PolotskSU

ICT, ELECTRONICS, PROGRAMMING

UDC 004.05

HYPERTEXT TRANSFER PROTOCOL SECURE. SECURITY WEB SERVICES

DZMITRY TATARYN, IRYNA BURACHONAK
Polotsk State University, Belarus

The features of the HyperText Transfer Protocol Secure Protocol (HTTPS) and the problems that can be encountered when using it are considered. Conclusions are made on the necessary measures to ensure complete security when using the HTTPS protocol.

When you enter the site name, everyone is accustomed to seeing the http: // or HyperText Transfer Protocol (HTTP) links at the beginning – the Hypertext Transfer Protocol – the standard data transfer protocol from the server hosting the site to the user. However, despite its popularity, more and more websites prefer to use a more advanced protocol, HTTPS, since it protects transmitted data from being intercepted by attackers by encrypting it. In the present article we will consider this protocol in more detail: how it works, who is recommended to use it and what is needed to connect it to the site.

HTTPS is a protocol that ensures the confidentiality of data exchange between a site and a user device. Information security is ensured through the use of cryptographic protocols SSL (Secure Sockets Layer) – a layer of secure sockets and its predecessor TLS (Transport Layer Security) – a transport layer security protocol, further SSL / TLS, having three levels of security:

– data encryption - allows you to avoid their interception;

– data integrity - any change in data is recorded;

– authentication - protects against user redirection.

SSL can be compared with the "candy wrapper" in which HTTP data is wrapped to hide it from outsiders. The SSL / TLS protocol helps two unfamiliar with each other Internet users to establish a secure connection through a normal, non-secure channel. Using mathematical algorithms, both users – the client and the server – agree on the secret key without transferring it directly through the connection. Even if someone manages to connect to the connection and intercept all transmitted data, it will not be able to decrypt it.

SSL uses a multilayered environment: on the one hand, it is the client program protocol, for example, (Internet Message Access Protocol (IMAP) – an application layer protocol for access to e-mail, File Transfer Protocol (FTP) – file transfer protocol and HTTP), and on the other, TCP / IP transport. The name TCP / IP comes from the two most important protocols of the family – Transmission Control Protocol (TCP) / Internet Protocol (IP) or Transmission Control Protocol / Internet Protocol. For SSL encryption, symmetric and ac-symmetric keys obtained using various mathematical models are used [1].

Next, we will conduct a comparative analysis of the use of the HTTPS and HTTP protocols.

For HTTPS connections, TCP port 443 is commonly used. HTTPS is widely used to protect information from interception, and also, as a rule, provides protection against man-in-the-middle attacks – if the certificate is verified on the client, and at the same time, the private key of the certificate was not compromised, the user did not confirm the use of the unsigned certificate, and the certificates of the malefactor's certificate were not implemented on the user's computer. The man-in-the-middle attack can be seen in Figure 1.

Currently, HTTPS is supported by all popular web browsers.

Mandatory use of a secure data transfer protocol requires all information relating to making payments on the Internet: payment for goods in online stores in any way (individual payment card, online payment systems, etc.), payment for services through Internet banking, making payments online services (casino, online courses, etc.) and many more. The use of the HTTPS protocol is also recommended on sites that request user data for access to certain content, for example, a passport number – such data must be protected from being intercepted by hackers.

If your site uses something similar, then you should seriously consider switching to HTTPS. Therefore, we will consider below what is necessary for this.

The operation of the HTTPS protocol is based on the fact that the user's computer and the server select a shared secret key with which the transmitted information is encrypted. This key is unique and is generated for each session. It is believed that it is impossible to fake it, because it contains more than 100 characters. In order to avoid data interception by a third party, a digital certificate is used – this is an electronic document (ED) that

PolotskSU

identifies the server. Each owner of the site (server) must have such a certificate to establish a secure connection with the user. This ED specifies the owner and signature. With the help of a certificate, you confirm that:

– the person to whom it is issued, really exists,

– it is the owner of the server (site) that is specified in the certificate.

