ISSN No. 0976-5697

Volume 4, No. 2, Jan-Feb 2013

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Proxy Server Experiment and the Behavior of the Web

Mr.S.V.Gumaste*
Research Scholar, Sant Gadge Baba
Amravati University, Amravati, India
svgumaste@gmail.com

Dr. V. M. Thakare
Professor & Head, Department of CSE
Sant Gadge Baba Amravati University, Amravati, India
vilthakare@yahoo.co.in

Dr. M. U. Kharat
Professor & Head, Department of CSE
MET's Institute of Engineering,
BKC, Nashik, India
mukharat@rediffmail.com

Aditya Patki
Programmer, Department of CSE,
Smt Kamala & Sri Venkappa M Agadi College of
Engineering & Technology, Laxmeshwar, India
Patki.aditya@gmail.com

Abstract: Use of proxy server could help in providing adequate access and response time to large numbers of World Wide Web (WWW) users requesting previously accessed page. The result of the study showed that hit ratios of the proxy servers range from 21% to 39% for large organization (More number of Users) 40% to 70% for medium organizations (medium number of users) and over 70% of web pages are dynamic. The study indicates the effectiveness of the proxy server and helps to evaluate the trade-off between money spent on higher bandwidth lower latency connections, versus the cost/performance of using caching intermediary.

Keywords: caching, hit ratios, Miss, DNS Proxy, performance.

I. INTRODUCTION

A web cache is a mechanism for the temporary storage (caching) of web documents, such as HTML pages and images to reduce bandwidth usage, server load, and perceived lag. A web cache stores copies of documents passing through it; subsequent requests may be satisfied from the cache if certain conditions are met.

Web caches can be used in various systems.[1]

a. A search engine may cache a website.
b. A network-aware forward cache is just caches heavily accessed items.
c. A reverse cache sits in front of one or more Web servers and web applications, accelerating requests from the Internet.
d. A client, such as a web browser, can store web content for reuse. For example, if the back button is pressed, the local cached version of a page may be displayed instead of a new request being sent to the web server.
e. A web proxy sitting between the client and the server can evaluate HTTP headers and choose to store web content.
f. A content delivery network can retain copies of web content at various points throughout a network.

There are two main reasons for use of Web caches:

a. To reduce latency — as the request is satisfied from the cache (which is closer to the client) instead of the origin server, it takes less time for it to get the representation and display it. This makes the Web seem more responsive.
b. To reduce network traffic — as representations are reused, it reduces the amount of bandwidth used by a client. This saves money if the client is paying for traffic, and keeps their bandwidth requirements lower and more manageable.

II. KINDS OF WEB CACHES

A. *Browser Caches:*

Cache setting available in Web browser (like Internet Explorer, Safari or Mozilla), this lets us set aside a section of computer's hard disk to store representations that have seen, just. The browser cache works according to fairly simple rules. It will check to make sure that the representations are fresh, usually once a session (that is, the once in the current invocation of the browser).This cache is especially useful when users hit the "back" button or click a link to see a page that just looked at. Also, if you use the same navigation images throughout your site, they'll be served from browsers' caches almost instantaneously.

B. *Proxy Caches:*

Web proxy caches work on the same principle, but at much larger scale. Proxies serve hundreds or thousands of users in the same way; large corporations and ISPs often set them up on their firewalls, or as standalone devices (also known as intermediaries).Because proxy caches aren't part of the client or the origin server, but instead are out on the network, requests have to be routed to them somehow. One way to do this is to use your browser's proxy setting to manually tell it what proxy to use; another is using interception. Interception proxies have Web requests redirected to them by the underlying network itself, so that clients don't need to be configured for them, or even know about them. Proxy caches are a type of shared cache; rather than just having one person using them, they usually have a large number of users, and because of this they are very good at reducing latency and network traffic. That’s because popular representations are reused number of times.

III. PROXY SERVER

In computer networks, a proxy server is a server (a computer system or an application) that acts as an intermediary for requests from clients seeking resources from other servers [2]. A client connects to the proxy server, requesting some service, such as a file, connection, web page, or other resource available from a different server. The proxy server evaluates the request as a way to simplify and control their complexity. Today, most proxies are web proxies, facilitating access to content on the World Wide Web.

A. Uses:

A Proxy server has a variety of potential purposes, including:

a. To keep machines behind it anonymous, mainly for security.
b. To speed up access to resources (using caching). Web proxies are commonly used to cache web pages from a web server.
c. To prevent downloading the same content multiple times and hence saves bandwidth.
d. To log / audit usage, e.g. to provide company employee Internet usage reporting.
e. To scan transmitted content for malware before delivery.
f. To scan outbound content, e.g., for data loss prevention.
g. Access enhancement/restriction: To apply access policies e.g. blocking of undesired sites,

To by-pass security / parental controls.[3]

Security: the proxy server is an additional layer of defense and can protect against some OS and Webserver specific attacks. However, it does not provide any protection to attacks against the web application or service itself, which is generally considered the larger threat.

