MESR
منار الشرق للدراسات والأبحاث
Manar Elsharq for Studies & Research
Manar Alsharq Journal
for Education and Instructional Technology
Homepage:
http://meijournals.com/ojs/index.php/majeit/index
ISSN: 2790-6698
مجلة منار الشرق
للتربية و تكنولوجيا
التعليم

Comparison between HTTP and HTTPS based on Software engineering Concepts

Ahmed Saleem Abbas[1], Rusul Khalil Hussein[2], Noor Thamer Mahmood[3]

1. Prof. PhD, Software Department. College of information technology, university of Babylon, Science of College, University of Hilla; Babylon, Hilla, 51001, Email: ahmed_saleam@uobabylon.edu.iq. orcid.org/0000-0001-8556-2739
2. Information Technology, College of Pharmacy, Al- Al-Mustaqbal University, Iraq. Email: rusul.khalil.hussein@uomus.edu.iq
3. Information Technology, Network, Computer Center, University of Babylon, Iraq. Email: nour.thamer95@uobabylon.edu.iq, noorthamer2020@uomus.edu.iq

استلام البحث: 21/10/2025 مراجعة البحث: 22/11/2025 قبول البحث:18/12/2025

Abstract

Hypertext Transfer Protocol Secure (https) has many advantages in terms of performance and security; most importantly, it achieves higher quality standards than Hypertext Transfer Protocol (http). All browsers strongly advise users to trust websites that use HTTPS because it is the only way to protect themselves from a number of risks, attacks, in this paper, the most important characteristics of HTTPS versus HTTP will be discussed, and Tested based on software engineering concepts and selected test tool in comparison with Software Quality Factors.

Keywords: HTTP, HTTPS, SEO (Search Engine Optimization), TLS (Transport Layer Security), comparison HTTP and HTTPS, SSL (Secure Sockets Layer), Software Quality Factors.

1. Introduction

Browsers and servers rely on a common protocol to communicate with each other, as when two people talk to each other it is necessary to use the same language to deliver the message, and this protocol called HTTP [1], which is short for Hypertext Transfer Protocol. This protocol used to exchange data between the client and the server, and it is the most basic protocol on the Internet. However, it works by exchanging information in plain text, which is easy to access and intercept while being transmitted over the Internet, making HTTP sites can be hacked by hackers. This is where HTTPS evolve. So, what is the deal with the added "S"? HTTPS, or Hypertext Transfer Protocol Secure, uses Secure Sockets Layer to encrypt data sent over the World Wide Web, protecting personal info like credit card numbers, passwords, and addresses [2].

As a result, sites that employ the HTTPS protocol are more trustworthy to users, because the protocol encrypts data in both sides, making it safer for both users and web owners. Some people believe that because the web does not deal with sensitive data, no security is required. However, some Internet service providers inject advertisements into web sites that use the HTTP protocol, which is likely to be harmful, so modern web browsers are now restricting unsafe site functions. As a result, once the site secured, these injected adverts will no longer be possible, see figure 1 that illustrate Security advantage of HTTPS. The implementation of HTTPS mainly applied by those websites that deal with money transactions or transfer user's personal data, which could be highly sensitive. Banking websites are famous examples. In non-technical language, HTTPS ensures that users watch websites that they want to watch.

Fig. 1 Normal HTTP and Secure HTTPS [2]

In this paper, we discuss the comparison between HTTP and HTTPS from software engineering point of view [3].

According to Pressman the Web-based software applications defined as “a web pages that retrieved by a browser and that incorporates executable instructions (e.g., HTML, or Java), and data (e.g., hypertext and a variety of visual and audio formats)” [4].

2. Related Work

Arthur Goldberg, *et al*, 1998, this study made a comparison between the non-encrypted and encrypted Web communications in performance site of view [2].

Kefei Cheng, *et al.*, 2010, this paper discusses two possible methods of attacking the HTTPS session, which are certificate and transfer from HTTPS data to HTTP attack, which is dangerous for HTTPS connections, and in return, it proposes three defensive and effective methods to enhance the security and integrity of the data [5].

Pawan Kumar Janbandhu, 2012, This article documents the steps for configuring HTTPS on web servers of all types of web that would serve as an authoritative reference point for those responsible for configuring HTTPS for secure connections [6].

