Engineering and Technology Journal e-ISSN: 2456-3358
Volume 07 Issue 11 November-2022, Page No.-1624-1629
DOI: 10.47191/etj/v7i11.01, I.F. – 6.484
© 2022, ETJ

An Optimized Cloud Based Proxy Server Techniques to Improve User Experience of Spectrum Utilization

Adeniji Oluwashola David[1], Olatunji Oluwadare Oluwasola[2]
[1,2]Department of Computer Science, University of Ibadan, Ibadan, Nigeria.

ABSTRACT: The global growth and popularity of user experience with the Internet from academia research to multimedia has significantly increased the amount of information online and services available to public domain. This growth has created an unprecedented level of congestion on network infrastructures; scarce network bandwidths, valuable resources which are important and are obstacle to Internet access. The emerging data access can be priced per-byte at high cost, consuming up to 25% of a user's total income, while in developed markets, data usage caps are a persistently zero, requiring users to track and manage consumption to avoid overage fees. In order to address this bandwidth problem, caching algorithm of proxies is employed at network access points to reduce the round trips of client server connections. However, caching is limited during queries of database because most of the content on the web today are dynamic. The focus of the study is to provide data compression techniques for reducing network access latency and to analyse the network traffic based on users' experience.

KEYWORDS: cloud, data compression, optimization algorithm, Latency, bandwidth efficiency

I. INTRODUCTION

The size of the average web page has increased over the years due to the amount of available multimedia information on the internet and this in turn means it takes more network bandwidth and time to load web pages. Likewise, the growth of mobile internet usage is rapidly increasing. In Asia and Africa, 38% of web page views are performed on mobile devices as of May 2014, a year- over-year increase exceeding 10% [1]. In North America, mobile page loads are 19% of total traffic volume with 8% growth yearly.

This high cost of mobile data may support the rapid growth of the Internet access and mobile web browsing especially in developing countries. The HTTP (Hypertext Transfer Protocol) is an application-level protocol for distributed, Intermediary for requests from clients seeking resources from other servers. A client connects to the proxy server, requesting some service, such as a file, collaborative, hypermedia information systems. HTTP is used by web applications because of it request/response protocol. In practice a client sends a request to the server composed of a request method, request-URI (Uniform Resource Identifier), and HTTP version, which is message by header containing request modifiers, client information, and the content. HTTP is a text-based protocol with hundred bytes of header. HTTP is usually transported using TCP/IP. The HTTP 1.1 revision provides a compression data MIME (Multipurpose Internet Mail Extensions) type that provides any client browser that supports HTTP 1.1 to view compressed HTTP data automatically Proxy server software in computer networks is a server (a computer (stem or an application) that acts as a 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. In a research conducted by [2] they summarized how data reduction can be achieved. Most proxies are design to adopt data structure and encapsulation to distributed systems. Web optimization improve different methods of internet data access and transmission. These methods can include content caching and data compression. Web optimization proxy server is an application software that acts as an intermediary between clients(browsers and apps) and remote servers to improve the speed and efficiency of data transmission between the client and the server by the reducing the size of data that needs to be transferred. This paper is organized as follows: Section 2 explains the: cloud and data compression. In section 3 optimization algorithm and bandwidth efficiency while in section 4 discussion of result was provided. The conclusion of paper was made in Section 5.

1. Data Compression

Data compression is explain space–time complexity trade-off of information. If a compression scheme for video may require expensive hardware for the video to be decompressed fast enough to be viewed as it is being decompressed, and the option to decompress the video in full before watching it may be inconvenient or require additional storage. Therefore the design of data compression schemes involves trade-offs among various factors, including the degree of compression, the amount of distortion introduced (when using loss data compression), and the computational resources required to compress and decompress the data. The process of reducing

the size of a data file is often referred to as data compression. In the context of data transmission, it is called source coding; encoding done at the source of the data before it is stored or transmitted.

Computational resources are consumed in the compression process and, usually, in the reversal of the process (decompression). The theoretical background of compression is provided by Information Theory for lossless compression and rate–distortion theory for lossy compression. These areas of study were essentially forged by Claude Shannon, who published fundamental papers on the topic in the late 1940s and early 1950s. Coding theory is also related to this. The idea of data compression is also deeply connected with statistical inference. Information and report from lukew.com shows that: Data from 347 sites, 86% of them delivered the same assets to all devices and 3G connections are 40% slower and 4G connections are 12% slower than the average desktop connection as depicted in fig 1

Fig 1. Graph showing the increasing size of web

For mobile devices, this means that as the page size grows and is not optimized for mobile delivery, overall web page load speed will decline.

