ISSN: 2319-8753

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2014

Proxy Impact on Network Performance Review Study

Amna Hashim Mohammed Ali[1], Dr.AminBabikerAbdelnabi Mustafa[2],

Dr.AshrafGasim AL Seed ABDALLA[3]

Department of Communication, Faculty of Engineering / Al Neelain University, Sudan [1]

Department of Communication, Faculty of Engineering / Al Neelain University, Sudan[2]

Department of Communication, Faculty of Engineering / Future University, Sudan[3]

ABSTRACT:The security and traffic management in computer networks is very important topic for many researchers in the area of network engineering. Proxy as principle baseline element in security and traffic management adds advantages to the network control, reducing the threat and improve the security and network performance. This paper is a review paper discusses many concepts in network security and management which adopted by proxy.

KEYWORDS:UDDI, API, HTTP, XML, SOAP, BPEL, TCP.

I. INTRODUCTION

A proxy server is a computer that offers a computer network service to allow clients to make indirect network connections to other network services. A client connects to the proxy server, and then requests a connection, after connectivity, it will be easy for client to access file, or other resource available on a different server. The proxy provides the resource either by connecting to the specified server or by serving it from a cache. In some cases, the proxy may alter the client's request or the server's response for various purposes, usually to ignore some restrictions polices.Also proxy server may locate between a client application, such as a Web browser, and a real server. It intercepts all requests to the real server to evaluate the fulfil possibility of the requests itself. If the result negative then it will forwards the request to the real server. [1] Proxy server has two mandatory functions improve performance and filter requests .performance. improvement come from proxy capability to save the results of pervious requests for a certain period of time, for example if client request implemented and within the same period of time another client repeat the same request it will be easier for proxy server to achieve saving consuming operation and add significant improvement of performance. Improve performance is clear in application s such as Translation, Accessing services anonymously and Security.

Another function of proxy server is filter requests of the client and classifies it according to its own criteria and according to the network administration policy, since there may be some sites are dedicated for special class of users or may be some sites are prohibited for specific class of user, this feature add more flexibility for network management. Filter request can control Content software, Filtering encrypted data, Bypassing filters and censorship and Logging and eavesdropping. Proxy server type contain A forward proxy and reverse proxy, where forward proxy is an Internet-facing proxy used to retrieve from a wide range of sources (in most cases anywhere on the Internet).A reverse proxy [2]is usually an Internet-facing proxy used as a front-end to control and protect access to a server on a private network. A reverse proxy commonly also performs tasks such as load balancing, authentication, decryption or caching. [3]The main difference between the static and the dynamic proxies is that when static discovery is used, services that can substitute for the monitored services are noted tightly associated with the code of the static proxy. For the dynamic proxy, a look-up mechanism is utilizing to query a service registry at runtime for services that can used to replace failed services. The registry technology used in this case is the Universal Description, Discovery and Integration protocol (UDDI) [3], which is a specification for the publication and discovery of Web services. UDDI specifies a set of data structures, messages and API for creating and maintaining information about Web services in distributed registries. UDDI utilizes existing Web service technologies, such as HTTP, XML, and SOAP.

DOI: 10.15680/IJIRSET.2014.0311002

ISSN: 2319-8753

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2014

II. COMPARATIVE AND INVESTIGATES STUDIES

ADAPTIVE PROXY CACHING FOR WEB SERVERS IN SOFT REAL-TIME APPLICATIONS[4]:-

An adaptive cache proxy developed to improve the performance of web access in soft real-time applicationsIn comparison with the original adaptive TTL algorithm, the modified adaptive TTL has better adaptive performance when the data change rapidly and the user constraint is more critical. With better TTL adaptation, the modified adaptive TTL algorithm has better performance in terms of the combination of temporal coherency and system overheads.