Vamsi Krishna Madasu, *et al.* , 2015, This paper explains the difference between the role of authentication and authorization in networks and the difference between HTTP and HTTPS protocols and their important role in protecting data integrity [1].

Martin Husák, *et al*., 2015, Network traffic analysis and network forensics made more difficult by the increasing amount of encrypted network communication. A concise, real-time definition of HTTPS, Client Network, and SSL/TLS fingerprinting given in this paper [3].

Jannis Müthing, *et al*., 2017, this paper analyzes the security situation in mobile health applications. Mobile health applications are becoming increasingly important as they monitor patients' health, so insufficient transfer of security will lead to a violation of patient privacy and safety issues related to data integrity [7].

Manh Cong Tran*, et al.*, 2018, In this paper, an application is proposed to monitor and analyze HTTP connections based on the basic characteristics of HTTP ،This application helps early detection of risks in environment of HTTP by collecting and definition of traffic, so necessary and appropriate decisions will be taken [8].

3. Software Quality Factors of Web Application

Regardless of budget, application success is crucial for any company. Each year, tens of thousands of apps fail owing to low quality and other obvious causes. Hundreds of apps also rejected from app stores because they do not match the required requirements [9] [10].

Software engineers approve Eleven Software Quality Factors as important factor to make software Succeed but five factors from them must be available in web applications [11], as shown in figure 2.

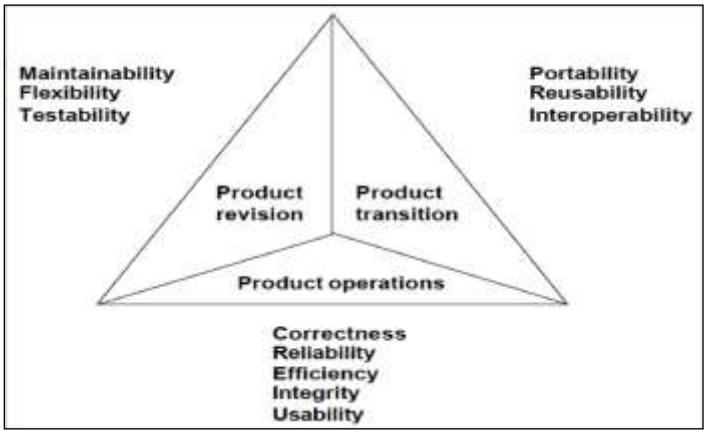

Fig. 2 McCall's Software Quality Factors [12]

Accordingly, we will compare the two protocols (HTTP and HTTPS) according to these five factors listed below, they are:

- Correctness: How well a program adheres to its attributes and accomplishes the customer's requirements.
- Reliability: The probability that a program will fulfill its intended function with the needed precision.

- A program's efficiency defined as the amount of computing resources and code necessary to fulfill its function.
- Integrity: The degree of controlling illegal access to data or software.
- Usability is the amount of effort necessary to understand, operate, prepare input, and comprehend a program's output.

Another six software quality factors is also important but they are essential in any successful software products, so there is no clear difference according them between the two protocols.

4. Protocol Productivity

In this section the work sequence of both protocols will explained and tested according to Software quality Factors.

A. HTTP Protocol Work

When a computer (client) types a website URL that begins with "HTTP://" into a web browser, the browser sends a request to the web server hosting the website. The web server then sends a response to the browser in the form of an HTML page or another document type. The browser displays the user's answer from the server [13] [14].

Requests and answers in HTTP used to communicate between clients and servers, Figure 3 explain the work steps of HTTP, [15]:

1. A browser (client) sends an HTTP request to the web.
2. The request received by a web server.
3. To handle the request, the server launches an app.
4. The server sends the browser an HTTP response (output).
5. The client receives the response.