2. Optimization Algorithm and Bandwidth Efficiency

The web optimization proxy server can be used for increasing the bandwidth efficiency and improving end user application. The effects of data pricing as adopted in [3], performance in [4] and page layout on user behaviour were evaluated. This review reinforces the motivation behind the web optimization proxy server. Modern cloud infrastructure can also benefit from transport-level performance optimizationsT. Proxy optimizations such as prefetching in [5] was developed for performance analysis..Further information on an intelligent spam-scammer filter mechanism using bayesian techniques [6] was reviewed. Attackes on the user are harmful even to the Networks.Various research has been carried out with the aim of solving detection and preventing such intrusive attacks [7] *The tradeoff between the two protocols can provide a significant impact on the networks.in [8]*. A significant angle in the load control is to limit the work spent by the proxy server to distribute the load and pass on to the main servers[9].*T*he prediction of incoming attacks is achieved in a timely manner which enables security professionals to install defense systems in order to reduce the possibility of such attacks [10].

II. METHODOLOGY

In this study a cloud based optimization server that analysis user experience was developed. The software design is divided into two packages: The proxy server was installed on a cloud server and configured on clients for processing and optimising HTTP requests. This package was developed as a server program using Golang. The web dashboard for displaying daily network traffic and savings statistics was implemented. This package was developed as a web application using Python. A modified version of the software development life cycle was employed. The system analysis phase of the software development life cycle determines how the optimisation system would function. The system design phase involves converting the description of the system at the analysis phase into logical and then physical system specifications. A flowchart is a type of diagram that represents an algorithm, workflow or process, showing the steps as boxes of various kinds, and their order by connecting them with arrows. This diagrammatic representation illustrates a solution model to a given problem. The fig 2 below present the Model flow of optimized server.

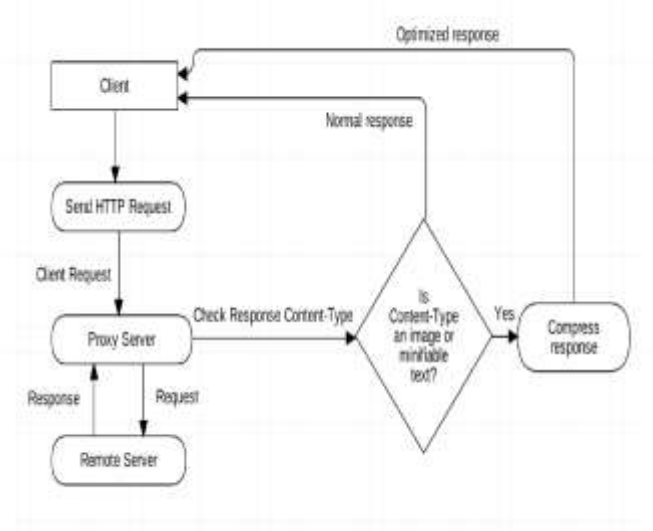

Fig 2. Flow of the optimized server.

Step 1. A Send HTTP Request: The client ends an HTTP request from a browser or mobile application to the proxy server.

Step 2. Get HTTP Response: The remote host sends an HTTP response from the remote server back to the proxy for processing.

Step 3. Optimise HTTP Response: The proxy server reads the Content-Type of the incoming response and determines the appropriate data compression technique to use.

Step 4. Optimise HTTP Response: The proxy server reads the Content-Type of the incoming response and determines the appropriate data compression technique to use.

Step 5.View Bandwidth Analytics: The network administrator can view the web dashboard where the graph of all the daily network traffic is shown.

Step 6. Configure Proxy Server: The network administrator configures a client device or router to forward all HTTP requests to the remote proxy server.

The different techniques employed during the study includes the HTML minification, CSS minification, Javascript minification, JPEG compression and PNG compression. Portable Network Graphics is a raster graphics file format that supports lossless data compression. Generalized Lloyd Algorithm (GLA), which is also called, Linde-Buzo-Gray (LBG) Algorithm is a Vector Quantization algorithm was used in PNG compression. The algorithm uses a mapping function to partition training vectors into N clusters. Socket programming in Golang was created as shown below algorithm:

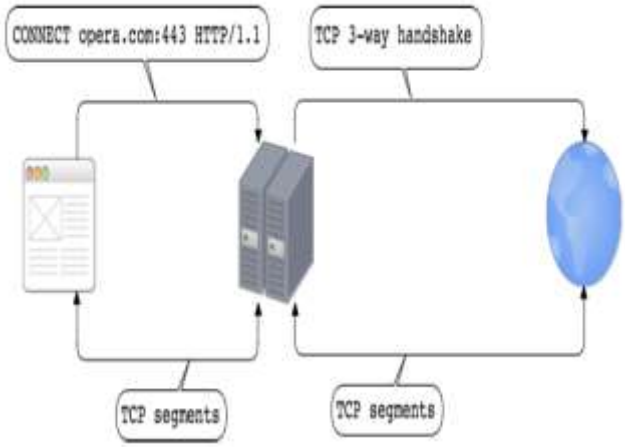

Fig 3. Socket Program process of the optimized server.