PROXY SERVER EXPERIMENT AND THE CHANGING NATURE OF THE Web[5]:-

They study the behaviour of a proxy server over a six-month period. The result of the study showed that hit ratios of the proxy servers range between 21% to 39% and over 70% of web pages are dynamic. The study indicates the effectiveness of the proxy server and helps evaluate the trade-off between money spent on higher bandwidth lower latency connections, versus the cost/performance of using a caching intermediary.

MODELING TCP PERFORMANCE WITH PROXIES [6]:-

This paper investigates the TCP dynamics and performance over proxies that shorten the TCP feedback loop by segmenting the end-to-end connection. Such proxies are often use to improve TCP performance, e.g., a splitting/spoofing proxy in the satellite communication, and more commonly, a web cache. By analysis, we attempt to develop a basic understanding of the properties of TCP dynamics when such proxies are used, and further obtain certain design principles of systems involving such proxies. We present simple models capturing some features of the proxy performance in both the lossless and lossy scenarios. Due to the complexity involved, detailed analysis is only available in the lossless scenario, and our discussion in the lossy scenario is largely limited to steady state behaviour. They are able to obtain useful insight through such analysis. We identify conditions under which using a proxy provides significant or marginal performance gain by investigating factors including initial window size, congestion level of the proxy, and the level of asymmetry between the links segregated by the proxy. We also discuss how these conditions affect the deployment and provisioning of systems using proxies.

ANALYZING THE NETWORK RESPONSE TIME AND LOAD BALANCING[7]:-

This paper describes the network response time that represent a significant indicator for network performance and how load balancing can improve the network performance.

PROXIES IMPACT IN CLOUD APPLICATIONS [8]:-

A rich cloud system is grows with clouds emerging to provide platforms and services of many shapes and sizes, which guess that future network applications may wish to utilize and combine capabilities from multiple clouds. The problem requires significant data communication that derives from the client server paradigm imposed by most clouds. To address this bottleneck, we propose a cloud proxy network that allows optimized data-centric operations to be perform at strategic network locations, which show the potential of this architecture for accelerating cloud applications.

DOI: 10.15680/IJIRSET.2014.0311002

ISSN: 2319-8753

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2014

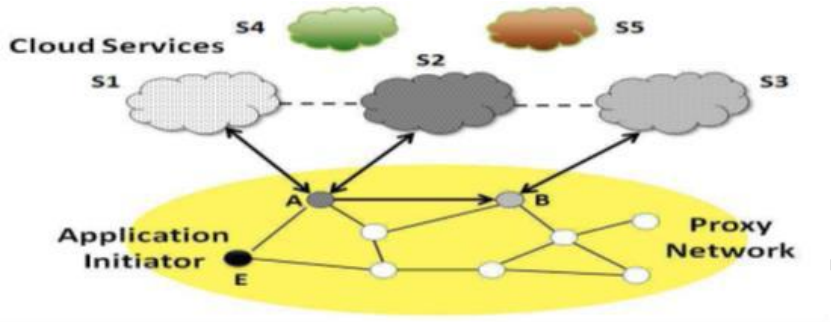


Fig (1): System Model: Si are the cloud services, nodes A and B are proxies, and E is an initiator for an application that uses clouds S1, S2 and S3. Solid arcs represent actual proxy-to-proxy and proxy-cloud interactions, and dotted lines represent logical cloud-to-cloud interactions [8].

