SECURE WEB BROWSING OVER LONG-DELAY BROADBAND NETWORKS

Recommendations for Web Browsers

Doug Dillon
Assistant Vice President, Software Engineering, Hughes Network Systems
11717 Exploration Lane, Germantown, MD 20876, USA

Gurjit Singh Butalia
Hughes Software Systems, Electronic City, Plot 31, Sector 1
Gurgoan -122015, Haryana, India

Pawan Kumar Joshi
Hughes Sortware Systems, 27 Gandhi Sadan
Mandir Marg, New Delhi -110001, India

Keywords: Satellite Broadband, Secure Web Browsing, Performance Analysis, HTTPS, Web Browser, SSL, TLS

Abstract: Current browser implementations provide less than desirable secure web page response time over geosynchronous satellite and other long delay broadband networks (e.g., intercontinental access across the Internet). This document defines the issues and recommends a set of enhancements that improve response time without compromising security. These enhancements are shown, by analysis, to provide more than a 50% response time reduction for a typical secure web page.

1 INTRODUCTION

Security is important on the Internet. Secure Web Browsing, in the form of the HTTP protocol running over an Secure Sockets Layer (SSL) transport (A. Frier, 1996) has proved to be the key enabling technology for E-Commerce on the Internet (Thomas, 2000) and is becoming the preferred method for secure remote access to enterprise Intranets. The SSL protocol was introduced by Netscape and has been standardized by the Internet Engineering Task Force (IETF) under the name Transport Layer Security (TLS) (Jungmaier, 2002).

The security provided by use of the HTTP protocol running over SSL transport (HTTP/SSL) comes at the cost of reduced performance. Most research and commercial product development aimed at reducing the performance impact has been focused on reducing web server processing requirements (Apostolopoulos, 2000).

The response time performance cost of HTTP/SSL has not received similar scrutiny. This paper demonstrates how current browser implementations of HTTP/SSL amplify the latency inherent in broadband networks, how this impact is felt for secure web page retrieval across networks employing either transcontinental or satellite links and provides recommendations for reducing the response time impact.

2 LONG-DELAY NETWORKS

Table 1 shows typical round-trip times over various broadband networks as measured by the authors. Consumer and enterprise satellite networks, such as the Hughes Network Systems Inc. DIRECWAY® service and the Starband consumer Internet access service, utilize demand assignment to increase the effective capacity of a satellite transponder. This introduces a second satellite round trip, which results in an overall typical round-trip time of 1300 msec.

As can be seen from the table, intercontinental Internet access round-trip time is often an order of magnitude higher than intercity, round-trip time even when no satellite links are involved.

For each of these networks, an end-user transaction that involves a single round trip provides acceptable response time for most applications. Table 1:Measured Round-Trip Times Over Broadband Networks

Dillon D., Singh Butalia G. and Kumar Joshi P. (2004).
SECURE WEB BROWSING OVER LONG-DELAY BROADBAND NETWORKS - Recommendations for Web Browsers.
In *Proceedings of the First International Conference on E-Business and Telecommunication Networks*, pages 160-168
DOI: 10.5220/0001391901600168

Table 1:Measured Round-Trip Times Over Broadband Networks

Network Type	Ping Response Time (Sec)
Fixed Assignment Satellite Network	.65
Demand Assigned Satellite Network	1.3
East Coast USA to India via Internet	.30
East Coast USA to Moscow via Internet	.18
Washington DC to New York	.03
Local Area Network	.001

The sections that follow analyze the number of round-trips required to retrieve a typical secure web page.

3 HTTP/SSL TRANSACTIONS

Apart from any optimizations, the HTTP or Secure Sockets Layer (HTTPS) protocol used for secure web browsing allows a web browser to retrieve a URL after four or five round-trip transactions as illustrated in figure 1. DNS lookup is required for the first retrieval of a URL from a website and, depending on where the DNS server is located within the network and the contents of its cache, may not require the response time impact of a full round trip.

