Comparing HTTP and HTTPS

April 15, 2015

Team Leader: Tyler Sedam
Members: Alex Sedam and Jebreen Aljebreen

COM-309
Instructor: Dr. Van Nguyen

Abstract

The purpose of this document is to provide a brief explanation of what HTTP and HTTPS are, and to explain how, why, and when they were developed. Additionally, this document serves to outline how each protocol, and their subprotocols, function, as well as to highlight the differences between HTTP and HTTPS. The differences presented and discussed will serve to show that the usage of HTTPS, having an extra layer of protocols and subprotocols to provide security, functions as effectively and efficiently as HTTP with the added security. Therefore, the goal is to provide a detailed and well researched document to exhibit the benefits of using HTTPS over HTTP.

Table of Contents

1 Introduction....4
2 HTTP....4
3 HTTPS....5
3.1 History of HTTPS....5
3.2 SSL....6
3.2.1 History of SSL....6
3.3 TLS....7
4 HTTP vs HTTPS....8
Conclusion....10
Appendix A References....11

1 Introduction

The Hypertext Transfer Protocol, or HTTP, and the more secure HTTPS, pervades and controls the dissemination of hypertext documents and applications across the Internet. The most widely used sites such as Facebook, Google, and YouTube, employ HTTP or HTTPS to display or send text, videos, images, audio clips, and other multimedia-related services and components. A web browser, the program designed to access the Internet, is designed to search for, interpret, and display such information. The majority of activities that each of us do on the Internet largely involve using HTTP or HTTPS-based content.

The purpose of this document is to provide a brief explanation of what HTTP and HTTPS are, and to explain how, why, and when they were developed. Additionally, this document serves to outline how each protocol, and their subprotocols, function, as well as to highlight the differences between HTTP and HTTPS. The differences presented and discussed will serve to show that the usage of HTTPS, having an extra layer of protocols and subprotocols to provide security, functions as effectively and efficiently as HTTP with the added security. Therefore, the goal is to provide a detailed and well researched document to exhibit the benefits of using HTTPS over HTTP.

2 HTTP

Sir Timothy Berners-Lee is referred to as the inventor of the World Wide Web (WWW), and proposed the concept of a “distributed hypertext system” at CERN. He defined the concept of the WWW, HTML, and invented the first browser and server side software [ADDA]. Since the 1980s, the WWW has worked under the concept of Hypertext and used the standardized markup language called HTML, which is like SGML (Standard Generalized Markup Language) and is used to share information on the internet. HTML is a language for Web pages and it uses the tags that the web browser understands it and format the page according to the html tags instructions.

The HTML protocol on the WWW supports the HTTP protocol, which is responsible for sending text and multimedia documents along with HTML documents on the internet. HTML is the bunch of rules for transferring files such as text files, images, video, sound and other multimedia documents on the internet. In the early stages, the web browser NeXTStep was developed that have provision to handle html based web pages. Later in 1993, XMosaic was introduced by NCSA that works on a single window where authoring needs to be done using the separate window [ADDA]. At that time, the web page rendering was so simple. One could create a link from where the other web page can be opened. The NeXTStep has brought up the idea of back and forward operation which emerged the concept of content page that contains a path of other pages. There are two models available in browsers one is back and forward and the other is next-previous.

In 1999, more advanced web browsers were designed that were more hypertext-enabled with some other powerful features. XML undoubtedly brought a tremendous addition to WWW hypertext functionality, as it provides a "better hypertext" and allows many different document formats to be defined and transferred back and forth on the web [Cailliau98].

XML provides more advanced linking of "one-to-many" links and provides the feature that one can execute the code by clicking on the link. It also has the feature of pointing of any tag and at any position in the document. By using XML, powerful search engines for the WWW are being made.

Some people think that the web browser features like bookmarks, back button etc. are most useful features that the end user usually uses it. However, the technology never restricts anyone to use only the browser features only. The hypertext functionality such as third-party links and computed links are being developed for improvements in web rendering and fast browsing [Cailliau98].

There are some new hypertext functions of XML introduced, which made significant improvements and will be extensively used. Hypertext is an integral part of the WWW and navigating the websites on the internet is very common now. The hypertext functions are now part of XML standard and are supported globally. However, the functionality of SGML browsers is important in a sense that it deals with the older hypertext features, like document type definitions. The main point is that hypertext is not an application-specific feature now but it is considered a primary feature of electronic documentation [Cailliau98].