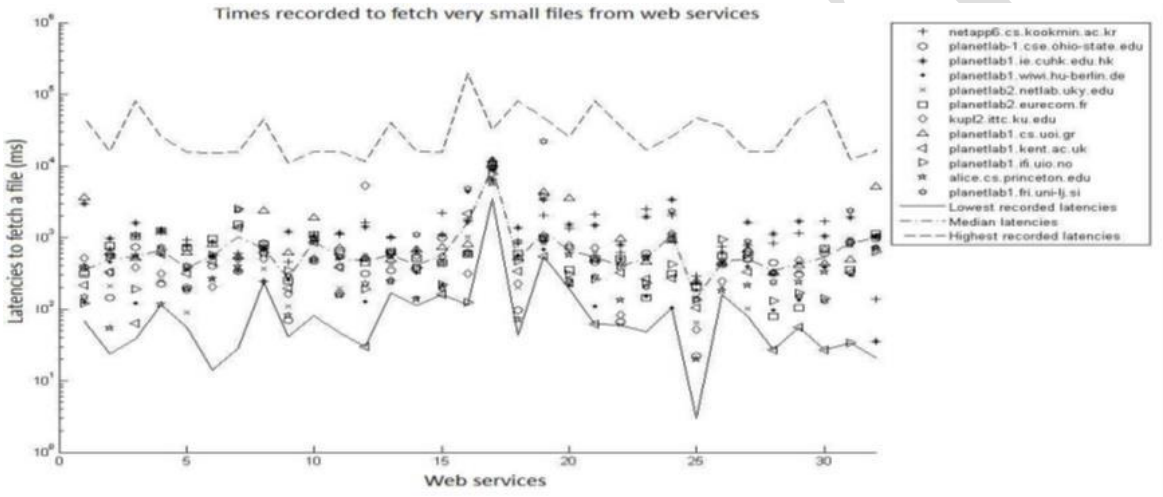


Fig (2): Latency from proxies to different web services. Each proxy is represented by a different shape.[8]

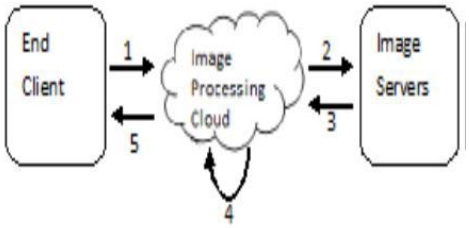


Fig (3-a) Basic flow[8]

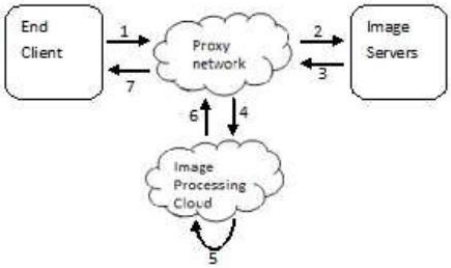


Fig (3-b) Application Workflow[8]