Socket programming was used to create a proxy server that receives a request and forwards it then compress the response before sending it back to the client. A proxy server is basically a socket program that acts as an intermediary between clients and servers, it can perform processing or just forward requests downstream as shown in fig 3.

III. RESULTS AND DISCUSSION

To study the impact for various distribution and attributes of the content. The results simulation in the study provide the content type distribution, daily traffic graph, average bandwidth saving and CPU utilization. Configuring the proxy server for use on Macintosh OS X is depicted below in fig 4.

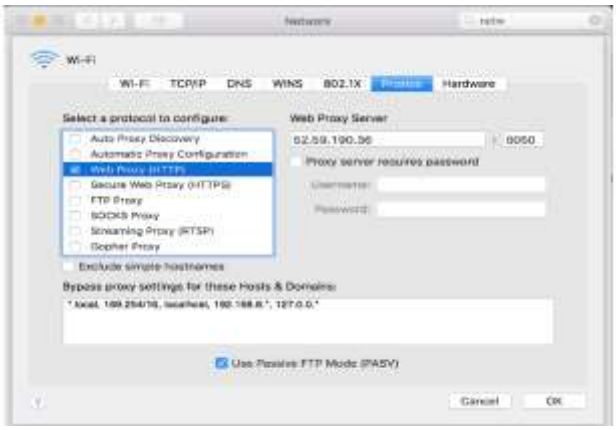

Fig 4. Configuring the proxy server for use on Macintosh OS X.

The databased log during configuration is shown below. Information about the source, destination, path, connection size and the optimized size. It was observed that the correct optimized size was captured for evaluation and the database log is shown below.

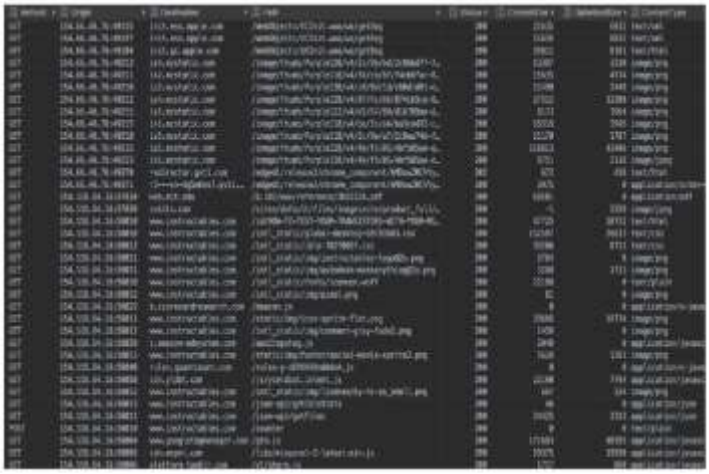

Fig 4.1. Database Screenshot of Network Traffic Logs

The simulation of the Amazon Elastic Compute Cloud (EC2) AWS EC2 Server CPU utilization was monitored for three month per day/weeks and results below was gathered.

Fig 5. AWS EC2 Server CPU utilization

The total requests during the implementation was 86894HTTP request with a total bandwidth of 588424334 bytes at optimized bandwidth of 138877151 in bytes. The optimization ratio was 0.24.This traffic was taken on a daily bases for one month as shown in table6 below.

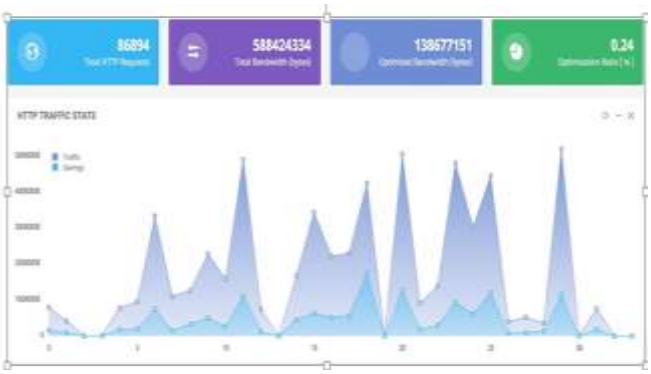

Fig6. Developed Dashboard Monitoring.

