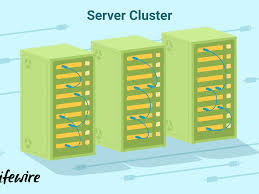
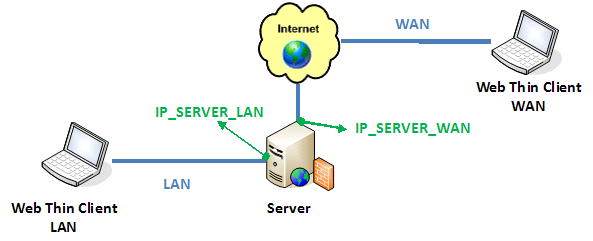
1. **Server**

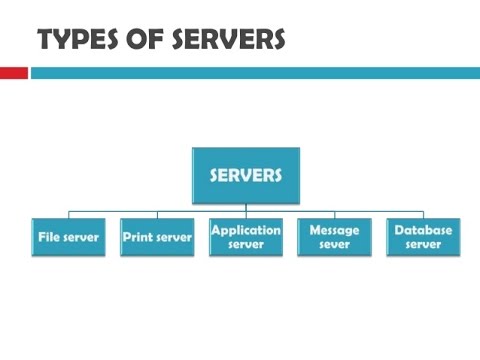
A server is a computer that provides data to other computers. It may serve data to systems on a local area network (LAN) or a wide area network (WAN) over the Internet. Many types of servers exist, including web servers, mail servers, and file servers. Each type runs software specific to the purpose of the server.

A server is a computer that provides data to other computers. For example, a Web server may run Apache HTTP Server or Microsoft IIS, which both provide access to websites over the Internet. A mail server may run a program like Exim or iMail, which provides SMTP services for sending and receiving email.





1. **Types of server**

****

server types used today:

**1. Application Servers**Sometimes referred to as a type of [middleware](http://searchsoa.techtarget.com/definition/middleware), application servers occupy a large chunk of computing territory between database servers and the end user, and they often connect the two.

**2. Client Servers**In the client/server programming model, a server is a program that awaits and fulfills requests from client programs in the same or other computers. A given application in a computer may function as a client with requests for services from other programs and also as a server of requests from other programs.

**3. Collaboration Servers**In many ways, collaboration software, once called ‘groupware,’ demonstrates the original power of the Web. Collaboration software designed to enable users to collaborate, regardless of location, via the Internet or a corporate intranet and to work together in a virtual atmosphere.

**4. FTP Servers**  
One of the oldest of the Internet services, File Transfer Protocol, makes it possible to move one or more files securely between computers while providing file security and organization as well as transfer control.

**5. List Servers**List servers offer a way to better manage mailing lists, whether they be interactive discussions open to the public or one-way lists that deliver announcements, newsletters or advertising.

**6. Mail Servers**Almost as ubiquitous and crucial as Web servers, mail servers move and store mail over corporate networks ([via LANs and WANs](http://www.diffen.com/difference/LAN_vs_WAN)) and across the Internet.

**7. Open Source Servers**From your underlying open source server operating system to the server software that help you get your job done, open source software is a critical part of many IT infrastructures.

**8. Proxy Servers**Proxy servers sit between a client program (typically a Web browser) and an external server (typically another server on the Web) to filter requests, improve performance, and share connections.

**9. Real-Time Communication Servers**  
Real-time communication servers, formerly known as chat servers or IRC Servers, and still sometimes referred to as instant messaging (IM) servers, enable large numbers users to exchange information near instantaneously.

**10. Server Platforms**  
A term often used synonymously with operating system, a platform is the underlying hardware or software for a system and is thus the engine that drives the server.

**11. Telnet Servers**A Telnet server enables users to log on to a host computer and perform tasks as if they’re working on the remote computer itself.

**12. Virtual Servers**  
In 2009, the number of virtual servers deployed exceeded the number of physical servers. Today, server virtualization has become near ubiquitous in the data center. From hypervisors to hybrid clouds, ServerWatch looks at the latest virtualization technology trends.

**13. Web Servers**  
At its core, a Web server serves static content to a Web browser by loading a file from a disk and serving it across the network to a user’s Web browser. This entire exchange is mediated by the browser and server talking to each other using HTTP.