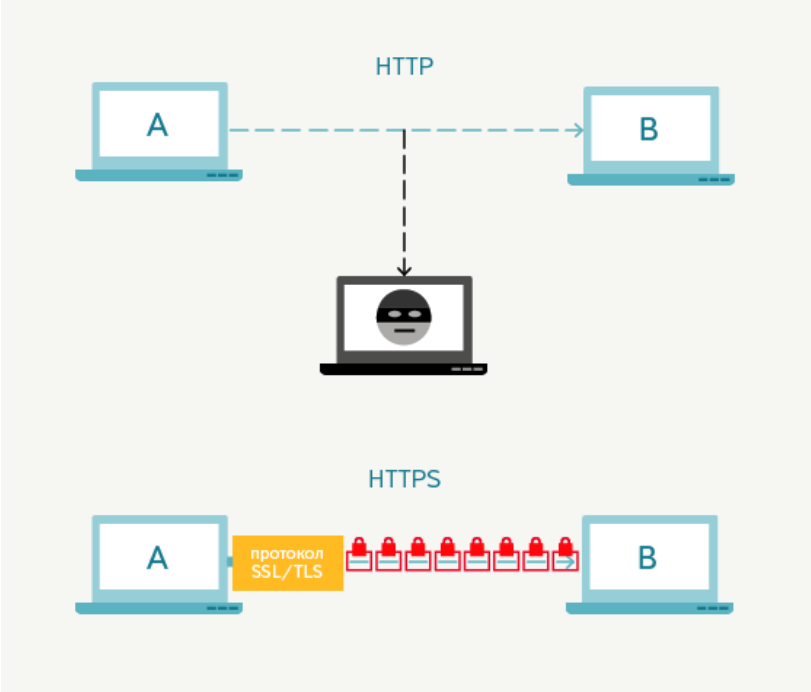

Figure 1. – Comparison of HTTP and HTTPS

The first thing that the browser does when establishing a connection via the HTTPS protocol is to check the authenticity of the certificate, and only in the case of a successful response, data exchange begins.

More details on the problems that can be encountered using HTTP:

1 Sharing HTTP and HTTPS. When sites use mixed HTTP and HTTPS functionality, this potentially leads to an informational threat to the user. For example, if the main pages of a certain site are loaded using HTTPS, and Cascading Style Sheets (CSS) and JavaScript are loaded via HTTP, then the attacker, at the time of downloading the latter, can load his code and get the HTML page data. Many sites, despite such vulnerabilities, download content through third-party services that do not support HTTPS. The HSTS mechanism allows you to prevent such vulnerabilities by forcing the use of HTTPS connections even where the default is HTTP [2].

2 Attacks using traffic analysis. In HTTPS, traffic analysis vulnerabilities were discovered. An attack with traffic analysis is a type of attack in which properties of protected channel data are displayed by measuring the size of traffic and the transmission time of messages in it. Traffic analysis is possible because SSL / TLS encryption alters the content of the traffic, but has a minimal impact on the size and time it takes to pass traffic. In May 2010, researchers from Microsoft Research and Indiana University found that detailed confidential user data can be obtained from non-essential data, such as package sizes. The traffic analyzer was able to get a history of diseases, data on used medications and user operations, data on family income, etc. All this was done despite the use of HTTPS in several modern web applications in the field of healthcare, taxation, etc. [3].

3 The man in the middle of https. When a man-in-the-middle attack is used, the HTTPS server sends a public key certificate to the browser. If this certificate is not trustworthy, then the transmission channel will be vulnerable to the attack of the malicious user. Such an attack replaces the original certificate certifying the HTTPS server with a modified certificate. The attack succeeds if the user neglects double checking the certificate when the browser sends a warning. This is especially common among users who often encounter self-certified certificates when accessing sites within a network of private organizations.

As a result of the study, it can be concluded that the HTTPS protocol is quite flexible and easy to use, it ensures the confidentiality of data exchange between the site and the user device. Thanks to the HTTPS proto-

PolotskSU

col, it has become possible to keep confidential information secret. To communicate with the client, HTTPS uses SSL / TLS for encryption with a fairly high level of security.

REFERENCES

1. Как работает SSL? Принцип работы https соединения. [Электронный ресурс]. – Режим доступа: https://www.ipipe.ru/info/kak-rabotaet-ssl-sertificat.html. – Дата доступа: 12.09.2018.
2. How to Deploy HTTPS Correctly. [Электронный ресурс]. – Режим доступа: https://www.eff.org/https-everywhere/deploying-https. – Дата доступа: 13.09.2018.
3. Side-Channel Leaks in Web Applications: a Reality Today, a Challenge Tomorrow. [Электронный ресурс].– Режим доступа: https://www.microsoft.com/en-us/research/publication/side-channel-leaks-in-web-applications-a-reality-to-ay-a-challenge-tomorrow – Дата доступа: 12.09.2018.