3 HTTPS

HTTPS (Hypertext Transfer Protocol Secure) is most simply defined as the Hypertext Transfer Protocol that is operating within another protocol to provide a secure connection over an otherwise unsecured network. Until the recent discovery of a major security flaw known as POODLE (Padding Oracle On Downgrading Legacy Encryption), SSL (Secure Sockets Layer) was known to be the most common method of providing end-to-end encrypted connections through which users were able to utilize services such as e-commerce websites and online bill payments in order to protect their credit card information from falling into the wrong hands, ensuring that data was delivered to the server that was meant to process the transaction [Clark13].

3.1 History of HTTPS

The first version of SSL was version 2.0, (version 1.0 was never released due to major security flaws) which was released in 1995. However, due to version 2.0 major security flaws, it was followed by version 3.0 soon after in 1996. However, in 1999, a new method of creating encrypted connections was developed, known simply as "Transport Layer Security". The main difference between the two methods is that HTTPS functions at the Application Layer while

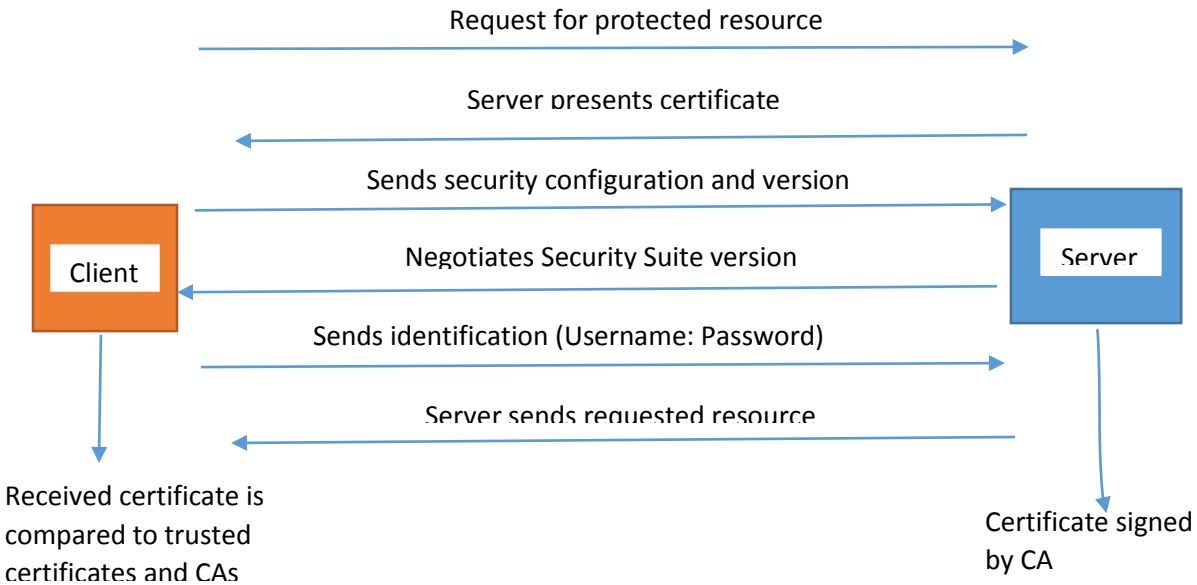
using SSL, where TLS operates at the Transport Layer, completely separating it from the applications and allowing it to function more securely.

3.2 SSL

Secure Sockets Layer, or SSL, was initially developed by Netscape in order for users to have a much more protected and viable option to transmit sensitive data over the web without fear of that data being tapped into, skimmed, or otherwise used for nefarious purposes by some third party. This type of encryption was designed specifically for HTTP for the explicit goals of the secure and private transfer of data over an unsecure network, as well as for server authentication to ensure that all the data that is sent is arriving to the requested server, and not possibly a middle-man or redirected server that is collecting data [Clark13].

