How does the Internet work?
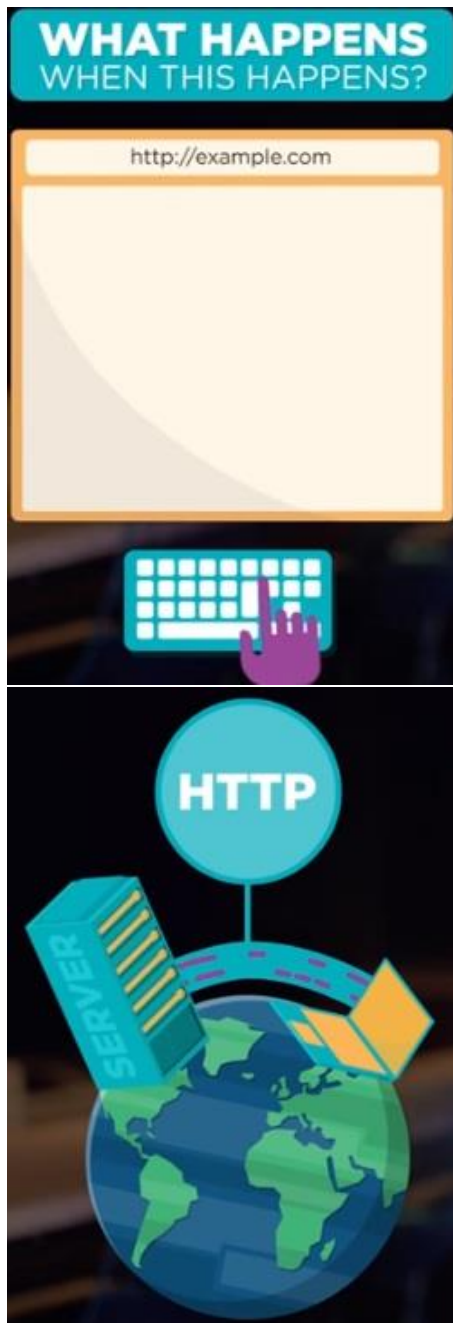
HTTP and HTML

HTTP is the standard protocol using which the webpages are transferred over the internet.

Let's look at web browsing.

First, you open a web browser. It's the app you use to access the web pages. Next, you type in the web address, or URL, which stands for Uniform Resource Locator of the website.