DOI: 10.15680/IJIRSET.2014.0311002

ISSN: 2319-8753

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2014

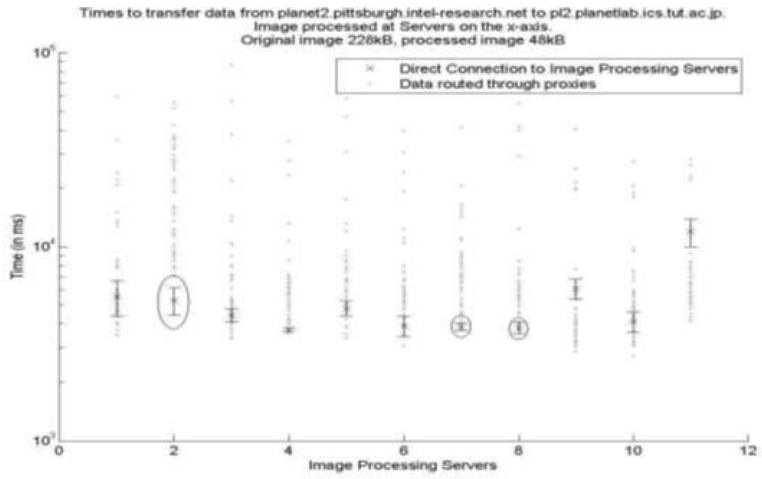


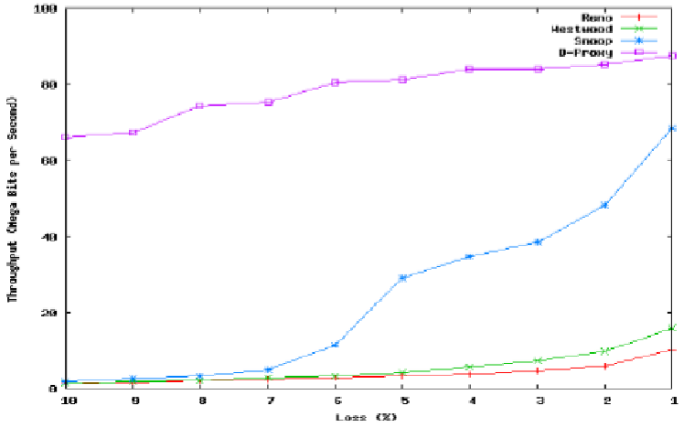
Fig (4): Proxy Acceleration: Pittsburgh End-client, Japan Image server. Each proxy is represented by a light dot and direct path with error[8]

The figures above have discussed the concept of a cloud proxy network, impact in accelerating applications spanning multiple specialized clouds and Latency from proxies to different web services as depicted in Fig (1, 2, 3, and 4). The figures described the broad features of the proxy network and presented specific evidence of its potential to improve the network performance of cloud applications by exploiting its network diversity.

III. PROPOSED MECHANISM STUDIES

DYNAMIC PROXY: RELIABILITY IN WIRELESS NETWORKS [9]:-

Dynamic Proxy is another model of TCP proxy is able to hide losses from wireless TCP. D-Proxy can recover packets, over highly loss links, and can significantly outperform other competing solutions. The scenario used to test D-Proxy was deliberately generic because we believe it to be broadly applicable to a multitude of wireless technologies. D-Proxy is proactive. It analyses the sequence of data frames, recovering and reordering the packets for in order delivery to the TCP receiver. The key benefit is that D Proxy is a negatively acknowledging proxy; sending messages only when a packet is missing. Overall, we believe that the performance of D-Proxy required serious consideration as a potential replacement for ARQ mechanisms. The depicted graph in Fig (5) and Fig (6) below are explain the Throughput of TCP using D-proxy.



Fig(5): Throughput of five TCP flows given losses of 10% to 1%[9]

ISSN: 2319-8753

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2014

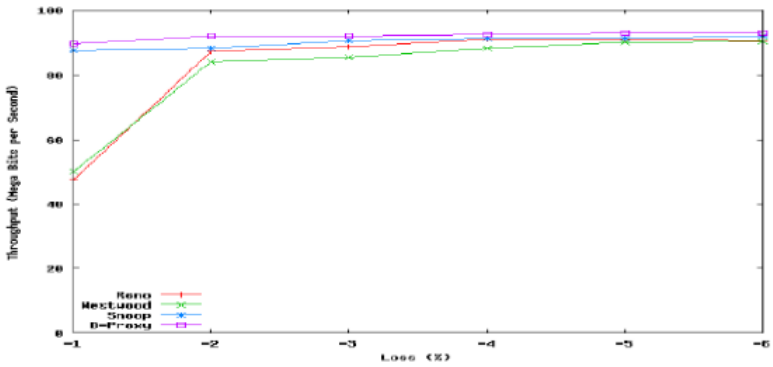


Fig (6): Throughput of five TCP flows given losses of 1x101 to 1x106[9]

A DYNAMIC PROXY ARCHITECTURE FOR VIDEO STREAMING BASED ON OVERLAY NETWORK [10]:-

Dynamic proxy solution based on overlay multicasting network for layered video streaming is presented. In this approach, all clients in an Internet service provider network video stream, which provided by a media server in the Internet, and organize into an overlay tree. The root of the overlay tree acts as a proxy, which receives video stream from original server and distributes it in the local overlay tree.