It works by the client first sending all its own specifications (version type, compression method etc.) and then the server checking to make sure those specifications are compatible with its own versions, and then it sends an ACK, or an acknowledgment signal, back to the client, or terminates the connection if they are incompatible, usually giving an error message to the client instead. This is known as the "handshake". After the handshake is successful, the server then sends the client a unique certificate that authenticates the server to the client by stating that it is, in fact, who it says it is. However, this certificate must be signed by a Certificate Authority (CA) to truly be considered a "secure" server. If the certificate is not signed, the user will be prompted to continue or not with the connection. This certificate will act as the public key with which the client can send encrypted data to the server with, of which, the server is the only machine that has the key to decrypt that message, and read its contents [Clark13]. By doing this, regardless of what happens with the packets that are sent, the message is encrypted to the point that if someone were to try and access that data, it would be impossible without the server's private key.

Example of HTTPS Certificate Exchange:



SSLv3 functions similarly to SSLv2 in a fundamental sense; however, there are differences. One such difference is the fact that, by design, if one were to try and force a v3 server to function with a non-random key, the servers will reject the request and will not carry out the actions requested. It is a means of preventing a forced rollback. Yet, one major issue for SSLv3 is the allowance of anonymous requests to be made. Making it very difficult to authenticate any requests and allowing attackers to force the rollback and thereby expose the server to attacks that take advantage of SSLv2's inherent weaknesses that v3 does not allow [Clark13].

3.2.1 History of SSL

The first version of SSL, rightfully named Version 1.0, was never released to the main public or anywhere outside of the prototyping phase due to its many security flaws. However, the first version that was universally adopted was version 2.0 [Clark13].

3.3 TLS

Transport Layer Security, or TLS, is a protocol that is designed to allow secure communications between servers and clients on the internet, or on any network. TLS, like SSL before it, helps to ensure that any and all information that is being sent or received is unable to be intercepted, read, or tampered with by third parties [Polk14]. TLS is comprised of several other

protocols and subprotocols that ensure that these security measures function properly and effectively. Even in the event of an error, these subprotocols are designed to correct the error by any means necessary without compromising the data or the session.

4 HTTP vs HTTPS

The major differences between HTTP and HTTPS are simple. Where HTTP was perfectly functional and is still to this day a viable way to transmit static, plain html pages to a requesting client, HTTPS takes that functionality a step further and includes encryption of data to and from the server to allow companies to deal in sensitive information online securely, such as when one purchases something off the internet using a credit card. Where stock HTTP, originally developed in 1989 by Sir Tim Berner-Lee, does not natively support any form of encryption [ADDA]. HTTPS takes the syntax of HTTP, and adds another layer of complexity to it by utilizing SSL, or TLS encryption methods, forming secure and safe connections over the most insecure networks. One other difference that is often overlooked is that HTTP communicates over port 80 while HTTPS communicates over port 443, definitely painting a picture that this is, in fact, a different technology, but one that lives harmoniously with its predecessor [Goldberg98].

Before HTTPS had become widely adopted, many theories and fears circulated that running an encrypted connection would drastically increase load times and tragically harm the speed of the connection. However, as you can see in Example 4.1.1, the problem was studied extensively, and the resulting data shows that there is no sizable difference in speed between HTTP and HTTPS that would warrant not utilizing the more secure and new technology, even in tandem with HTTP. The comparison is being made using two of the same PCs with the same technical specifications to compare the speeds of connections by Kbps (1024 bytes/ms) when transferring data over various forms of connections using various stages of encryption [Goldberg98].

Example 4.1.1)