So you've probably wondered what actually happens when you type an address into your web browser and then hit enter. In that moment, your computer starts talking to another computer, called a server, that's thousand of miles away. And in milliseconds your computer asks that server for a website, and that server starts to talk back to your computer in a language called HTTP.
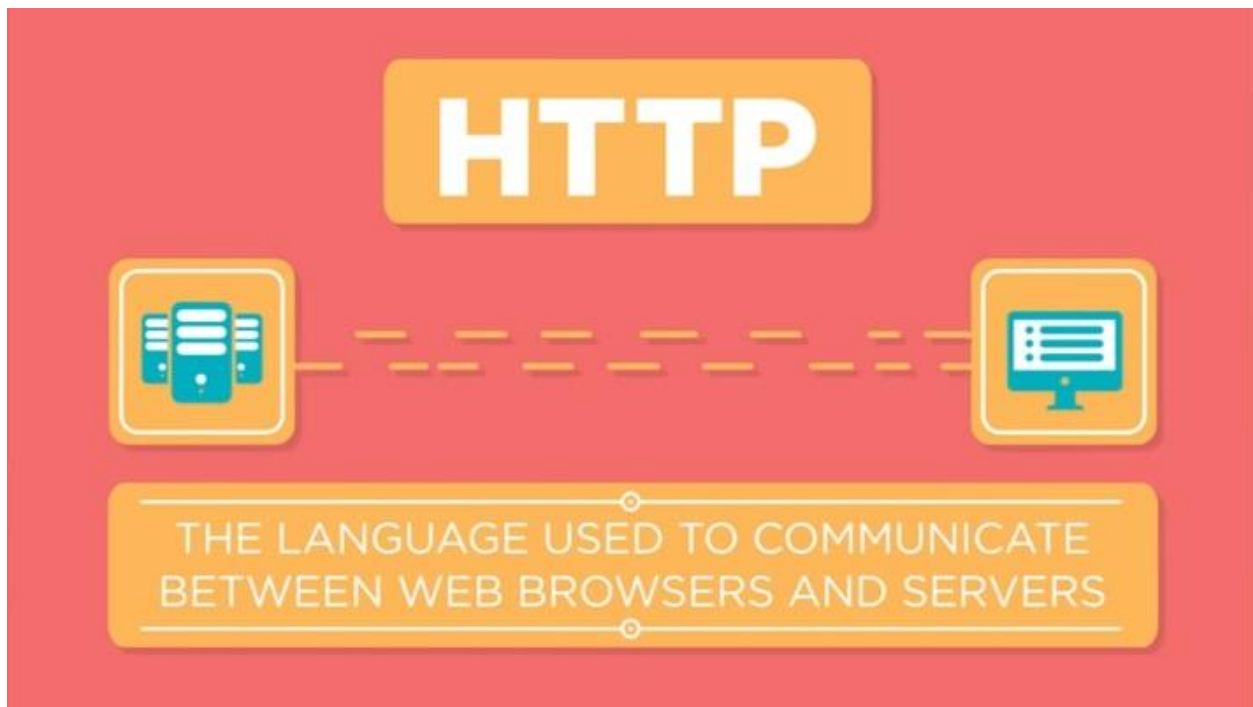
HTTP stands for `HyperText Transfer Protocol`. You can kind of think of it as the language that one computer uses to ask another computer for a document.
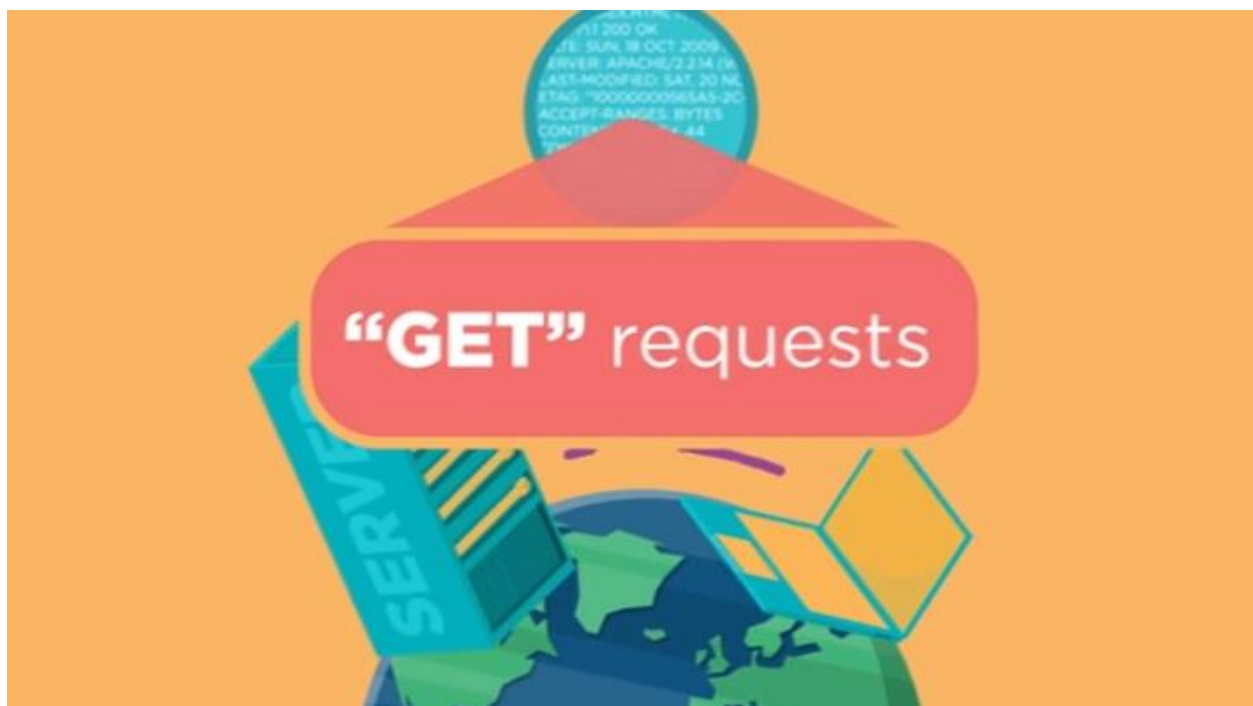
It's actually really pretty straight forward.