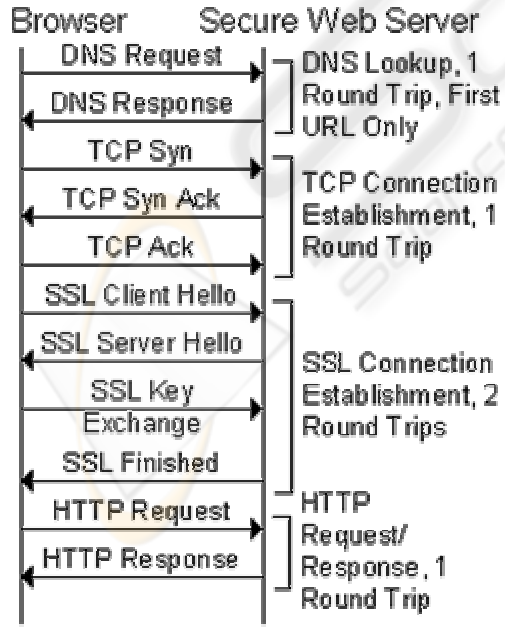

Figure 1: Unoptimized HTTP/SSL Transaction

As illustrated by figure 2, with the SSL Session Reuse optimization (aka, session resumption), HTTPS retrieval of a URL is reduced to three round-trip transactions. A DNS lookup is not typically performed with Session Reuse as the site name was resolved when the first SSL connection to the server was established.

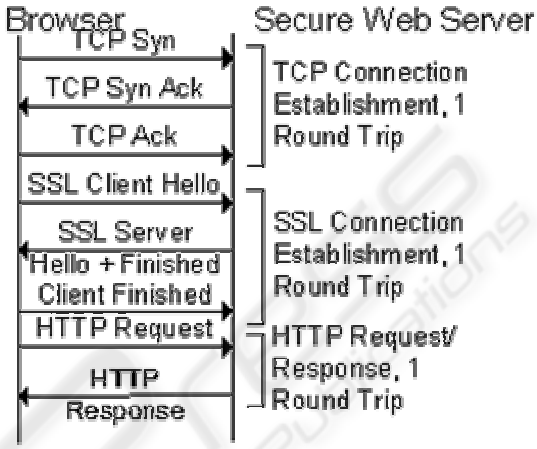

Figure 2: SSL Session Reuse

As illustrated by figure 3, with the use of HTTP persistent connections, HTTPS retrieval of a URL is reduced to one round-trip transaction for transactions that make use of a previously SSL established connection. The use of persistent connections is not possible when the server does not support persistent connection or when the server sends back an HTTP response entity body with neither a CONTENT-LENGTH field nor chunked encoding.

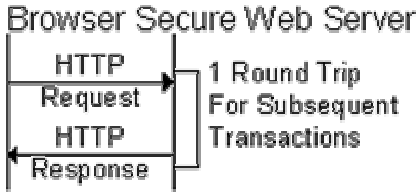

Figure 3: Persistent HTTP/SSL Connection

4 TYPICAL SECURE WEB PAGE

Web pages become more complex as time goes by. A typical secure web page consists of multiple URLs of various kinds. This section defines a typical web page as it currently exists and then analyzes its response time. The example web page analyzed in this subsection consists of the URLs described in table 2.

Table 2: Typical Secure Web Page

Num URLs	Description Of URL
1	HTML web page referencing two HTML frames
2	HTML page for each of the two frames
2	Cascading Style Sheet For Each HTML Frame
2	Background URL Referenced By Each Frame's Cascading Style Sheet
2	Javascript File 1 For An HTML Frame
2	Javascript File 2 For An HTML Frame
2	Redirection (301 or 302 response) to an image on another server
2	Redirected embedded images on the other server
14	Seven Embedded .gifs each HTML frame
29	**Total number of URLs**

5 PARALLELISM AND ITS LIMITS

Browsers utilize parallel HTTPS connections to retrieve URLs in parallel to reduce web page response time. For the analysis in this subsection we assume the browser is configured to support up to 16 parallel connections to each secure web server currently being accessed by the browser. Although having parallel connections make possible retrieval of URLs in parallel, other factors frequently limit the parallelism achieved. This section introduces those factors and quantifies their input.