**HTTP**

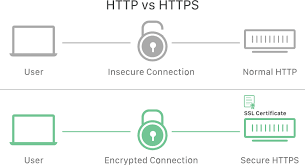
It work on presentation layer in OSI model.

Hypertext Transfer Protocol (**HTTP**) is an application-layer protocol for transmitting hypermedia documents, such as HTML. It was designed for communication between web browsers and web servers, but it can also be used for other purposes.

**HTTPS**

Hypertext transfer protocol secure (HTTPS) is the secure version of [HTTP](https://www.cloudflare.com/learning/ddos/glossary/hypertext-transfer-protocol-http/), which is the primary protocol used to send data between a web browser and a website. HTTPS is encrypted in order to increase security of data transfer. This is particularly important when users transmit sensitive data, such as by logging into a bank account, email service, or health insurance provider.

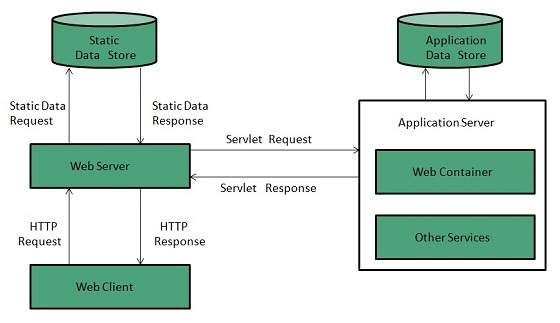
Any website, especially those that require login credentials, should use HTTPS. In modern web browsers such as Chrome, websites that do not use HTTPS are marked differently than those that are. Look for a green padlock in the URL bar to signify the webpage is secure. Web browsers take HTTPS seriously;

****

**HTTP Working**

**Featured snippet from the web**

How does **HTTP work**? As a request-response protocol, **HTTP** gives users a way to interact with web resources such as HTML files by transmitting hypertext messages between clients and servers. **HTTP** clients generally use Transmission Control Protocol (TCP) connections to communicate with servers.

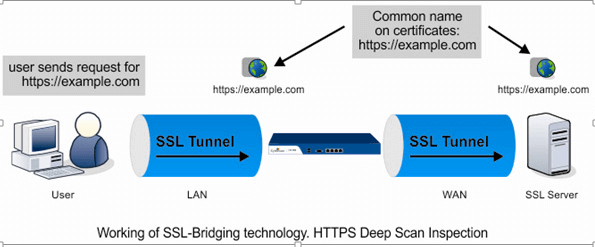


**HTTPS Working**

HTTPS takes the well-known and understood HTTP protocol, and simply layers a SSL/TLS (hereafter referred to simply as “SSL”) encryption layer on top of it. Servers and clients still speak exactly the same HTTP to each other, but over a secure SSL connection that encrypts and decrypts their requests and responses. The SSL layer has 2 main purposes:

* Verifying that you are talking directly to the server that you think you are talking to
* Ensuring that only the server can read what you send it and only you can read what it sends back

The really, really clever part is that anyone can intercept every single one of the messages you exchange with a server, including the ones where you are agreeing on the key and encryption strategy to use, and still not be able to read any of the actual data you send.



# HTTP – Methods

|  |  |
| --- | --- |
| 1 | **GET**  The GET method is used to retrieve information from the given server using a given URI.  Requests using GET should only retrieve data and should have no other effect on the data. |
| 2 | **HEAD**  Same as GET, but transfers the status line and header section only. |
| 3 | **POST**  A POST request is used to send data to the server,  for example, customer information, file upload, etc. using HTML forms. |
| 4 | **PUT**  Replaces all current representations of the target resource with the uploaded content. |
| 5 | **DELETE**  Removes all current representations of the target resource given by a URI. |
| 6 | **CONNECT**  Establishes a tunnel to the server identified by a given URI. |
| 7 | **OPTIONS**  Describes the communication options for the target resource. |
| 8 | **TRACE**  Performs a message loop-back test along the path to the target resource. |

# HTTP - Status Codes

|  |  |
| --- | --- |
| **S.N.** | **Code and Description** |
| 1 | **1xx: Informational**  It means the request has been received and the process is continuing. |
| 2 | **2xx: Success**  It means the action was successfully received, understood, and accepted. |
| 3 | **3xx: Redirection**  It means further action must be taken in order to complete the request. |
| 4 | **4xx: Client Error**  It means the request contains incorrect syntax or cannot be fulfilled. |
| 5 | **5xx: Server Error**  It means the server failed to fulfill an apparently valid request. |

HTTP status codes are extensible and HTTP applications are not required to understand the meaning of all the registered status codes. Given below is a list of all the status codes.