If you were to intercept the conversation between your computer and a web server on the internet, it's mainly made up something called `GET requests`. Those are really very simply the word GET and the name of the document that you're requesting.

So if you try to login into Tumblr and load Tumblr login page, all you're doing is sending a GET request to Tumblr's server that says GET /login. And that tells Tumblr's server that you want all of the HTML code for the Tumblr login page.

So HTML stands for `HyperText Markup Language` and you can think of that as the language you use to tell a web browser how to make a page look.

If you think about something like Wikipedia, which is really just a big simple document and HTML is the language that you use to make that title big and bold, to make the font the right font, to link certain text to certain other pages, to make some text bold, to make some text italic, to put an image in the middle of the page, to align the image to the right etc.

The text of a web page is included directly in the HTML, but other parts like images or videos are separate files with their own URL's that need to be requested.
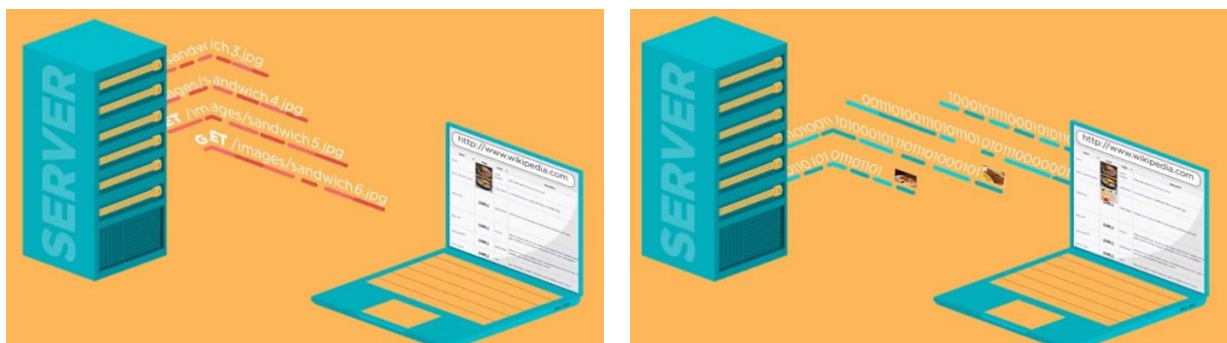
The browser sends separate HTTP requests for each of these and displays them as they arrive. If a web page has a lot of different images, each of them causes a separate HTTP request and the page loads slower.

Now sometimes when you browse the web, you're not just requesting pages with GET requests.

Sometimes you send information like when you fill out a form or type a search query. Your browser sends this information in plain text to the web server using an `HTTP POST request`.

Let's say you log in to Tumblr.