This document refers to a “wave” of URL retrievals as a set of URLs that can be retrieved in parallel. For example, a simple web page consisting of just an HTML page and a set of embedded images may be retrieved by a browser in two waves, one for retrieving the HTML and one for retrieving the embedded images after the HTML has been parsed. This concept of a wave is an oversimplification in that it does not consider parallelism that is lost when the number of available URLs to be retrieved in parallel exceeds browser limitations on the number of parallel

Various aspects of web page design limit the achievable parallelism and increase the number of waves required to retrieve a web page. These aspects include:

- HTML Frames - the use of HTML frames limits parallelism in that a browser cannot begin to retrieve URLs referenced by an HTML frame until after that frame has been retrieved and parsed. A web page that utilizes HTML frames will always require at least three waves of URL retrievals (one for the base HTML page, one for the HTML frames, one for the items referenced by the HTML frames).
- Javascript And Cascading Style Sheets (CSS) - when a browser is parsing an HTML web page and finds a reference to a javascript URL or a Cascading Style Sheet (CSS), it typically stops parsing the HTML and retrieves the referenced URL. It resumes parsing the HTML after it has completed parsing the referenced javascript or CSS URL. This is because it is possible for the javascript or CSS file to change the way the rest of the HTML is parsed. This has the effect of limiting the parallel retrieval of URLs in that URLs referenced by the HTML cannot be retrieved until after they have been parsed and parsing is suspended while the javascript or CSS URL is being retrieved. When a browser behaves this way, a web page with N javascript and cascading stylesheet URLs may be retrieved in N + 2 waves, in the best case. This involves one wave for the HTML retrieval, one wave for each of the javascript and cascading stylesheet URLs, and one wave for embedded image retrieval.
- Javascript And CSS Referenced URLs - both Javascript and Cascading style sheets may reference URLs that are required to paint the web page. When this occurs, such a URL cannot be retrieved until after the referencing javascript or cascading style sheet has been retrieved.

Table 3: Round-Trip Time Impact on Secure Web Page Response Time

Network Type	Ping Response Time (Sec)	Typical Secure Web Page Response Time (Sec)	Percentage of Response Time Due to Round Trips With .25 Sec Server Response Time
Fixed Assignment Satellite Network	0.650	10.4	98%
Demand Assigned Satellite Network	1.300	20.8	99%
East Coast USA to India via Internet	0.300	4.8	95%
East Coast USA to Moscow via Internet	0.180	2.88	92%
Washington DC to New York	0.030	0.48	66%
Local Area Network	0.001	0.016	6%

- Redirections - Often an embedded image or even an HTML page is actually available from a different web server than the one from which it is originally requested. A web server may instruct the browser to retrieve a URL from another web page with a redirection. A redirection may be accomplished either with an HTTP 301 or 302 response or with HTML. The use of redirection increases response time by requiring URL retrieval to obtain the redirection followed by another URL retrieval to actually obtain the URL.

6 TYPICAL WEB PAGE RESPONSE ANALYSIS

A simple table-based simulation of a web browser (see Table 5) reveals that the typical secure web page discussed is retrieved in no less than 16 round-trips even under the following set of reasonably optimistic assumptions:

• Broadband connection – i.e. the time to actually transmit packets and browser and server processing time are negligible compared to the response time contributed by the number of round trips required to paint the page.

• The browser permits up to a maximum of 16 simultaneous HTTPS connections to a given SSL server. The default value for this is for most browsers is 4 connections to HTTP 1.0 and 2 connections for HTTP 1.1. Assuming this is optimistic in that changing this configuration is for experts only in that it involves editing the windows registry, javscript files or something similarly for experts only.