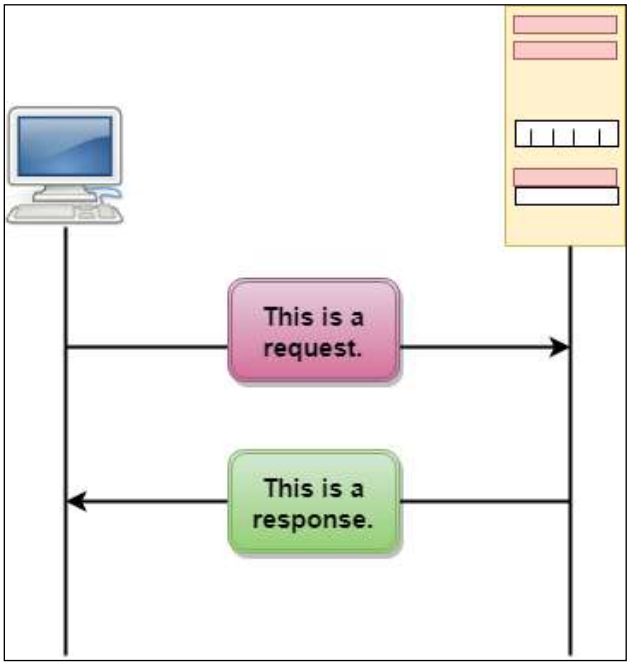

Fig. 3 HTTP sequence diagram [15]

In this protocol, we note that four of the software quality factors exist, they are (Correctness, Reliability, Efficiency, and Usability) but the Integrity factor not met.

B. HTTPS Protocol Work

HTTPS uses an SSL (Secure Sockets Layer) certificate to establish an encrypted connection between the server and the browser. It is a small digital certificate that used to authenticate the identity of the website. It mainly used by letting the user's computer to know that the visited website is trustworthy or not. After several steps of the verification process, a secure session created to exchange data between the user's computer and web server [16].

HTTPS distributes a shared key (symmetric) for authentication and data encryption through SSL/TLS using the encryption of public key. By default, it utilizes port 443, whereas HTTP uses port 80. Port 443, which also enables HTTP connections, which required for all secure transfers. An SSL/TLS handshake occurs between the browser and the server before a data transfer via HTTPS begins to determine the connection parameters. In order to establish a secure connection, the handshake is also necessary [17]. Figure 3 explain the HTTPS work steps, these steps as the following:

- The encryption standards of both parties communicated to each other.
- The browser and the server share the same certificate.
- The customer checks the validity of the certificate.
- The client generates a pre-master secret key using the public key.
- The public key used to encrypt the secret key, and then it shared with the server.
- Based on the value of the secret key, the client and server compute the symmetric key.
- Both parties agree that they have figured out the hidden key.
- Symmetric encryption used for data transfer.