Well the first thing you do is make a POST request that is a POST to Tumblr's login page that has some data attached to it. It has your email address, it has your password.
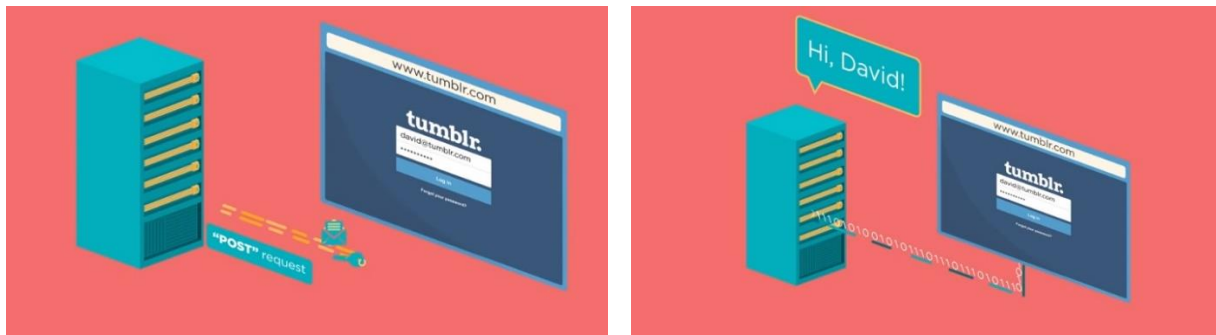
That goes to Tumblr's server. Tumblr's server figures out that okay, you're David. It sends a web page back to your browser that says, Success! Logged in as David.
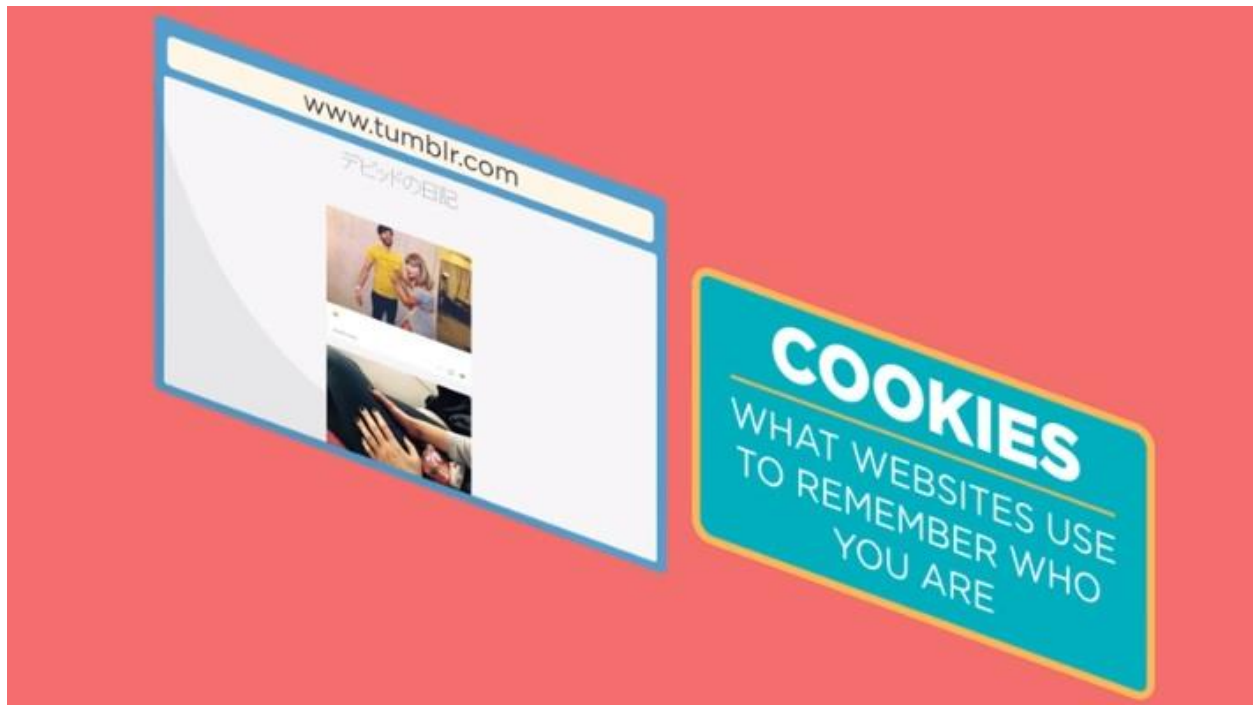
But along with that web page, it also attaches a little bit of invisible `cookie data` that your browser sees and knows to save. And it's really important because it's really the only way that a website can remember who you are.