• Use of persistent connections. This is enabled by default for most modern browsers and is supported in many cases by secure web servers.

• No persistent connections exist when the web page retrieval begins.

• Use of SSL session reuse. This is enabled by default for most modern browsers and servers.

• The DNS lookup for the main server does not cost a round-trip, although a DNS lookup for a redirected image's server does cost a round trip.

• Once a URL is assigned to a connection, it uses that connection even if it has to wait for the connection to be established.

• No HTTP pipelining. This paper's analysis of web page response time does not assume the use of HTTP pipelining because no popular browser has this enabled by default. This is probably due to flaws in the pipelining protocol design and server implementations thereof.

As can be seen in table 3, while round-trip time is a major contributor to response time for intercity broadband connectivity (much less so for LAN connectivity), it completely dominates intercontinental Internet and satellite network response time. This problem is significant enough for non-satellite networks that at least one Internet Startup, www.netli.com, is introducing a global distributed caching solution that advertises 1 sec secure web browsing to enterprise Intranet servers.

7 BROWSER RECOMMENDATIONS

Each of the following recommendations mitigates an inefficiency that applies especially to secure web browsing. The recommendations are as follows:

1. Connection Pooling – reduces the response time impact of SSL connection establishment whether the server supports persistent connections or not. Connection pooling establishes and maintains a pool of connections with a secure server so that an established connection can be allocated to the retrieval of a URL as soon as the need to retrieve it is determined. The recommended connection pooling maintains a historical record of the number

Table 4: Improved Response Time for First and Repeat Retrievals

Network Type	Ping Response Time (Sec)	Unoptimized Typical Secure Web Page Response Time (Sec)	Optimized Typical Web Page Response Time First Retrieval	Optimized Typical Web Page Response Time Repeat Retrieval
Fixed Assignment Satellite Network	0.650	10.4	7.80	4.55
Demand Assigned Satellite Network	1.300	20.8	15.60	9.10
East Coast USA to India via Internet	0.300	4.8	3.60	2.10
East Coast USA to Moscow via Internet	0.180	2.88	2.16	1.26
Washington DC to New York	0.030	0.48	0.36	0.21
Local Area Network	0.001	0.016	0.01	0.01
Response Time Reduction			**25%**	**56%**

of simultaneous connections actually utilized for previous visits to a secure site and trims the connection pool size accordingly.

2. Historical Prefetch – reduces the overall response time for repeat visits to a web page by initiating the retrieval of URLs sooner than permitted by unoptimized web browsers. Historical prefetch leverages the fact that users tend to visit the same secure web pages repeatedly (for example, once a day to check a bank or brokerage account). Historical prefetch maintains a persistent cache with entries for HTML URLs. A cache entry identifies the URLs that have in the past been consistently retrieved immediately after the HTML URL was retrieved. When the retrieval of a URL in the cache commences, historical prefetch immediately initiates the retrieval of those URLs that historically were consistently retrieved immediately after the web page.

3. Quick Parse – an unoptimized browser suspends the parsing of HTML and retrieval of embedded URLs (cascading style sheets, javascript URLs, and embedded images) when a cascading stylesheet or javascript file is referenced until that URL can be retrieved and parsed. This delays the retrieval of subsequent URLs. Quick Parse quickly parses HTML to determine the set of URLs that will needed without waiting for cascading style sheets and javascript files to be loaded and parsed one at a time.

8 RECOMMENDATION RESPONSE TIME ANALYSIS

Using a tabular simulation of a browser incorporating the above recommendations (see Tables 6 and 7), allows the number of round-trips for an initial visit to the typical web page from 16 to 12. Subsequent visits to the same page require only 7 round-trips. Table 4 summarizes the impact of this reduction in round-trips for various networks.

9 CONCLUSIONS

This paper presents the reasons why current browser implementations require a large number (e.g. sixteen) network round-trips to retrieve a typical, modern secure web page. The paper provides experimental data demonstrating that these round-trip times dominate secure web page response time over transcontinental and satellite broadband networks. The paper provides recommendations to browser implementers that allow the number of round trips to be significantly reduced, especially for pages that a user repeatedly visits.

