The **domain name system (DNS)** gives us humans an easy way to identify where we want to go on the Internet.

**Anatomy of a domain name**

Each domain name is made up of parts:

***third-level-domain.second-level-domain.top-level-domain***

* There are a limited set of top-level domains (TLDs), and many websites use the most common TLDs,".com", ".org", and ".edu".
* The second level domain is unique to the company or organization that registers it, like "wikipedia" or "khanacademy".
* The third level domain is also called a subdomain, because it's owned by the same group and that URL often directs you to a subset of the website, like "m.wikipedia.org" (mobile-optimized Wikipedia) or "es.khanacademy.org" (Spanish-language Khan Academy).

**Domains ↔ IP addresses**

Behind the scenes, each domain name maps to an IP address. When we type a URL in the address bar of our browser, the computer has to figure out its IP address.

The computer can't store a database of more than 300 million domain names locally, so it goes through a multi-step process to find out the IP address.

#### Step 1: Check the local cache

If you've visited a website once, there's a fairly good chance you'll visit it again. That's why computers keep their own local cache of domain name to IP mappings. The cache stays small, because it kicks out domains you haven't visited in a while or domains that send down expiration dates.

#### Step 2: Ask the ISP cache

Every ISP provides a domain name resolving service and keeps its own cache. Perhaps you haven't visited a particular website, but your neighbor just did, so the ISP can lookup the IP from their visit.

If it's not in the ISP's cache, then it's off to the next step.

#### Step 3: Ask the name servers

There are domain name servers scattered around the globe that are responsible for keeping track of a subset of the millions of domain names.

The servers are ordered in a hierarchy:

Root name servers → TLD name servers → Host name servers.

The ISP starts by asking the root name servers: "hey, which name server knows about .org domains?" The root name server responds with the IP address of a TLD name server that tracks ".org" domains.

Next, the ISP asks the TLD name server: "so, who knows about wikipedia domains?" The TLD name server responds with the IP address of a host name server that contains the "wikipedia" records.

Finally, the ISP asks the host name server: "okay, so where's www.wikipedia.org?" The host name server responds with an exact IP address.

**DNS Spoofing**

The domain name system is scalable, but it is not always secure. Cyber criminals figured out a way to exploit flaws in DNS name servers, in an attack known as **DNS spoofing** or **DNS cache poisoning.**

As we saw above, a domain resolver service must ask name servers when it doesn't already know the mapping of a domain to an IP.

If a cyber-criminal manages to take control of a name server or redirect requests to its own server, then it can reply with any IP address it wants:

The domain resolver now stores the new IP in its cache and sends that IP back to the requesting computer. The IP address often redirects users to a page that will download computer viruses or ask for their secure information.

DNS spoofing can be prevented. The **DNSSEC protocol** extends the original DNS protocol and specifies the best way for DNS resolvers to authenticate the information sent to them.