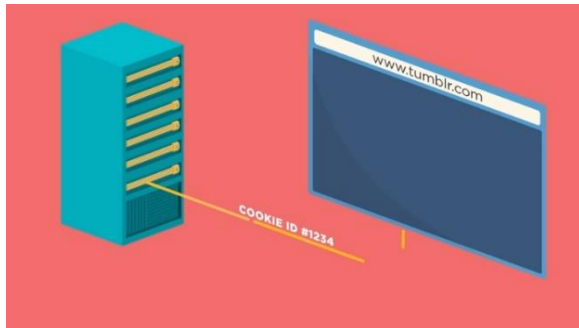All that cookie data really is an ID card for Tumblr. It's a number that identifies you as David. And your web browser holds on to that number and the next time you refresh Tumblr, the next time you go to Tumblr.com, your browser knows to automatically attach that ID number with the request that it sends over to Tumblr's servers.

So now Tumblr's servers sees the request coming from your browser, sees the ID number, and knows "Ok, this is a request from David".
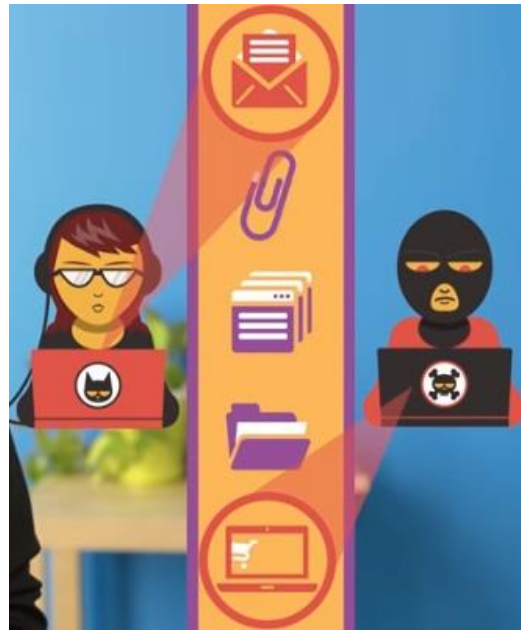
The internet is completely open. All of its connection are shared and information is sent in plain text. This makes it possible for hackers to snoop on any personal information that you send over the internet.
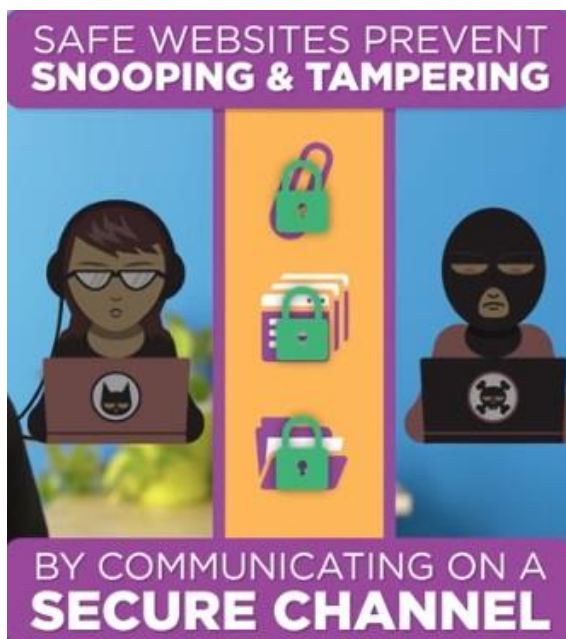
But safe websites prevent this by asking your web browser to communicate on a secure channel using something called `Secure Sockets Layer` and its successor Transport Layer Security.

You can think of SSL and TLS as a layer of security wrapped around your communications to protect them from snooping or tampering. SSL and TLS are active when you see the little lock that appears in your browser address bar, next to the HTTPS.