A pie chart showing the distribution of the average savings achieved with the system is provided.

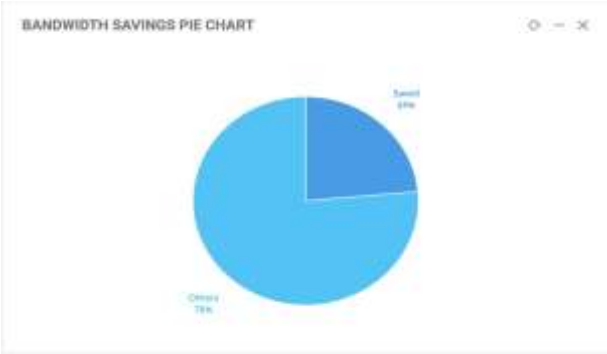

Fig 7. Distribution of the Average Savings.

Based on the unique features of the web protocol and data, the study uses a combination of different data reduction strategies and compression algorithms to reduce the size of web data based on their content type. The table below provide the detail content distribution Type. The characteristic of the distribution ranges from text to images.

Table 1. Percentage Content Distribution

Content Type	Percentage (%)
text/xml	2%
image/svg+xml	1%
image/png	26%
image/jpeg	36%
text/html	35%

IV. CONCLUSIONS

The result was very impressive; about 25% of the total network bandwidth was saved. The result is important because the compression process is completely transparent to the web user and it can work cooperatively with other bandwidth saving mechanisms. Compressed data can be downloaded more quickly, smaller sizes of data means that web pages arrive quicker and take less bandwidth

ACKNOWLEDGMENT

The research was conducted at the Department of Computer Science, University of Ibadan, Nigeria. The authors thank the department for their support in this research work.

REFERENCES

1. Victor Agababov, Michael Buettner, Victor Chudnovsky, Mark Cogan, Ben Greenstein, Shane McDaniel, Michael Piatek, Colin Scott, Matt Welsh, Bolian Yin. 2015. Flywheel: Google's Data

Compression Proxy for the Mobile Web..12th USENIX Symposium on Networked Systems Design and Implementation (NSDI '15) pp 367-380.

2. M. Chetty, R. Banks, A. Brush, J. Donner, and R. Grinter. You're Capped: 2012. Understanding the Effects of Bandwidth Caps on Broadband Use in the Home,.CHI '12: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems pp 3021-3030.
3. A. Patro, S. Rayanchu, M. Griepentrog, Y. Ma, and S. Banerjee. 2013. Capturing Mobile Experience in the Wild: a Tale of Two Apps.,coNEXT 13 Proceeding of the ninth ACM Conference on Emerging networking experiment and technologist pp 199-210.
4. D. Lymberopoulos, O. Riva, K. Strauss, A. Mittal, and A. toulas.) 2012. PocketWeb: Instant Web Browsing for Mobile Devices.ASPLOS XVII: Proceedings of the seventeenth international conference on Architectural Support for Programming Languages and Operating Systems Pp 1–12 .
5. B. Livshits and E. Kiciman.Doloto: Code Splitting for Network- Bound Web 2.0 Application, Conference: Proceedings of the ACM SIGSOFT International Symposium on the Foundations of SoftwareEngineering,2008.
6. Olabisi,A. A., Adeniji,, O. D. Abeng Enangha. 2019. A Comparative Analysis of Latency, Jitter and Bandwidth of IPv6 Packets Using Flow Labels in Open Flow Switch in Software Defined Network‖ Afr. J. MIS, Vol.1, Issue 3, pp. 30-36.
7. Adeniji O. D , Olubukola Adigun, Omowumi O Adeyemo — 2012. An intelligent spam-scammer filter mechanism using bayesian techniques. International Journal of Computer Science and Information Security (IJCSIS), Vol. 10, No. 3 pp 126..
8. Adeniji O. D, Osofisan Adenike, 2020. Route Optimization in MIPv6 Experimental Test bed for Network Mobility:
9. Renuka B. S , Prafulla Shashikiran · G. T. 2020. Model of Load Distribution Between Web Proxy Servers Using Network Traffic Analysis. SN Computer Science .1:103

   Trade-off.International Journal of Computer Science and Information Security (IJCSIS),
10. Adeniji O.d., Olatunji O.O. 2020. "Zero Day Attack Prediction with Parameter Setting Using Bi Direction Recurrent Neural Network in Cyber Security".International Journal of Computer Science and Information Security (IJCSIS), , Vol. 18, No. 3,111-118.