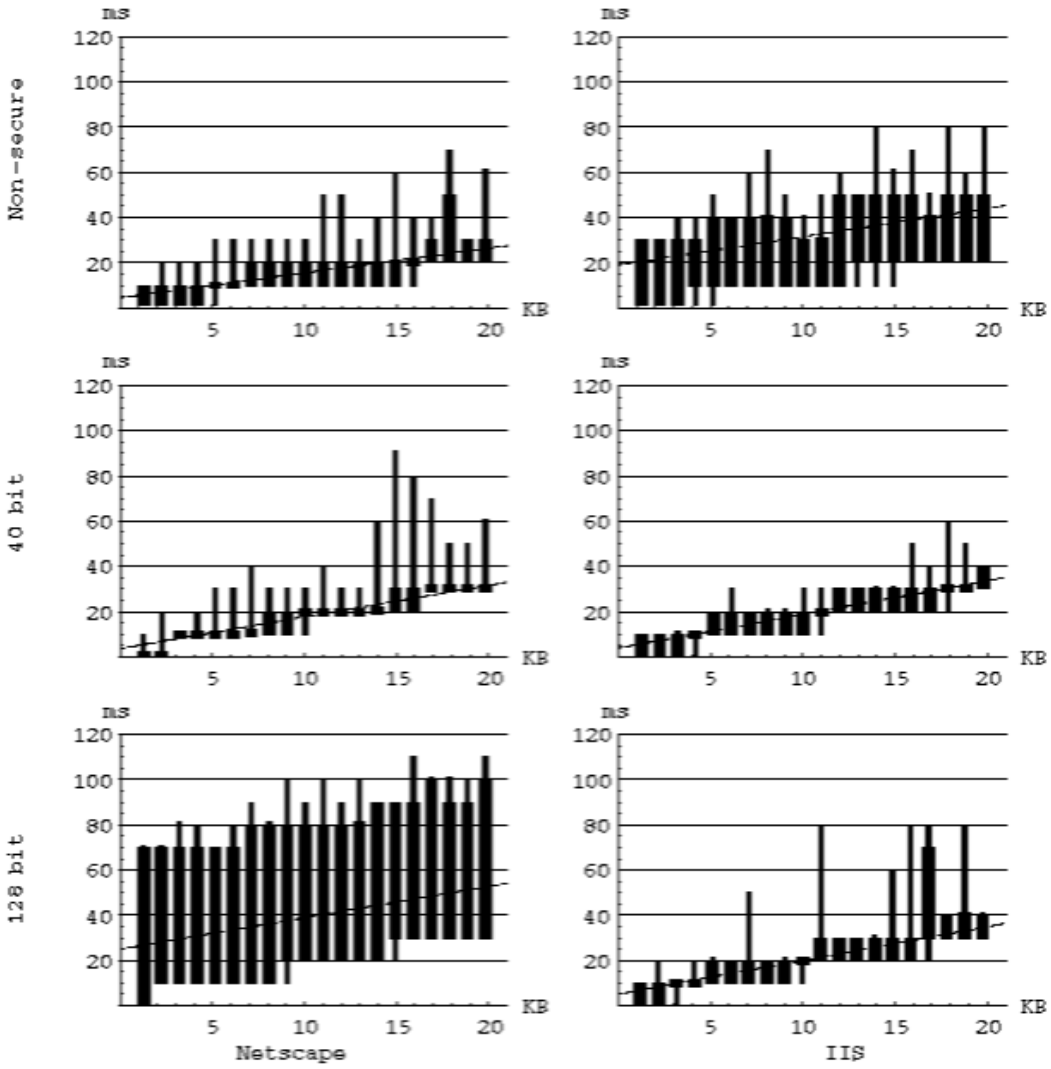


Figure 2. Durations for Non-secure HTTP and HTTPS with Two RC4 Key Sizes for Netscape and Microsoft Web Servers for Documents from 1 to 20 KB. The wide part of a bar marks the range of 75% of the measurements closest to median. The narrow vertical bars mark the range of 95% of the data. A dashed line indicates a least squares fit.

(GOLDBERG, BUFF, & SCHMITT, 1998)

Conclusion

In conclusion, the differences between HTTP and HTTPS are few, but the changes that are held between the two protocols are major. The inclusion of a secured connection options and a manner in which to ensure the identity of a server before one connects to it or sends sensitive data to it brought about the age of electronic commerce and various forms of electronic communications that continue to sculpt and connect the world through social media, electronic mail, community sites, blogs, and various web services that all would be impossible without the advent and implementation of SSL, and later TLS encryption protocols.

Appendix A References

[ADDA] *A history of HTML*, Addison Wesley Longman, 1998, http://www.w3.org/People/Raggett/book4/ch02.html

[Cailliau98] Cailliau, R. and Ashman, H., *Hypertext in the Web – a History*, 1998, http://cs.brown.edu/memex/ACM_HypertextTestbed/papers/62.html

[Clark13] Clark, J., van Oorschot, P., *SoK: SSL and HTTPS: Revisiting past challenges and evaluating certificate trust model enhancements*, 2013, http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6547130

[Goldberg98] Goldberg, A., Buff, R., and Schmitt, A., *A Comparison of HTTP and HTTPS Performance*, 1998, http://www.cs.nyu.edu/artg/research/comparison/comparison.html

[Polk14] Polk, T., McKay, K., and Chokhani, S., *Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations*, 2014, http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52r1.pdf