Performance Enhancing Proxies`:

A proxy that is designed to mitigate specific link related issues or degradations. PEPs (Performance Enhancing Proxies) are typically used to improve TCP performance in the presence of high Round Trip Times (RTTs) and wireless links with high packet loss. They are also frequently used for highly asynchronous links featuring very different upload and download rates.[4]

B. Filtering:

A content-filtering web proxy server provides administrative control over the content that may be relayed in one or both directions through the proxy. It is commonly used in both commercial and non-commercial organizations (especially schools) to ensure that Internet usage conforms to acceptable use policy. In some cases users can circumvent the proxy, since there are services designed to proxy information from a filtered website through a non-filtered site to allow it through the user's proxy [5].

A content filtering proxy will often support user authentication, to control web access. It also usually produces logs, either to give detailed information about the URLs accessed by specific users, or to monitor bandwidth usage statistics. It may also communicate to daemon-based and/or ICAP-based antivirus software to provide security against virus and other malware by scanning incoming content in real time before it enters the network.

Many work places, schools, and colleges restrict the web sites and online services that are made available in their buildings. This is done either with a specialized proxy, called a content filter (both commercial and free products are available), or by using a cache-extension protocol such as ICAP, that allows plug-in extensions to an open caching architecture.

Assuming the requested URL is acceptable, the content is then fetched by the proxy. At this point a dynamic filter may be applied on the return path. For example, JPEG files could be blocked based on flesh tone matches, or language filters could dynamically detect unwanted language. If the content is rejected then an HTTP fetch error is returned and nothing is cached.

Extranet Publishing: a reverse proxy server facing the Internet can be used to communicate to a firewalled server internal to an organization, providing extranet access to some functions while keeping the servers behind the firewalls. If used in this way, security measures should be considered to protect the rest of your infrastructure in case this server is compromised, as its web application is exposed to attack from the Internet.

Most web filtering companies’ use an internet-wide crawling robot that assesses the likelihood that content is a certain type. The resultant database is then corrected by manual labor based on complaints or known flaws in the content-matching algorithms.

a. ***Caching:*** A caching proxy server accelerates service requests by retrieving content saved from a previous request made by the same client or even other clients. Caching proxies keep local copies of frequently requested resources, allowing large organizations to significantly reduce their upstream bandwidth usage and costs, while significantly increasing performance. Most ISPs and large businesses have a caching proxy. Caching proxies were the first kind of proxy server.

Some poorly implemented caching proxies have had downsides (e.g., an inability to use user authentication). Some problems are described in RFC 3143 (Known HTTP Proxy/Caching Problems). [6]

Another important use of the proxy server is to reduce the hardware cost. An organization may have many systems on the same network or under control of a single server, prohibiting the possibility of an individual connection to the Internet for each system. In such a case, the individual systems can be connected to one proxy server, and the proxy server connected to the main server [6].

b. ***DNS proxy:*** A DNS proxy server takes DNS queries from a (usually local) network and forwards them to an Internet Domain Name Server. It may also cache DNS records. Bypassing filters and censorship

If the destination server filters content based on the origin of the request, the use of a proxy can circumvent this filter. For example, a server using IP-based geo-location to restrict its service to a certain country can be accessed using a proxy located in that country to access the service. Likewise, an incorrectly configured proxy can provide access to a network otherwise isolated from the Internet.

Logging and eavesdropping: Proxies can be installed in order to eavesdrop (Eavesdropping is the act of secretly listening to the private conversation of others without their consent, as defined by Black's Law Dictionary.) upon the data-flow between client machines and the web. All content sent or accessed including passwords submitted, cookies used can be captured and analyzed by the proxy operator. For this reason, passwords to online services (such as webmail and banking) should always be exchanged over a cryptographically secured connection, such as SSL. [7]