REFERENCES

Frier, A., Karton, P. and Kocher, P., 1996. *"The SSL 3.0 Protocol,"* Netscape, Nov 1996.

Thomas, Stephen A., 2000, *"SSL & TLS Essentials: Securing the Web"*, John Wiley & Sons

A. Jungmaier, E. Rescorla, M. Tuexen. , 2002, "*RFC 3436 Transport Layer Security over Stream Control Transmission"*, www.ietf.org

APPENDIX TABULAR BROWSER SIMMULATIONS

Tables 5, 6 and 7 provide the tabular simulations of browser retrieval of secure web pages without incorporating this paper's recommendations, for a first retrieval of a page when incorporating the recommendations and for a subsequent retrieval of the web page.

Table 5: Round-Trip Analysis of Typical Web Page Retrieval

Event ID	Predecessor Event IDs	HTTPS Conn ID	Total Round Trips At End Of This Event	Event Description
1		1	1	TCP connection establishment
2	1	1	3	SSL connection establishment
3	2	1	4	Retrieval of HTML web page
4	3	1	5	Retrieval of 1st HTML frame
5	3	2	5	TCP connection establishment for 2nd HTML frame
6	4	1	6	Retrieval of 1st frame's cascading style sheet
7	5	2	6	SSL connection establishment with session reuse for 2nd HTML frame
8	6	1	7	Retrieval of 1st frame's background URL
9	7	2	7	Retrieval of 2nd HTML frame
10	6	3	7	TCP connection establishment for 1st frame's 1st javascript URL
11	8	1	8	Retrieval of 2nd frame's cascading style sheet
12	10	3	8	SSL connection establishment with session reuse for 1st frame's 1st javascript URL
13	11	1	9	Retrieval of 2nd frame's background URL
14	11	2	9	Retrieval of 2nd frame's first javascript URL
15	12	3	9	Retrieval of 1st frame's 1st javascript URL
16	13,14	1	10	Retrieval of 2nd frame's 2nd javascript URL
17	14,15	2	10	Retrieval of 1st frame's 2nd javascript URL
18	16,17	1	11	Retrieval of 1st frame's redirection to an image on another server
19	16,17	2	11	Retrieval of 2nd frame's redirection to an image on another server
20	15,17	3	11	Retrieval of 1st frame's 1st embedded image
21	17	4..9	11	TCP connection establishment 1st frame's 2nd through 7th images
22	16	10..16	11	TCP connection establishment for the 2nd frame's seven embedded images
23	21	4..9	12	SSL connection establishment with session reuse for 1st frame's 2nd through 7th images

24	22	10..16	12	SSL connection establishment with session reuse for 2nd frame's seven embedded images
25	18	17	12	DNS lookup to other server for 1st frame's redirected image
26	19	18	12	DNS lookup to other server for 2nd frame's redirected image
27	23	4..9	13	Retrieval of 2nd through 7th embedded images
28	24	10..16	13	Retrieval of 2nd frame's seven embedded images
29	25	17	13	TCP connection establishment to other server for 1st frame's redirected image
30	26	18	13	TCP connection establishment to other server for 2nd frame's redirected image
31	29	17	15	SSL connection establishment without session reuse to the other server for retrieval of the 1st frame's redirected image
32	30	18	15	SSL connection establishment without session reuse to the other server for retrieval of the 1st frame's redirected image
33	31	17	16	Retrieval of 1st frame's redirected image
34	32	18	16	Retrieval of 2nd frame's redirected image
			16	**Grand total of 16 round trips to paint page**

