

THE BEAUTY OF WEB INFRASTRUCTURE



Web infrastructure is one among many of the impeccable designs to exist among mankind. As some say, engineering is an art, and this is one of the ingenuities that showcases perfection and detailing.

The web is built like in empire. Fences (Firewalls) are meant to protect the people (client). The elected (web servers, databases, application servers) are meant to serve the people. The supervisors (load balancers) are supposed to distribute work to the ones responsible. Isn't this amazing?

This is just but a good enough introduction to show you what we mean by web infrastructure. But let's dive deeper into all this:

1. **Client**

The client is you, or your browser, your computer, or whatever is trying to access the internet on your behalf.

Computers have an IP address and this helps them to communicate among one another. So, to access the internet, you normally enter a URL (example google.com) and this corresponds to an IP address of a particular **web server**. The web server serves you the web page of google.com and boom! You have all you needed.



2. **TCP/IP**

TCP means transmission control protocol while IP means internet protocol. These two are used to establish a connection between two hosts so that messages/packets can be sent back and forth.

A host is just like a client, your computer is a **localhost**.

Take the IP as a postal address. It manages addressing of a message and how the message will travel.

Take TCP as the postal address's car. TCP will be in charge of transporting internet packets, or the message in short.

If you request for a web page (google.com), the web page is transported from the web server to your browser using TCP.

3. **Web server**

This is a software that delivers web pages. Ever wondered how you type google.com and get a page out of nowhere, yet it's not stored in your computer? Web servers are responsible for fetching web pages from a storage platform called a **database** and delivering them to you.

The web server can be both hardware and software. As a hardware, it's the computer that stores web server software, and connects to the internet to exchange data with other devices.

4. **Database**

Every institution has a filing system where they store their critical data. They ensure it can be easily accessed at any time. The web as well has its "filing system" called a database. A database is like a digital filing cabinet that stores and organizes information in a structured way, making it easy to find, retrieve, and manage data.

Let's assume an organization that carries out census collects data on names, age and cities people live in and then posts the results on the web. This data will have to be stored somewhere, won't it? This is where database comes in. Here is a sample database (MySQL):

```
mysql> SELECT * FROM class;
+-----+-----+-----+
| student_id | name | age |
+-----+-----+-----+
|          1 | John |  30 |
+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

5. Load balancer

No one expects a whole company work to be done by one person when there are other employees available. As data states, 400,000 google searches are conducted every minute. This is a lot of traffic for our web servers that need to serve web pages to us, and sometimes we can be very impatient.

Thanks to load balancers, they help distribute traffic to multiple servers.

Some special algorithms like round robin are used to do this.

Let's assume there are 5 web servers in google and also, let's assume google uses the 'least connection first scheduling algorithm'. If you request for a web page, google.com, the algorithm will check for the web server with least number of connections, then direct your traffic to it. This ensures the shortest response time to client requests.

6. Firewalls

As the name states, they are walls, however, the 'fire' just shows how serious they are at their job. Firewalls are a protective barrier between a private network and an external network, often the internet. They operate by enforcing rules that allow, limit or block traffic based on pre-defined criteria.

7. HTTP/SSL

a. HTTP

So, what is <http://google.com> ? In full, HTTP is hypertext transfer protocol. It defines how web browsers and web server communicate to retrieve and display web pages.

When you type a URL and press enter, a http request is sent to the web servers. It contains information about: the browser you are using, your OS, the specific page you want and what your browser accepts (maybe your browser wants text only for example)

The http response from the server will contain: the data being sent, when the data was last modified, how big the data is and information about the server that served you. This is a sample http response:

```
HTTP/1.1 200 OK
Server: nginx/1.18.0 (Ubuntu)
Date: Thu, 14 Sep 2023 14:19:38 GMT
Content-Type: text/html
Content-Length: 13
Last-Modified: Thu, 07 Sep 2023 10:38:30 GMT
Connection: keep-alive
ETag: "64f9a826-d"
X-Served-By: 117256-web-01
Accept-Ranges: bytes
```

b. **SSL and HTTPS**

SSL means secure socket layer. SSL works with HTTPS which means HTTP secure. When you visit a website that uses HTTPS, your data exchange is encrypted. This means that anyone who wants to intercept the exchange must have an encryption key to do this.

SSL are certificates that are used to verify the authenticity of the website you want to visit. It contains a public and private key. The private key is stored securely in a web sever while the public key is given to anyone who needs it.

Your browser comes with some pre-installed list of trusted root SSL certificates. If you connect to the web, the web server sends its SSL certificate to your browser and it is the browsers duty to cross check that against it root certificates. If its valid, your browser trusts it. The browser now uses its public key to initiate encryption key exchange with the web server. This involves generating a symmetric encryption key that will be used for secure communication.

Well, hope you learnt something new today!