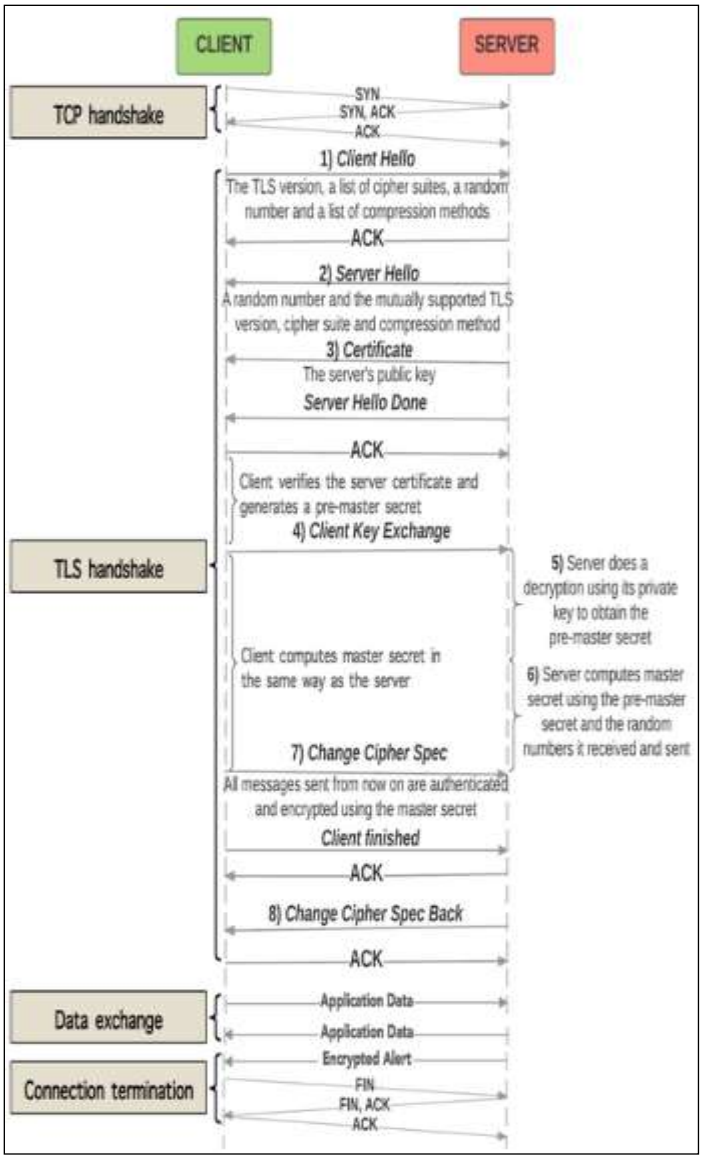

Fig. 4 HTTPS sequence diagram [17]

In this protocol, we note that five of the software quality factors are exist, which are (Correctness, Reliability, Efficiency, Usability, and Integrity)**.** In this protocol, we notice that all Web Software Quality Factors, including Integrity are available and verified, which regard as the most significant Factor because it works to prevent unauthorized people from accessing programs or data.

5. Disadvantages of HTTPS

In this section, the four key issues of using HTTPS will mentioned and discussed.

Price: To switch to https, there is a need to buy an SSL certificate and need to update annually for a cost. In addition, there are a free SSL certificate, but for security reasons it is not recommended. For example, some company offer a full line of SSL services as shown in figure 5.

Fig. 5 SSL Certificate plans as example

Execution: Due to the employment of several mathematical operations to encrypt and decrypt data in https connections, there is a delay in response and the Internet's speed diminishes.

Using Caching: In HTTPS, caching problem occur in some materials. The occurrence of public caching that occur previously will not happen again. ISPs will be unable to cache encrypted content as a result. Sites with many visits are more likely to have issues like these. However, these concerns mitigated because of increased bandwidth, see figure 6 that illustrate caching.

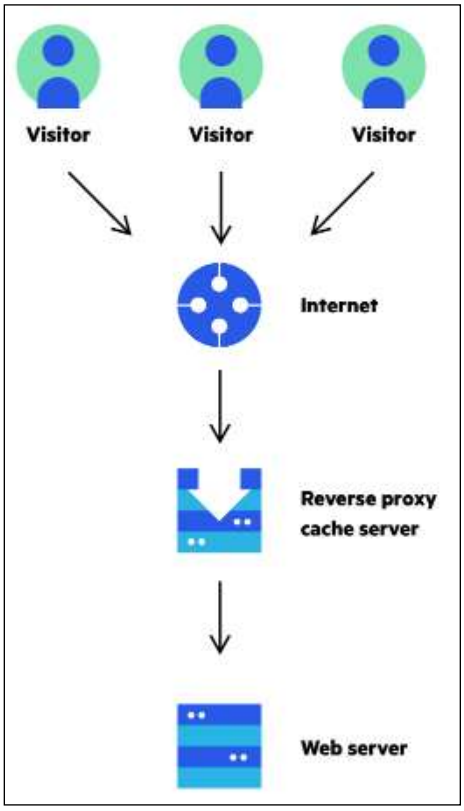

Fig. 6 Server caching

Expenses for overhead: Additional computing costs imposed on both the servers and the browser because of data encryption and decryption effort. Due to the additional latency emitted by the connection setup, this penalty is usually not visible. This could be a problem if a site's HTTPS connection manages numerous HTTPS connections at the same time.

6. The importance of HTTPS for e-commerce sites

Recently, individuals have gradually gone to online shopping, which has expanded the competition among e-commerce website owners Customer trust in the product is the key ingredient to the success of any online store. One of the means of purchasing interaction is to give the customer his Master card data, this data considered sensitive and must guaranteed, if not the security component would be absent. By encrypting data, HTTPS delivers a safe surfing experience, making it far more difficult for cyber hackers to steal information. Data transmitted between a number of computers during online transactions, and if the site is not HTTPS, anyone can see it. This implies that HTTPS protects important data like credit card numbers, home addresses, login IDs, and other personal information. It also safeguards online retailers and their consumers against identity theft, ensuring that the correct individual makes each transaction.