1xx: Information

|  |  |
| --- | --- |
| **Message** | **Description** |
| 100 Continue | Only a part of the request has been received by the server,  but as long as it has not been rejected, the client should continue with the request. |
| 101 Switching Protocols | The server switches protocol. |

2xx: Successful

|  |  |
| --- | --- |
| **Message** | **Description** |
| 200 OK | The request is OK. |
| 201 Created | The request is complete, and a new resource is created . |
| 202 Accepted | The request is accepted for processing, but the processing is not complete. |
| 203 Non-authoritative Information | The information in the entity header is from a local or third-party copy, not from the original server. |
| 204 No Content | A status code and a header are given in the response, but there is no entity-body in the reply. |
| 205 Reset Content | The browser should clear the form used for this transaction for additional input. |
| 206 Partial Content | The server is returning partial data of the size requested. Used in response to a request specifying a *Range* header. The server must specify the range included in the response with the *Content-Range* header. |

3xx: Redirection

|  |  |
| --- | --- |
| **Message** | **Description** |
| 300 Multiple Choices | A link list. The user can select a link and go to that location. Maximum five addresses  . |
| 301 Moved Permanently | The requested page has moved to a new url . |
| 302 Found | The requested page has moved temporarily to a new url . |
| 303 See Other | The requested page can be found under a different url . |
| 304 Not Modified | This is the response code to an *If-Modified-Since* or *If-None-Match* header, where the URL has not been modified since the specified date. |
| 305 Use Proxy | The requested URL must be accessed through the proxy mentioned in the *Location* header. |
| 306 *Unused* | This code was used in a previous version. It is no longer used, but the code is reserved. |
| 307 Temporary Redirect | The requested page has moved temporarily to a new url. |

4xx: Client Error

|  |  |
| --- | --- |
| **Message** | **Description** |
| 400 Bad Request | The server did not understand the request. |
| 401 Unauthorized | The requested page needs a username and a password. |
| 402 Payment Required | *You can not use this code yet*. |
| 403 Forbidden | Access is forbidden to the requested page. |
| 404 Not Found | The server can not find the requested page. |
| 405 Method Not Allowed | The method specified in the request is not allowed. |
| 406 Not Acceptable | The server can only generate a response that is not accepted by the client. |
| 407 Proxy Authentication Required | You must authenticate with a proxy server before this request can be served. |
| 408 Request Timeout | The request took longer than the server was prepared to wait. |
| 409 Conflict | The request could not be completed because of a conflict. |
| 410 Gone | The requested page is no longer available . |
| 411 Length Required | The "Content-Length" is not defined. The server will not accept the request without it . |
| 412 Precondition Failed | The pre condition given in the request evaluated to false by the server. |
| 413 Request Entity Too Large | The server will not accept the request, because the request entity is too large. |
| 414 Request-url Too Long | The server will not accept the request, because the url is too long. Occurs when you convert a "post" request to a "get" request with a long query information . |
| 415 Unsupported Media Type | The server will not accept the request, because the mediatype is not supported . |
| 416 Requested Range Not Satisfiable | The requested byte range is not available and is out of bounds. |
| 417 Expectation Failed | The expectation given in an Expect request-header  field could not be met by this server. |

5xx: Server Error

|  |  |
| --- | --- |
| **Message** | **Description** |
| 500 Internal Server Error | The request was not completed.  The server met an unexpected condition. |
| 501 Not Implemented | The request was not completed.  The server did not support the functionality required. |
| 502 Bad Gateway | The request was not completed.  The server received an invalid response from the upstream server. |
| 503 Service Unavailable | The request was not completed. The server is temporarily overloading or down. |
| 504 Gateway Timeout | The gateway has timed out. |
| 505 HTTP Version Not Supported | The server does not support the "http protocol" version. |

