

## The differences between HTTP and HTTPS

---

HTTP (HyperText Transfer Protocol) is the most widely used protocol for data transfer over the web. But the data (i.e. hypertext) exchanged using HTTP isn't as secure as we would like it to be.

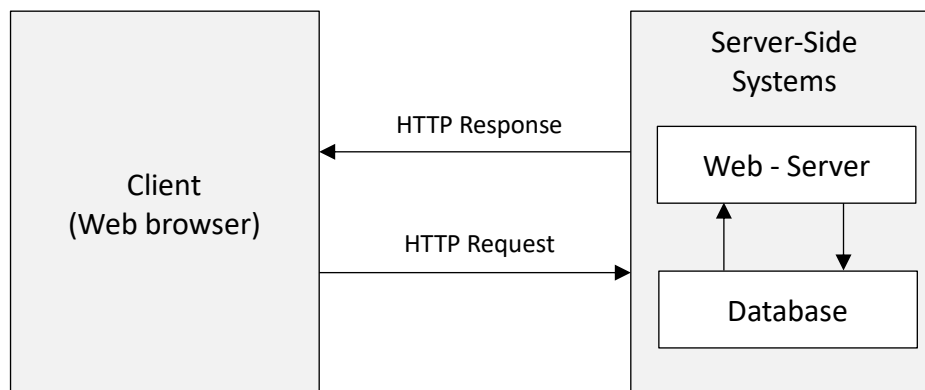
HTTPS (Hypertext Transfer Protocol Secure) is an extended version of the HTTP and it is used for secure communications, in fact is like HTTP but with verification and encryption. In HTTPS, the communication protocol is encrypted using TLS (Transport Layer Security).

Transport Layer Security (TLS) is a cryptographic protocol designed to provide communications security over a computer network, such as the Internet. The TLS protocol aims primarily to provide security, including privacy (confidentiality), integrity, and authenticity through the use of cryptography, such as the use of certificates, between two or more communicating computer applications. TLS was derived from a security protocol called Secure Socket Layer (SSL).

## HTTP request and response

---

### Representation of a Request-Response cycle



### HTTP Request Structure

A simple request message from a client consists of the following components:

- A request line to get required resource
- Headers
- An empty line
- A message body (optional)

Example:

```
POST / HTTP/1.1
Host: developer.mozilla.org
User-Agent: curl/8.6.0
Accept: */*
Content-Type: application/json
Content-Length: 345

{
  "data": "ABC123"
}
```

## HTTP Response Structure

A simple response from the server contains the following components:

- HTTP Status Code
- Headers
- An empty line
- A message body (optional)

Example:

```
HTTP/1.1 403 Forbidden
Server: Apache
Date: Fri, 21 Jun 2024 12:52:39 GMT
Content-Length: 678
Content-Type: text/html
Cache-Control: no-store

<!DOCTYPE html>
<html lang="en">
(more data)
```

## Lists of common HTTP methods and status codes

---

### HTTP Methods

Method	Description	Use case
<b>GET</b>	The GET method requests a representation of the specified resource. Requests using GET should only retrieve data and should not contain a request content.	Fetching a web page or data from an API.
<b>POST</b>	The POST method submits an entity to the specified resource, often causing a change in state or side effects on the server.	Provide a block of data, such as the result of submitting a form, to a data-handling process.
<b>PUT</b>	The PUT method replaces all current representations of the target resource with the request content. The difference between PUT and POST is that PUT is idempotent: calling it once is no different from calling it several times successively (there are no side effects).	Same as POST method.
<b>DELETE</b>	The DELETE method deletes the specified resource.	Deleting a specified resource like an html file or data from an API.
<b>PATCH</b>	The PATCH method applies partial modifications to a resource.	Modifying only just some keys of a data resource like "user" ({ "age": 45} to { "age": 46})

## Status Codes

Status Code	Description	Scenario
<b>200 - OK</b>	The request succeeded. The result and meaning of "success" depends on the HTTP method.	For GET, the resource has been fetched and transmitted in the message body.
<b>403 - Forbidden</b>	The client does not have access rights to the content; that is, it is unauthorized, so the server is refusing to give the requested resource.	If an unauthorized user tries to access to a page or resource that requires authorization.
<b>404 – Not Found</b>	<p>The server cannot find the requested resource.</p> <p>In the browser: the URL is not recognized.</p> <p>In an API: the endpoint is valid but the resource itself does not exist.</p> <p>Servers may also send this response instead of 403 Forbidden to hide the existence of a resource from an unauthorized client.</p>	If a user requests a page or resource that doesn't exist
<b>500 – Internal Server Error</b>	The server has encountered a situation it does not know how to handle. This error is generic, indicating that the server cannot find a more appropriate 5XX status code to respond with.	There are many possible causes of 500 errors, including: improper server configuration, out-of-memory (OOM) issues, unhandled exceptions, improper file permissions, or other complex factors.
<b>502 – Bad Gateway</b>	This error response means that the server, while working as a gateway to get a response needed to handle the request, got an invalid response.	There are many causes of 502 errors, and fixing such problems probably requires investigation by server owners or administrators.