The HTTPS protocols ensure that your HTTP requests are secure and protected.

When a website asks your browser to engage in a secure connection, it first provides a digital certificate which is like an official ID card proving that it's the website it claims to be.
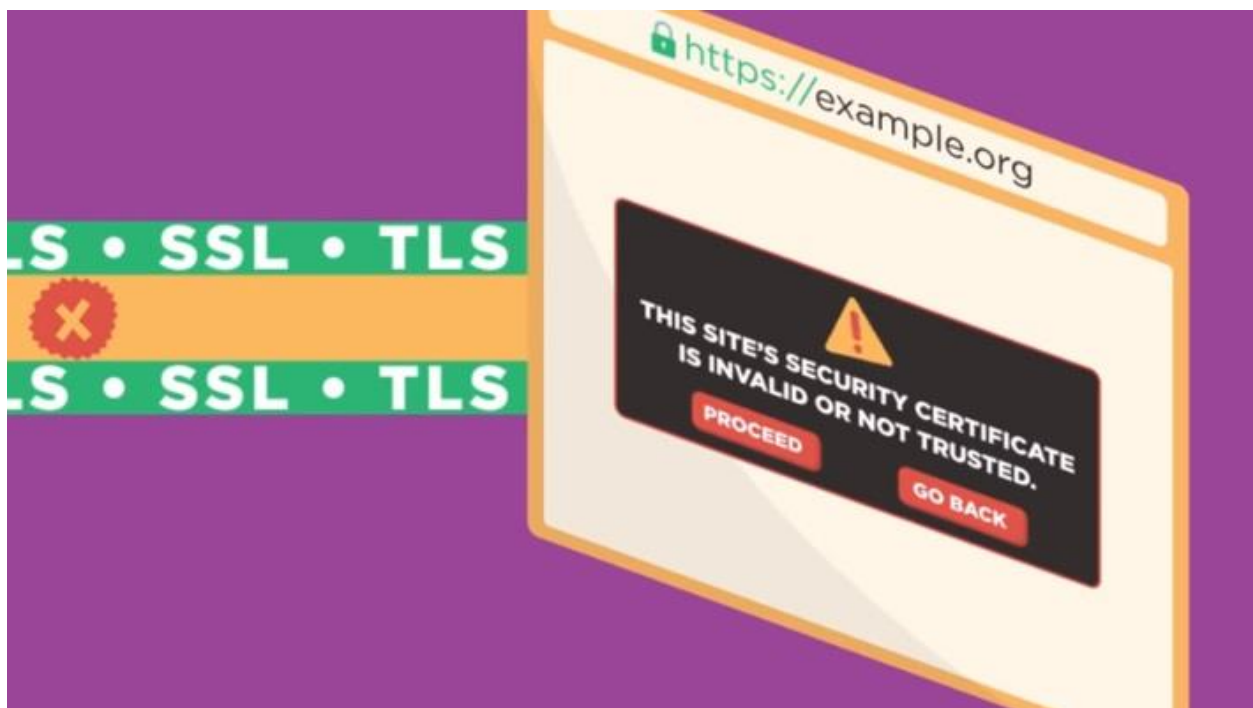