7. SEO Advantages of Switching to HTTPS

The following are the three key SEO benefits of adopting HTTPS:

- **Improved Ranking**: While it may not be a significant improvement, SEO experts expect that the HTTPS ranking signal will become more powerful with time.

- **Referral Data:** When looking at the Google Analytics data for an HTTP site, traffic that comes from referral sources can show as "direct" traffic. The security of the referring domain preserved on an HTTPS website.

- **Privacy and Security:** The advantages of greater security benefits are as follows:
  - Website authentication and server communication.
  - Keeping third-party damage to a minimum.
  - Data and communication encryption, such as browsing history and credit card information.

8. Results

In this section, the results of the performance comparison between HTTP and HTTPS will be computed and discussed, the test according to software quality factors (SQF) will be done. Because of HTTP simplicity, HTTP is generally faster than HTTPS. Unlike HTTP, HTTPS requires an additional SSL handshake phase. This extra step slows down the website's page load time somewhat. However, this is dependent on a number of factors, including the duration of the session, Static to dynamic content ratio, client-caching behavior, server software, hardware, and so on.

As example, if the server has a lot of dynamic material, HTTPS is less likely to slow down the page load; the time spent on the handshake (SSL) is minor in compared to the time spent on content development, and this a reason of that slow. The overhead is higher when there is a lot of static information. SSL handshake duration has an impact on very brief sessions as well. However, in the case of long sessions, this fee spent at the start of the session, and subsequent requests will faster process. But the increased security given by HTTPS much outweighs the minor performance hit. There are a few other options for improving performance of HTTPS. This contains the following:

HTTP/2: HTTPS only become faster with HTTP/2, which offsets any performance overheads. Multiplexing and concurrency, stream dependencies, header compression, and server push are the key benefits and features of HTTP/2. Brotli compression is a Google-developed open-source lossless compression method. It helps to speed up content loading by reducing bandwidth use. HPACK compression reduces the header size by roughly 30% and is based on Huffman encoding. HPACK is immune to compression-based attacks and can encode big headers. OCSP (Office of Civil Service

Personnel) (Online certificate status protocol) Stapling is a technique for quickly verifying an SSL certificate.

http://www.httpvshttps.com/ is a website used to compares HTTPS with HTTP. The test result showed that the HTTPS version took 113.688 seconds to load and HTTP version took 41.080 seconds to load for the identical website, So HTTP 64% faster than HTTPS. This test performed in the Chrome browser, as shown in figure 7. Other testing platforms and visual comparison tools that compare the load times of HTTPS and HTTP page versions are also available.

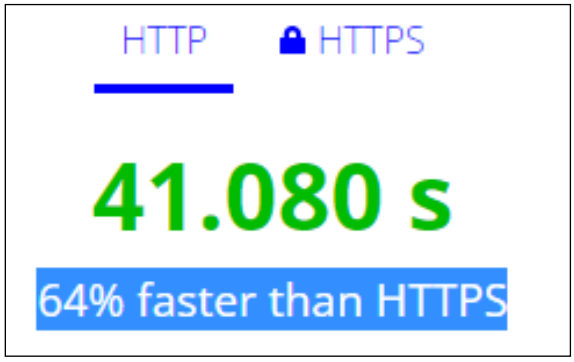

Fig. 7: Web page loading test Result

See table 1, that illustrate the comparison between the two protocols (HTTP and HTTPS) according to the selected five software quality factors.

Table 1: comparison result according to SQF

Software Quality Factor	Levels	HTTP	HTTPS
Correctness	High		yes
	Medium	yes	
	Low		
Reliability	High		yes
	Medium		
	Low	yes	
Program's efficiency	High		yes
	Medium	yes	
	Low		
Integrity	High		yes
	Medium		
	Low	yes	
Usability	High	yes	
	Medium		yes
	Low		

9. Conclusion

In this paper, the most important characteristics of HTTPS versus HTTP discussed, HTTPS has many advantages in terms of performance and security, and most importantly, it achieves higher

quality standards than HTTP. We strongly advise users to trust websites that use HTTPS because it is the only way to protect themselves from a number of risks and attacks.

