that all the chunked response has been read. Finally, the proxy prints the chunked response and relays it to the client.

Figure 13 displays the requested resource by a web client on the web browser via the RUN Proxy. The output

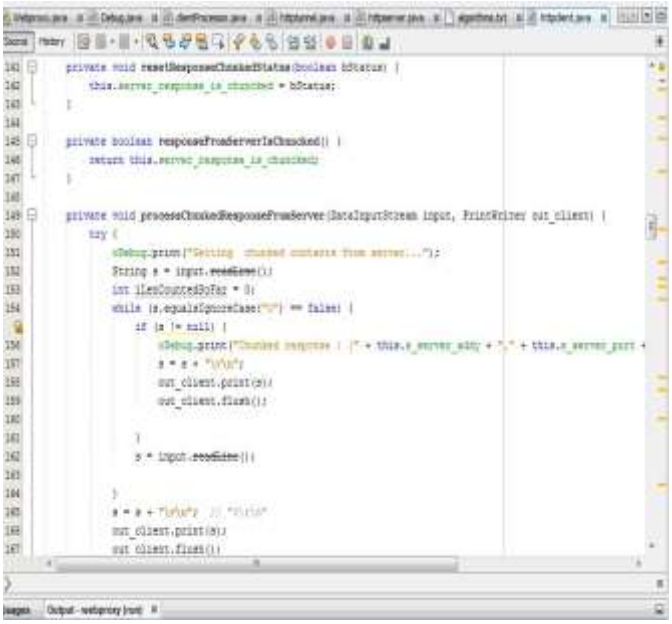

Figure 12. A view of the chunked process source code.

Figure 13. The output result displayed on the web browser.

result received by the client when it request for the URL (http://reg.run.edu.ng/FP_growth/FP_growth.php?mat_no=run-300/kunle) from the Apache server via the proxy. The resource requested by the web browser via the RUN Proxy is displayed on the webpage.

Figure 14 show how the proxy keeps record of its connection with the Origin Server. The Proxy is constructed to keep records of some request-header fields such as the host field which represents the naming authority of the origin server or gateway given by the

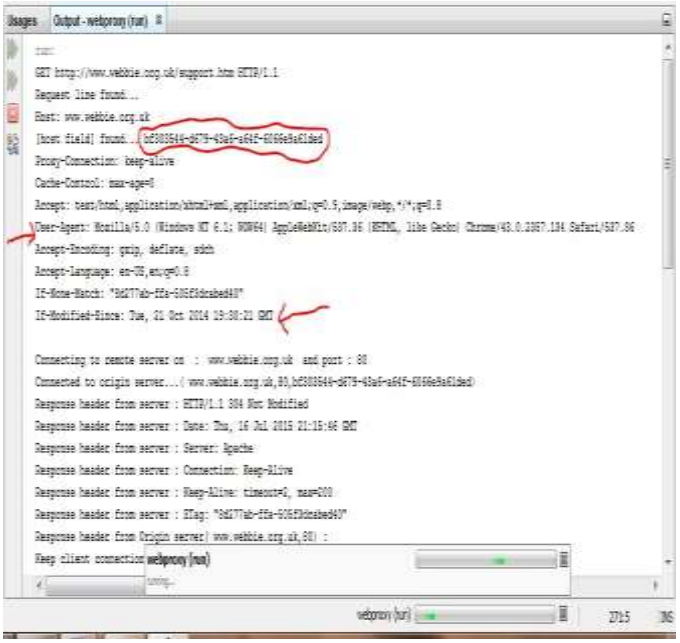

Figure 14. Proxy recording the process between itself and Origin Server.

original URL, the user agent which is the means by which the resource was requested, the If-Modified-Since request-header field, which is used with a method to make it conditional; that is, if the requested resource has not been modified since the time specified in this field, an entity will not be returned from the server and a host of other fields. The pointers show the different header field used by the RUN Proxy.

Figure 15 shows the instance where a socket connection is closed between the requesting client and the responding server. When the socket is closed, it means that the established thread connection created by the proxy to communicate with both the client and the origin server is closed. A socket is an abstraction of client-server communication. The pointer in the Figure 15 implies that the socket connection is closed and the RUN proxy will then continue to listen for other connections.

Conclusion

The use of a proxy should be adapted by organizations and institutions, as this would save lots of time in tracking and monitoring the web activities of their clients. With the RUN Proxy, any client can have access to surf any website with barring policy anytime; so long as it requests the resource via the proxy. This helps to mitigate the loss of vital information provided by origin servers capable of supplying this resource. This study has tackled the problem of providing anonymous communication and that of monitoring clients' web activities. Furthermore, the

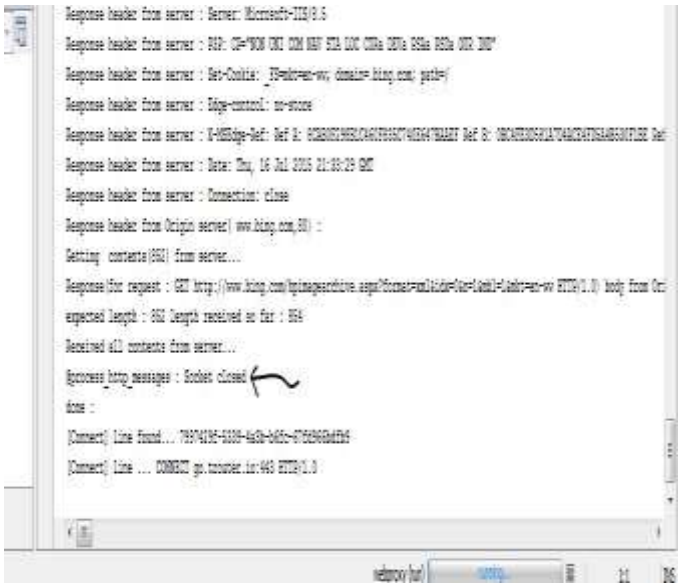

Figure 15. RUN Proxy closing socket connection

benefits of using a proxy can be realized for large populations. These findings, coupled with intuition about the benefits of web proxies to motivate the adoption of anonymous communication in local environments.