Demand for video streaming applications over the Internet such as television broadcasts and video on demand is increasing every day. Although multicast is the most effective way of distributing video to large number of users over the Internet, due to lack of a widely acceptable IP multicast service in the Internet, most of the Internet Service Providers (ISPs) use the traditional video proxy servers for video streaming applications. In traditional video proxy solutions, clients directly connect to a dedicated video proxy server in order to receive the required video streams. In this approach, supporting a large number of users for various video streams is inefficient and costly.

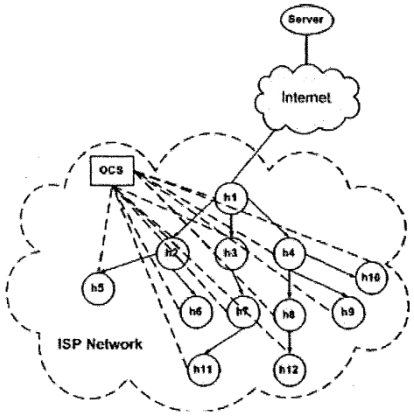


Fig (7): The proposed architecture [10]

MOBILE AND DYNAMIC WEB SERVICES [11]:-

"Making mobile phones capable of consuming Web services over wireless networks is a challenging task because of the different issues to be addressed and the limited resources of mobile devices. In this paper, we focus on the issue of how to perform dynamic discovery and invocation of Web services from mobile phones when a J2ME wireless middleware is used. In order to solve the limitations of the middleware platform when mobile phones act as Web services requestor we propose a Web service based dynamic proxy between service providers and mobile consumers. With this approach, we provide the following features to mobile devices: (1) support of dynamic binding, (2) support of UDDI specification, (3) support of SOAP messages with encoded representation and (4) handling of complex data types. The paper includes the description of the dynamic proxy, implementation and experimental results with the performance of the approachproposed."

ISSN: 2319-8753

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2014

PERFORMANCE OF WEB PROXY CACHING IN HETEROGENEOUS BANDWIDTH ENVIRONMENTS[12]:-

In this paper, they presented results of a detailed simulation study of Web proxy caching in heterogeneous bandwidth environments. We made three main observations. First, latency reductions from caching persistent connections can be much greater than caching data latency. The two types of caching are complementary and should be use together. Second, the bandwidth saved by caching can be more than offset by the bandwidth wasted by aborted connections. It is important to manage aborted connections well. Third, hit ratios can be severely reduce by the growing percentage of documents containing cookies. More study is need into effective ways to cache documents with cookies. An important overall lesson was that low-level details have a significant impact on all aspects of system performance: hit ratios, bandwidth utilization, and user perceived latency.

TRANSPARENT AUTONOMIZATION IN AGGREGATEWEB SERVICES USING DYNAMIC PROXIES [13]:-

They presented an approach to transparently adapting BPEL processes to tolerate run-time and unexpected faults and to improve the performance of overly loaded Web services. They have introduced the dynamic proxy and demonstrated how it is use to encapsulate autonomic behaviour. With the use of a case study, they demonstrated the self-healing and self-optimization behaviour of the dynamic proxy.

NACHE: DESIGN AND IMPLEMENTATION OF A CACHING PROXY FOR NFSV4 [14]:-

In this paper, they present Nache, a caching proxy for NFSv4 that enables a consistent cache of a remote NFS server to be maintain and shared across multiple local NFS clients. Nache leverages the features of NFSv4 to improve the performance of file accesses in a wide area distributed setting by bringing the data closer to the client. Conceptually, Nache acts as an NFS server to the local clients and as an NFS client to the remote server. To provide cache consistency, Nache exploits the read and write delegations support in NFSv4. Nache enables the cache and the delegation to be share among a set of local clients, thereby reducing conflicts and improving performance. They have implemented Nache in the Linux 2.6 kernel. Using File bench workloads and other benchmarks, they present the evaluation of Nache and show that it can reduce the NFS operations at the server by 10-50% [Network File System (NFS)].