C. Squid (Software):[8]

Squid (software) Squid is a proxy server and web cache daemon. It has a wide variety of uses, from speeding up a web server by caching repeated requests; to caching web, DNS and other computer network lookups for a group of people sharing network resources; to aiding security by filtering traffic. Although primarily used for HTTP and FTP, Squid includes limited support for several other protocols including TLS, SSL, Internet Gopher and HTTPS. Squid was originally designed to run on Unix-like systems. Squid is free software and was originally developed by Duane Wessels [9] as the Harvest object cache, part of the Harvest project at the University of Colorado at Boulder. Further work on the program was completed at the University of California, San Diego and funded via two grants from the National Science Foundation .Duane Wessels forked the "last pre-commercial version of Harvest" and renamed it to Squid to avoid confusion with the commercial fork called Cached 2.0, which became NetCache. Web proxy caching is a way to store requested Internet objects (e.g. data like web pages) available via the HTTP, FTP, and Gopher protocols on a system closer to the requesting site. Web browsers can then use the local Squid cache as a proxy HTTP server, reducing access time as well as bandwidth consumption. This is often useful for Internet service providers to increase speed to their customers, and LANs that share an Internet connection. Because it is also a proxy (i.e. it behaves like a client on behalf of the real client), it can provide some anonymity and security. However, it also can introduce significant privacy concerns as it can log a lot of data including URLs requested, the exact date and time, the name and version of the requester's web browser and operating system, and the referrer. A client program (e.g. browser) either has to specify explicitly the proxy server it wants to use (typical for ISP customers), or it could be using a proxy without any extra configuration: “transparent caching”, in which case all outgoing HTTP requests are intercepted by Squid and all responses are cached. The latter is typically a corporate set-up (all clients are on the same LAN) and often introduces the privacy concerns mentioned above.

IV. RELATED WORK

Caching can be applied at several locations namely; the web client, web server and within the network (proxy servers) [3]–[5]. Several studies have reported performance increase due to proxy servers. The result of a study in [3] showed that the average response time of a hit may be five times smaller than a miss. A 20% to 25% improvement in user perceived response time was reported in [6], [7]. Research on the effectiveness of proxy caching is very active. A study at SKSVMACET, Laxmeshwar, (Karnataka, India) has shown that hit rates of 30% to 50% can be achieved by a caching proxy with default values of squid 2.6 stable. Other studies gave a range from between 20% to 60% hit rate [3],[4] reported hit rates of between 10% to 40% for a three level caching hierarchy, and about 35% to 40% for a university-level web proxy cache.

A. Data collection and analysis:

We added Proxy server (Simple Quad core configuration with 4 MB RAM) between Router and Firewall (Cyberoam). Squidversion2.6, stable software we installed and the access logs for our study were collected, DHR computed & shown in Table 1.

A HIT means the domain was found in the cache. A MISS means the domain was not found in the cache. A numeric hit means the supposed domain name was an IP address literal; a negative Hit means the domain was found in the cache, but the record indicated that it doesn't exist.

Suppose a client using a proxy makes requests r1, r2...........rn to pages, if a page has F objects out of which C can be obtained from the cache (Hits) and W from the origin server (Miss). Total request R will be

$$R = \sum_{i=1}^{n} ri,$$

but not all requests will bring back data. Hence, all requests that will result in data transfer will be:

$$F = \sum_{i=1}^{n} Wi + \sum_{i=1}^{n} Ci + \sum_{i=1}^{n} Nui + \sum_{i=1}^{n} Nei$$

Where Nui is numeric Hits and Nei is Negative Hits. (The Authors of [9] Olatunde Oet@ all have neglected Numeric and Negative hits) while calculating, Even by considering them too, we got approximately 69% Hit rate (DHR). So we can compute the Document hit ratio (DHR) as,

$$\sum_{i=1}^{n} Ci / \sum_{i=1}^{n} Wi + \sum_{i=1}^{n} Ci$$

Table 1: Hit and Miss data in Numeric.

Total requests	Hits	Miss	Negative Hit	Numeric Miss	Document Hit Ratio
40	5	31	0	4	0.125
427	213	207	0	207	0.339713
1503	797	494	0	212	0.530273
7770	4195	2287	7	1281	0.539897
60729	37857	20177	95	2600	0.623376
712688	494922	196239	732	20796	0.694443
740485	512704	204569	764	22448	0.692389
858442	594286	233001	853	30302	0.692284

Graphical Analysis of Proxy server is shown in Figure 1 and Figure 2. Figure 1 mainly concentrates with only three readings showing initial miss rate is high as all requests to Proxy are new. As number of requests increases, request for similar pages arise, gradually miss rate is lower that hit rate (Figure 2).

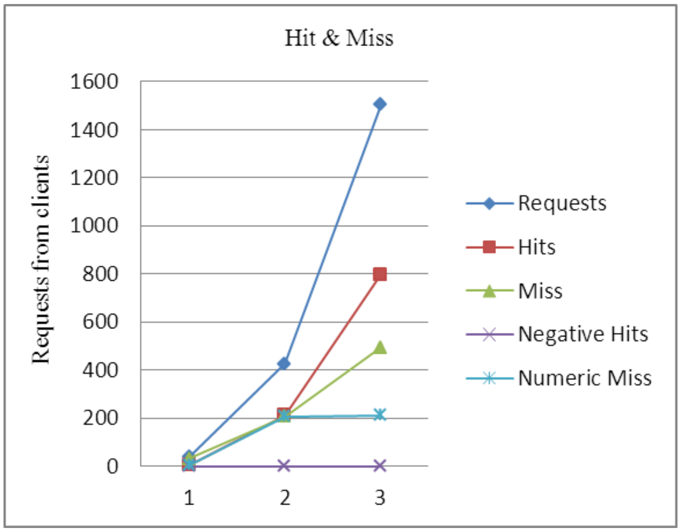

Figure-1. Considering only 3 responses, showing miss is initially high than Hit.