References

[1] V. K. Madasu, "Web Authentication and Authorization and Role of HTTP , HTTPS Protocol in Networking," *Jmest*, 2015. vol. 2, no. 3, pp. 381–383,.

[2] M. Husák, M. Čermák, T. Jirsík, and P. Čeleda, "Network-based HTTPS client identification using SSL/TLS fingerprinting," *Proc. - 10th Int. Conf. Availability, Reliab. Secur. ARES 2015*, pp. 389–396, 2015, doi: 10.1109/ARES.2015. 35.

[3] A. Goldberg, R. Buff, and A. Schmitt, "Comparison of HTTP and HTTPS performance," *C. Proc.*, vol. 1, pp. 226–230, 1998.

[4] Pressman, " Software engineering practitioners approach ", Book, 8th edition.

[5] K. Cheng, M. Gao, and R. Guo, "Analysis and research on HTTPS hijacking attacks," *NSWCTC 2010 - 2nd Int. Conf. Networks Secur. Wirel. Commun. Trust. Comput.*, vol. 2, pp. 223–226, 2010, doi: 10.1109/NSWCTC.2010.187.

[6] P. K. Janbandhu, "Implementing HTTPS for securing webservers," no. June 2012, 2019, doi: 10.13140/RG.2.2.26172.39042/1.

[7] J. Müthing, T. Jäschke, and C. M. Friedrich, "Client-focused security assessment of mHealth apps and recommended practices to prevent or mitigate transport security issues," *JMIR mHealth uHealth*, 2017, vol. 5, no. 10, , doi: 10.2196/mhealth.7791.

[8] M. C. Tran, M. H. Nguyen, and T. Q. Nguyen, "An application for monitoring and analysis of HTTP communications," *J. Commun.*, 2018, vol. 13, no. 8, pp. 456–462, doi: 10.12720/jcm.13.8.456-462.

[9] M. Jørgensen, "Software quality measurement," *Adv. Eng. Softw.*, vol. 30, no. 12, pp. 907–912, 1999, doi: 10.1016/S0965-9978(99)00015-0.

[10] D. Nabil, A. Mosad, and H. A. Hefny, "Web-Based Applications quality factors: A survey and a proposed conceptual model," *Egypt. Informatics J.*, 2011, vol. 12, no. 3, pp. 211–217, , doi: 10.1016/j.eij.2011.09.003.

[11] B. W. Boehm, "Software engineering-as it is," *Softw. Eng. Barry W. Boehm'S Lifetime Contrib. to Softw. Dev. Manag. Res.*, pp. 663–685, 2007, doi: 10.1109/9780470187562.ch8.

[12] D. Galin, "Software Quality Factors (Attributes)," *Softw. Qual. Concepts Pract.*, pp. 23–44, 2018, doi: 10.1002/9781119134527.ch2.

[13] M. Kassim, M. Ismail, K. Jumari, and M. I. Yusof, "Bandwidth gain analysis for HTTP and HTTPs traffic on IP based network," *IEEE Symp. Wirel. Technol. Appl. ISWTA*, pp. 303–308, 2012, doi: 10.1109/ISWTA.2012.6373866.

[14] S. Dyllan, H. Dahimene, P. Wright, and P. Xiao, "Analysis of HTTP and HTTPS Usage on the University Internet Backbone Links," *J. Ind. Intell. Inf.* 2014, vol. 2, no. 1, pp. 67–70, , doi: 10.12720/jiii.2.1.67-70.

[15] M. Husák, M. Čermák, T. Jirsík, and P. Čeleda, "HTTPS traffic analysis and client identification using passive SSL/TLS fingerprinting," *Eurasip J. Inf. Secur.*, vol. 2016, no. 1, pp. 1–14, doi: 10.1186/s13635-016-0030-7.

[16] D. Naylor *et al.*, “The cost of the ‘s’ in HTTPS,” *Conex. 2014 - Proc. 2014 Conf. Emerg. Netw. Exp. Technol.*, 2014,pp. 133–139, doi: 10.1145/2674005.2674991.

[17] H. Kolamunna *et al.*, “Are wearable devices ready for HTTPS? Measuring the cost of secure communication protocols on wearable devices,” 2016, [Online]. Available: http://arxiv.org/abs/1608.04180.