How Does SSL Work? | SSL Certificates and TLS

SSL, also known as TLS, uses encryption to keep user data secure, authenticate the identity of websites, and stop attackers from tampering with Internet communications.

Share

How SSL Works Learning Objectives

After reading this article you will be able to:

- Understand what SSL/TLS means
- Explain how SSL/TLS keeps Internet communications secure
- Learn how to obtain an SSL certificate, and how SSL certificates keep user data safe

Related Content

What is an SSL Certificate?

What is SSL?

SSL Handshake

Keyless SSL

Why Use HTTPS?

What is SSL?

SSL stands for Secure Sockets Layer, and it refers to a protocol for encrypting and securing communications that take place on the Internet. Although SSL was replaced by an updated protocol called TLS (Transport Layer Security) some time ago, "SSL" is still a commonly used term for this technology.

The main use case for SSL/TLS is securing communications between a client and a server, but it can also secure email, VoIP, and other communications over unsecured networks.

How does SSL/TLS work?

These are the essential principles to grasp for understanding how SSL/TLS works:

- Secure communication begins with a TLS handshake, in which the two communicating parties open a secure connection and exchange the public key
- During the TLS handshake, the two parties generate session keys, and the session keys encrypt and decrypt all communications after the TLS handshake
- Different session keys are used to encrypt communications in each new session
- TLS ensures that the party on the server side, or the website the user is interacting with, is actually who they claim to be
- TLS also ensures that data has not been altered, since a message authentication code (MAC) is included with transmissions

With TLS, both HTTP data that users send to a website (by clicking, filling out forms, etc.) and the HTTP data that websites send to users is encrypted. Encrypted data has to be decrypted by the recipient using a key.

The TLS handshake

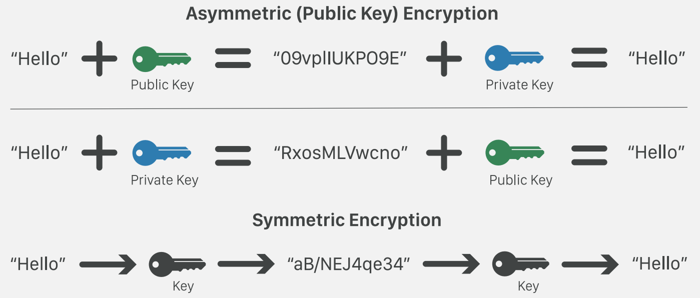
TLS communication sessions begin with a TLS handshake. A TLS handshake uses something called asymmetric encryption, meaning that two different keys are used on the two ends of the conversation. This is possible because of a technique called public key cryptography.

In public key cryptography, two keys are used: a public key, which the server makes available publicly, and a private key, which is kept secret and only used on the server side. Data encrypted with the public key can only be decrypted with the private key, and vice versa.

During the TLS handshake, the client and server use the public and private keys to exchange randomly generated data, and this random data is used to create new keys for encryption, called the session keys.

Symmetric encryption with session keys

Unlike asymmetric encryption, in symmetric encryption the two parties in a conversation use the same key. After the TLS handshake, both sides use the same session keys for encryption. Once session keys are in use, the public and private keys are not used anymore. Session keys are temporary keys that are not used again once the session is terminated. A new, random set of session keys will be created for the next session.



Authenticating the origin server

TLS communications from the server include a message authentication code, or MAC, which is a digital signature confirming that the communication originated from the actual website. This authenticates the server, preventing on-path attacks and domain spoofing. It also ensures that the data has not been altered in transit.

What is an SSL certificate?

An SSL certificate is a file installed on a website's origin server. It's simply a data file containing the public key and the identity of the website owner, along with other information. Without an SSL certificate, a website's traffic can't be encrypted with TLS.

Technically, any website owner can create their own SSL certificate, and such certificates are called self-signed certificates. However, browsers do not consider self-signed certificates to be as trustworthy as SSL certificates issued by a certificate authority.

How does a website get an SSL certificate?

Website owners need to obtain an SSL certificate from a certificate authority, and then install it on their web server (often a web host can handle this process). A certificate authority is an outside party who can confirm that the website owner is who they say they are. They keep a copy of the certificates they issue.

Is it possible to get a free SSL certificate?

Many certificate authorities charge for SSL certificates. To help make the Internet more secure, Cloudflare offers free SSL certificates. Cloudflare was the first Internet security and performance company to do so. Cloudflare also has worked to optimize SSL/TLS performance so that websites moving from HTTP to HTTPS don't have their performance impacted. Learn more about Cloudflare and SSL.

What is the difference between HTTP and HTTPS?

The S in "HTTPS" stands for "secure." HTTPS is just HTTP with SSL/TLS. A website with an HTTPS address has a legitimate SSL certificate issued by a certificate authority, and traffic to and from that website is authenticated and encrypted with the SSL/TLS protocol.

To encourage the Internet as a whole to move to the more secure HTTPS, many web browsers have started to mark HTTP websites as "not secure" or "unsafe." Thus, not only is HTTPS essential for keeping users safe and user data secure, it has also become essential for building trust with users. Test a website for SSL/HTTPS issues.

About SSL

What is SSL?

What is TLS?

How SSL Works

Contact Sales:

+1 (888) 99 FLARE

About HTTPS

What is HTTPS?

Why Use HTTPS?

HTTP Security Gaps

Connection Not Private

About Encryption

What is Encryption?

Public Key Encryption

Asymmetric Encryption

Lava Lamp Encryption

Cryptographic Key

What is a Session Key?

SSL Glossary

What is Mixed Content?

SSL Handshake

What is an SSL Certificate?

SSL Certificate Types

Why Use TLS 1.3?

What is SNI?

What is Encrypted SNI?

What is Domain Spoofing?

Learning Center Navigation

Learning Center Home

DDoS Learning Center

CDN Learning Center

DNS Learning Center

Performance Learning Center

Security Learning Center

Serverless Learning Center

Bots Learning Center

Cloud Learning Center

Access Management Learning Center

Network Layer Learning Center