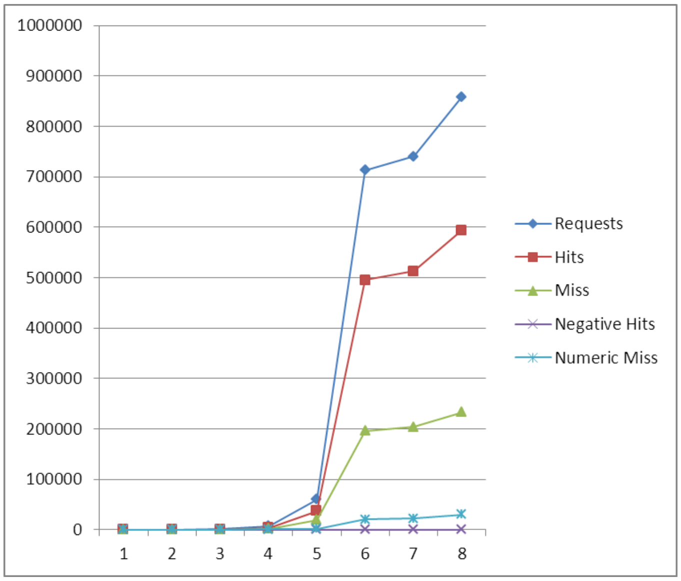

Figure-2. Considering 8 responses, showing miss is decreases and Hit increases as requests increases.

V. CONCLUSION

This paper focuses on testing the performance of proxy servers experimentally and investigates the effect of web behavior on proxy server performance. We carry out an in-depth study of the behavior of a proxy server over a three month period. Access logs of varying durations were collected; we analyzed the logs using cache Manager (Squid). HITS, MISSES along with Negative HITS and Numeric Hits are taken into account. Though Negative Hits are negligible, we took them in computing DHR. Initially HITS are too low but gradually they increased and we achieved 69% HITS with minimum configured system as Proxy. The results of our experiment show the effectiveness of the proxy server and helps evaluate the tradeoff between money spent on higher bandwidth lower latency connections, versus the cost/performance of using a caching intermediary. Future work will focus on other techniques that can be used to improve proxy performance, this include caching and routing.

VI. FUTURE SCOPE

By applying various page replacement policies such as LRU, Belady’s algorithm, Optimal Page replacement policies, NRU, FIFO etc.., proxy behaviors can be analyzed.

VII. REFERENCES

[1]. Flickenger R., K.R. Sreenivasan “How to Accelerate Your Internet, A practical guide to Bandwidth Management and Optimization using Open Source Software” (ISBN: 0-9778093-1-5), Publishers: INASP/ICTP pp. 178 - 191.

[2]. G. Abdulla, E. Fox, M. Abrams, and S. Williams, “www proxy traffic characterization with application to caching”, Technical Report TR-97-03, Computer Science Department, Virginia Tech., March 1997.

[3]. L. DiDio, “Proxy servers gain user appeal,” Computerworld, v31 n16, pp.16 (1) April 21, 1997.

[4]. A. Rousskov, and V. Soloviev, “On performance of caching proxies” Proceedings of the 1998 ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems SIGMETRICS'98/PERFORMANCE '98, Vol. 26, no. 1, pp. 272–273, 1998.

[5]. R. Caceres, F. Douglis, A. Feldmann, G. Glass, and M. Rabinovich, “Web proxy caching: the devil is in the details”, ACM Performance Evaluation Review, vol. 26, no. 1, pp. 11–15, December 1998.

[6]. S. Glassman, “A caching relay for the world wide web”, proc First International World-Wide Web Conference, pp. 69–76, May 1994; also appeared in Computer Networks and ISDN Systems 27, no. 2, 1994.

[7]. Black's Law Dictionary 2nd ed. (St. Paul, Minn.: West Publishing, 1910) ISBN 1-886363-10-2

[8]. Kulbir Saini, “Squid Proxy Server – Beginner’s Guide” ISBN 978-1-849513-90-6, Packet Publishing

[9]. Olatunde O. Abiona (Member, IEEE), Tricha Anjali (Member, IEEE), Clement E. Onime, and Lawrence O. Kehinde ”Proxy Server Experiment and the Changing Nature of the Web” 2008 IEEE.