Table 6: Optimized First Retrieval Web Page Response Time

Event ID	**Predecessor Event IDs**	**HTTPS Conn ID**	**Total Round Trips at End of this Event**	**Event Description**
1		1	1	TCP connection establishment
2		1	3	SSL connection establishment
3	2	2..4	4	Pooled TCP connection establishment
4	2	1	4	Retrieval of HTML web page
5	4	1	5	Retrieval of 1st HTML frame
6	3	2..4	5	Pooled SSL connection establishment with session reuse
7	4	5..16	5	Pooled TCP connection establishment triggered by need for more than one connection
8	6	1	6	Retrieval of 1st frame's cascading style sheet
9	6, 4	2	6	Retrieval of 2nd HTML frame
10	6, 4	3..4	6	Retrieval of 1st frame's 1st and 2nd javascript URLs
11	7	5..16	6	Pooled SSL connection establishment with session reuse
12	8, 9	1	7	Retrieval of 2nd frame's cascading style sheet
13	9, 10	2..3	7	Retrieval of 2nd frame's 1st and 2nd javascript URLs
14	10, 5	4	7	Retrieval of 1st frame's redirection to image on other server
15	11, 9	5	7	Retrieval of 2nd frame's redirection to image on other server

16	11, 5	6..12	7	Retrieval of 1st frame's embedded images
17	11, 9	13..16	7	Retrieval on 2nd frame's 1st through 4th embedded images
18	12, 13, 5	1..3	8	Retrieval of 2nd frame's 5th through 7th embedded images
19	14, 8	4	8	Retrieval of 1st frame's background URL
20	14	17	8	DNS lookup for retrieval of 1st frame's redirected image
21	15	18	8	DNS lookup for retrieval of 2nd frame's redirected image
22	20	17	9	TCP connection establishment for retrieval of 1st frame's redirected image.
23	21	18	9	TCP connection establishment to other server to retrieve 2nd frame's redirected image
24	22	17	11	SSL connection establishment for retrieval of 1st frame's redirected image
25	23	18	11	SSL connection establishment to other server with session reuse to retrieve 2nd frame's redirected image
26	24	17	12	Retrieval of 1st frame's redirected image
27	25	18	12	Retrieval of 2nd frame's redirected image
			12	**Grand total of 12 round trips to paint page**

Table 7: Optimized Web Page Repeat Retrieval

Event ID	**Predecessor Event IDs**	**HTTPS Conn ID**	**Total Round Trips at End of this Event**	**Event Description**
1		1	1	TCP connection establishment
2		2	1	DNS lookup to retrieve 1st frame's redirected image on other server
3		3	1	DNS lookup to retrieve 2nd frame's redirected image on other server
4	2	2	2	TCP connection establishment retrieve to 1st frame's redirected image on other server
5	3	3	2	TCP connection establishment retrieve to 1st frame's redirected image on other server
6	1	1	3	SSL connection establishment
7	6	1	4	Retrieval of HTML web page
8	4	2	4	SSL connections establishment to retrieve 1st frame's redirected image on other server.
9	5	3	4	SSL connections establishment to retrieve 1st frame's redirected image on other server.
10	2	4..18	4	Pooled TCP connection establishment triggered by need for more than one connection which is triggered by historical prefetches being queued up
11	7	1	5	Retrieval of 1st HTML frame

12	8	2	5	Retrieval of 1st frame's redirected image
13	9	3	5	Retrieval of 2nd frame's redirected image
14	10	4..18	5	Pooled SSL connection establishment with session reuse
15	11, 14	1, 4..7	6	Retrieval of 1st frame's cascading style sheet, 1st javascript, 2nd javascript, background URL and redirection to image on another server
16	14	8..13	6	Retrieval of 2nd HTML frame, 2nd frame's cascading style sheet, 1st javascript, 2nd javascript, background URL and redirection to image on another server
17	14	14..18	6	Retrieval of 1st frame's 1st through 5th embedded images
18	15	1, 4	7	Retrieval of 1st frame's 6th and 7th embedded images
19	15, 16	5..11	7	Retrieval of 2nd frame's seven embedded images
			7	**Grand total of 7 round trips to paint page**