Digital certificates are published by certificate authorities, which are trusted entities that verify the identities of websites and issue certificates for them. Just like a government can issue ID's or passport.
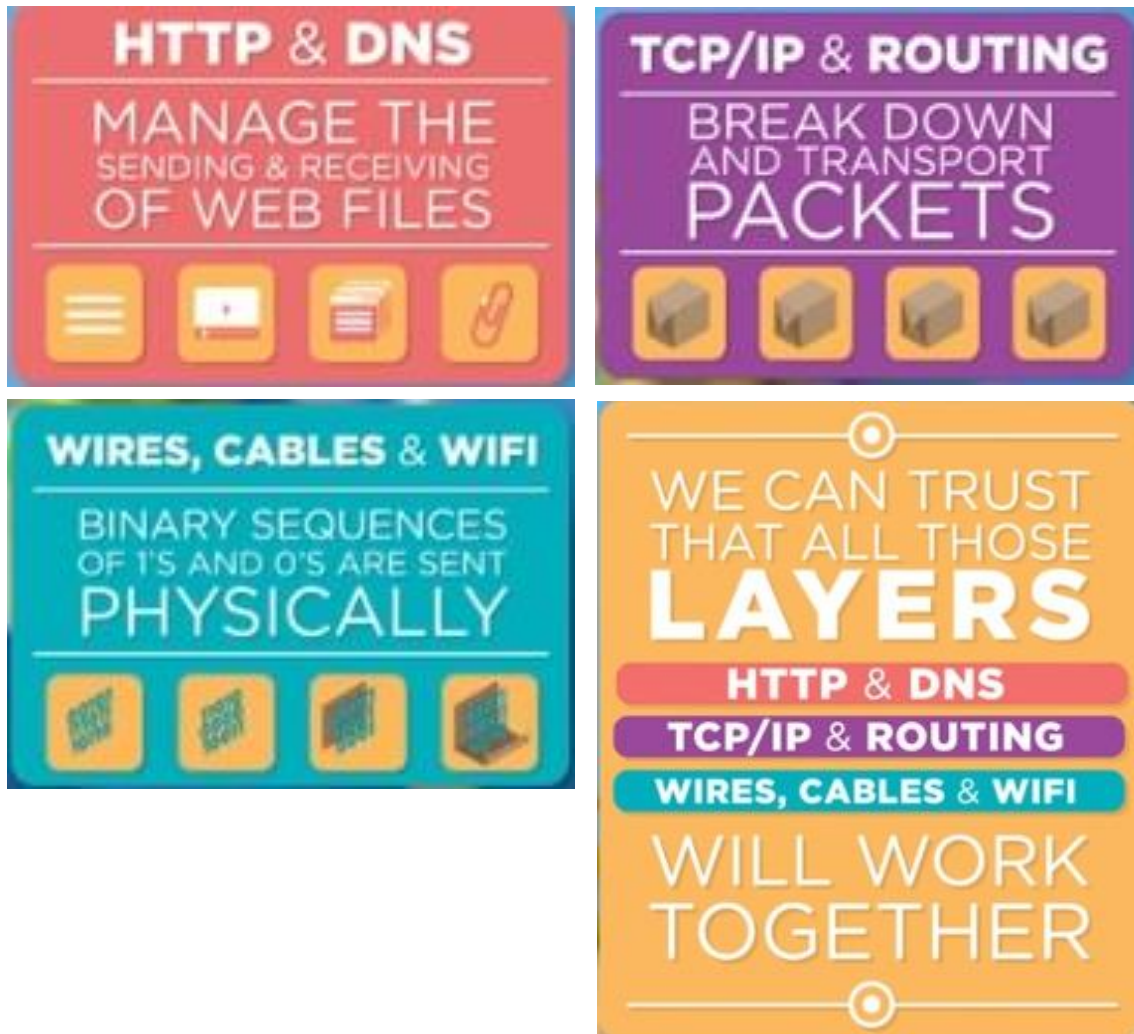
Now if a website tries to start a secure connection without a properly issued digital certificate, your browser will warn you.

That's the basic of web browsing! The part of the internet we see day to day.

Summary

HTTP and DNS manage the sending and receiving of HTML, media files, or anything on the web. What makes this possible under the hood are TCP/IP and router networks that break down and transport information in small packets. Those packets themselves are made up of binary, sequences of 1 and 0 that are physically sent through electric wires, fiber optic cables, and wireless networks.

Fortunately, once you've learned how one layer of the internet works, you can rely on it without remembering all the details. And we can

trust that all those layers will work together successively deliver information at scale and with reliability.

Source

How does the Internet work?

[The Internet: HTTP & HTML](#)