USE PROXIES IN TREE TOPOLOGY ARCHITECTURES TO REDUCE THE COMMUNICATION TIME IN MEMBRANES SYSTEMS [15]:-

The use of proxies in the communication of the membranes is extend regardless of the architecture used. This article has proposed a variant of the proxy introduced by Tejedor. With this, new proxy, at each step of evolution, at least three communications between processors are reduced to just one the operation of the proxy, which satisfies a general idea, has a small difference if the parallelism of the Hierarchical Peer-to-Peer Architecture is used. Using this proxy, which connects the dissolved phase with the objects shipping, allowing the membranes begin the rules application phase but not all membranes have completed their communication phase, reducing wait times and improving global times in the evolution of the system.

IV. CONCLUSION

This paper is a review paper discusses many concepts in network security and management which adopted by proxy. Using proxy improve the security and network performance even with a difference type of data and a huge number of user.

REFERENCES

[1] proxy server (Ed), (2014) available at http://www.webopedia.com/TERM/P/proxy_server.htm (accessed 16 October 2014)
[2] Reverse proxy (Ed), (2014) available at Jump up to: a b "Forward and Reverse Proxies". httpdmod_proxy. Apache. Retrieved 20 December 2010. (accessed 16 October 2014)
[3] OnyekaEzenwoye and S. MasoudSadjadi Autonomic Computing Research Laboratory (ACRL) School of Computing and Information Sciences Florida International University 11200 SW 8th St., Miami, FL 33199, 2006
[4] Albert M. K. Cheng Zhubin Zhang Real-Time Systems Laboratory Department of Computer Science University of Houston Houston, TX 77204-3010, USA (cheng, zbzhang)@cs.uh.edu ,2002
[5] Olatunde O. Abiona Member, IEEE, Tricha Anjali Member, IEEE, Clement E. Onime, and Lawrence O. Kehinde,2002

DOI: 10.15680/IJIRSET.2014.0311002

ISSN: 2319-8753

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2014

[6] Mingyan Liu, NavidEhsan Electrical Engineering and Computer Science Department University of Michigan Ann Arbor, MI 48109-2122, US,2001
[7] Mădalina MLAK, Bucharest, Romania, Madalina.Mlak@ie.ase.ro RevistaInformaticaEconomică, nr. 4 (44)/2007
[8] Jon Weissman and SiddharthRamakrishnan Department of Computer Science and Engineering University of Minnesota, Twin Cities 2008.
[9] David Murray Murdoch University D.Murray@murdoch.edu.au , Terry Koziniec Murdoch University T.Koziniec@murdoch.edu.au and Michael Dixon Murdoch University M.Dixon@murdoch.edu.au 2010
[10] B. Akbari, H. R. Rabiee Department of Computer Engineering Sharif University of Technology Tehran, Iran M. Ghanbari Department ofElectrical Engineering University of Essex Colchester, United Kingdom 2005
[11] Elena Sánchez-Nielsen, Sandra Martín-Ruiz, Jorge Rodríguez-Pedrianes Depot. E.I.O. y Computation – Escolar Technical Superior deIngeniería Informatics Universidad de La Laguna, 38271 La Laguna, Spain2007
[12] AnjaFeldmannRam´onC´aceres Fred Douglis Gideon Glass Michael Rabinovich AT&T Labs–Research, Florham Park, NJ, USA,2000
[13] OnyekaEzenwoye and S. MasoudSadjadi Autonomic Computing Research Laboratory (ACRL) School of Computing and Information Sciences Florida International University 11200 SW 8th St., Miami, FL 33199, 2006
[14] Ajay Gulati Rice University gulati@rice.edu ,ManojNaik IBM Almaden Research Center manoj@almaden.ibm.com, RenuTewari IBM Almaden Research Center tewarir@us.ibm.com ,2007
[15] Miguel Ángel Peña, Ángel Castellanos, Alberto Arteta, Francisco Gisbert International Journal "Information Theories and Applications", Vol. 19, Number 3, 2012