The RUN Proxy is able to overcome the major problem with the TOR proxy, in that it can monitor the activities of its users on the Web. This is not possible with TOR as its distributed architecture makes user tracking virtually impossible. In this way, the TOR is often used for illegal activities.

The limitations of this study are that the RUN Proxy is faced with the following constraints as a result of restrictions. These are as follows:

i) The RUN Proxy would not be able to view SSL (Secure Socket Layer) websites;
ii) The RUN Proxy will not be able to effectively track websites with images and picturesque components;
iii) It is also limited by the use of one programming language.

The future direction of this work can be to apply it to increase proficiency in anonymous communications. The incorporation of a system with SSL (Secure Socket Layer) capabilities (that is, a web proxy that is capable of providing resources from SSL websites) and viewing of websites with images can be an important way to improve the system.

Conflict of interest

The authors have not declared any conflict of interest

REFERENCES

Al-Khouri AM (2012). PKI In Government Identity Management Systems. Emirates Identity Authority, Abu Dhabi, United Arab Emirates.

Baentsch M, Baum L, Molter G, Rothkugel S, Sturm P (1997). World Wide Web Caching - The Application level view of the Internet. IEEE Commun. 35:6.

Balachander K, Rexford J (2001). Web protocols and Practice. Addison Wesley.

Bauer K, McCoy D, Grunwald D, Kohno T, Sicker D (2007). Low-Resource Routing Attacks Against Tor. ACM 1-10.

Berthold O, Federrath H, Kpsell S (2001). Web mixes: A system for anonymous and unobservable internet access. In International Workshop on Designing Privacy Enhancing Technologies, pringer-Verlag New York, Inc. pp. 115-129.

Dingledine R, Mathewson N, Syverson P(2004). Tor: the second-generation onion router. In *SSYM'04: Proceedings of the 13th conference on USENIX Security ymposium*, Berkeley, CA, USA, 2004. USENIX Association pp. 21-21.

Edman M, Yener B (2009). On anonymity in an electronic society: A surveyof anonymous communication systems. ACM Computing Surveys pp. 1-35.

Fielding R, Gettys J, Mogul J, Frystyk H, Mansiter L, Leach P, Berners-Lee T (1998). Hypertext Transfer Protocol - HTTP/1.1. HTTP Working Group. Retrieved from http://www.w3.org/Protocols/HTTP/1.1

Gaurav S, Pruthi S (2012). A Review on Various Identity Management Systems. Int. J. Innovative Technol.Exploring Eng. (IJITEE) I(2):113-115.

Gautam N, Petander H, Noel J(2013). Comparison of the cost and energy efficiency of prefetching streaming of mobile video. In Proceedings of the 5thWorkshop on Mobile Video, ACM pp. 7-12.

Gross J, Rosson M (2007). End user concern about security and privacy threats. ACM pp. 167-168.

Jelenkovic RP, Radovanovic A (2009). Asymptotic optimality of the static frequency caching in the presence of correlated requests. Operations Res. Lett. 37(5):307-311.

Li B, Erdin E, Gunes MH, Bebis G, ShipleyT (2011). An analysis ofanonymity technology usage. Springer-Verlag 108-21.

Liska T, Sochor T, Sochorova H (2010). Comparison between normal and TOR-anonymized web client traffic. Procedia Social and Behavioral Sci. pp. 542-546.

Luotonen A, Altis K (1994). World Wide Web Proxies. *First International Conference on the World Wide Web.* Retrieved from http://www.cern.ch/papersWWW94/luotonen.ps

Müller J, Böhm K (2011). *Using Federated Identity Management in a Business-Process-Management System – Requirements, Architecture, and Implementation.* Faculty of Informatics, Karlsruhe Institute of Technology.

NCTU-education S (2007). Retrieved from http://people.cs.nctu.edu.tw: http://people.cs.nctu.edu.tw/~chwong/course/sysadm/slide/Web%20Proxy.pdf

Piatek M (2004). Distributed Web Proxy Caching in a Local Network Environment. Pittsburgh: Department of Mathematics and Computer Science, Duquesne University.

SANS-Institute (2003). Exploring Identtity Management. SANS reading room pp. 1-14.

Shapiro M (1986). Structure and encapsulation in distributed systems: The Proxy Principle. Int. Conf. distributed Computer Syst. pp. 198-204.

Shubina A, Smith S (2003). Using caching for browsing anonymity. SIGecom Exch pp. 11-20.

Sochor T (2012). Anonymization of web client traffic efficiency study. 19th International Conference on Computer Networks.Berlin Heidelberg: Springer Verlag pp. 237-246.

Sochor T(2013). Automatic Control of Configuration of web anonymization. Int. J. New Computer Architectures Appl. (IJNCAA) pp. 1-10.

Squid-Cache WIki (2014). Retrieved from Wiki-Squid: http://wiki.squid-cache.org/WhySquid

Wikipedia (2014). Retrieved from Wikipedia: http://en.m.wikipedia.org/wiki/identity_management_system