

Title:

How the Internet actually works

Word Count:

3659

Summary:

In this article I will explain how the Internet works, all the way from what goes through the wires and how the wires across the globe connect, to how meaningful activities are performed on your computer.

Unlike other Internet articles, I won't try to explain the history behind the Internet of today - it's complex enough, and like me, you probably don't care very much. I also won't be confusing you with highly technical explanations.

Keywords:

internet, tcpip, protocol, url, http, ip, address, modem, website, dns, nameserver, ipaddress, webpage, web, ssl

Article Body:

To most people, the Internet is the place to which everyone plugs in their computer and views webpages and sends e-mail. That's a very human-centric viewpoint, but if we're to truly understand the Internet, we need to be more exact:

<blockquote>The Internet is THE large global computer network that people connect to by-default, by virtue of the fact that it's the largest. And, like any computer network, there are conventions that allow it to work.</blockquote>

This is all it is really - a very big computer network. However, this article will go beyond explaining just the Internet, as it will also explain the 'World Wide Web'. Most people don't know the difference between the Internet and Web, but really it's quite simple: the Internet is a computer network, and the Web is a system of publishing (of websites) for it.

Computer networks

And, what's a computer network? A computer network is just two or more of computers connected together such that they may send messages between each other. On larger networks computers are connected together in complex arrangements, where some intermediary computers have more than one connection to

other computers, such that every computer can reach any other computer in the network via paths through some of those intermediary computers.

Computers aren't the only things that use networks - the road and rail networks are very similar to computer networks, just those networks transport people instead of information.

Trains on a rail network operate on a certain kind of track - such a convention is needed, because otherwise the network could not effectively work. Likewise, roads are designed to suit vehicles that match a kind of pattern - robust vehicles of a certain size range that travel within a certain reasonable speed range. Computers in a network have conventions too, and we usually call these conventions 'protocols'.

There are many kinds of popular computer network today. The most conventional by far is the so-called 'Ethernet' network that physically connects computers together in homes, schools and offices. However, WiFi is becoming increasingly popular for connecting together devices so that cables aren't required at all.

Connecting to the Internet

When you connect to the Internet, you're using networking technology, but things are usually a lot muddier. There's an apt phrase, "Rome wasn't built in a day" because neither was the Internet. The only reason the Internet could spring up so quickly and cheaply for people was because another kind of network already existed throughout the world - the phone network!

The pre-existence of the phone network provided a medium for ordinary computers in ordinary people's homes to be connected onto the great high-tech military and research network that had been developed in years before. It just required some technological mastery in the form of 'modems'. Modems allow phone lines to be turned into a mini-network connection between a home and a special company (an 'ISP') that already is connected up to the Internet. It's like a bridge joining up the road networks on an island and the mainland - the road networks become one, due to a special kind of connection between them.

Fast Internet connections that are done via '(A)DSL' and 'Cable' are no different to phone line connections really - there's still a joining process of some kind going on behind the scenes. As Arthur C. Clarke once said, 'any sufficiently advanced technology is indistinguishable from magic'.

The Internet

The really amazing about the Internet isn't the technology. We've actually had

big Internet-like computer networks before, and 'The Internet' existed long before normal people knew the term. The amazing thing is that such a massive computer network could exist without being built or governed in any kind of seriously organised way. The only organisation that really has a grip on the core computer network of the Internet is a US-government-backed non-profit company called 'ICANN', but nobody could claim they 'controlled' the Internet, as their mandate and activities are extremely limited.

The Internet is a testament both simultaneously due to the way technologists cooperated and by the way entrepreneurs took up the task, unmanaged, to use the conventions of the technologists to hook up regular people and businesses. The Internet didn't develop on the Microsoft Windows 'operating system' - Internet technology was built around much older technical operating systems; nevertheless, the technology could be applied to ordinary computers by simply building support for the necessary networking conventions on top of Windows. It was never planned, but good foundations and a lack of bottlenecks (such as controlling bodies) often lead to unforeseen great rises - like the telephone network before, or even the world-wide spread of human population and society.

What I have described so far is probably not the Internet as you or most would see it. It's unlikely you see the Internet as a democratic and uniform computer network, and to an extent, it isn't. The reason for this is that I have only explained the foundations of the system so far, and this foundation operates below the level you'd normally be aware of. On the lowest level you would be aware of, the Internet is actually more like a situation between a getter and a giver - there's something you want from the Internet, so you connect up and get it. Even when you send an e-mail, you're getting the service of e-mail delivery.

Being a computer network, the Internet consists of computers - however, not all computers on the Internet are created equal. Some computers are there to provide services, and some are there to consume those services. We call the providing computers 'servers' and the consuming computers 'clients'. At the theoretical level, the computers have equal status on the network, but servers are much better connected than clients and are generally put in place by companies providing some kind of commercial service. You don't pay to view a web site, but somebody pays for the server the website is located on - usually the owner of the web site pays a 'web host' (a commercial company who owns the server).

Making contact

I've established how the Internet is a computer network: now I will explain how two computers that could be on other sides of the world can send messages to each other.

Imagine you were writing a letter and needed to send it to someone. If you just wrote a name on the front, it would never arrive, unless perhaps you lived in a small village. A name is rarely specific enough. Therefore, as we all know, we use addresses to contact someone, often using: the name, the house number, the road name, the town name, the county name, and sometimes, the country name. This allows sending of messages on another kind of network - the postal network. When you send a letter, typically it will be passed between postal sorting offices starting from the sorting office nearest to the origin, then up to increasingly large sorting offices until it's handled by a sorting office covering regions for both the origin and the destination, then down to increasingly small sorting offices until it's at the sorting office nearest the destination - and then it's delivered.

In our postal situation, there are two key factors at work - a form of addressing that 'homes in' on the destination location, and a form of message delivery that 'broadens out' then 'narrows in'. Computers are more organised, but they actually effectively do exactly the same thing.

Each computer on the Internet is given an address ('IP address'), and this 'homes in' on their location. The 'homing in' isn't done strictly geographically, rather in terms of the connection-relationship between the smaller computer networks within the Internet. For the real world, being a neighbour is geographical, but on a computer network, being a neighbour is having a direct network connection.

Like the postal network with its sorting offices, computer networks usually have connections to a few other computer networks. A computer network will send the message to a larger network (a network that is more likely to recognise at least some part of the address). This process of 'broadening out' continues until the message is being handled by a network that is 'over' the destination, and then the 'narrowing in' process will occur.

An example 'IP address' is '69.60.115.116'. They are just series of digit groups where the digit groups towards the right are increasingly local. Each digit group is a number between 0 and 255. This is just an approximation, but you could think of this address meaning:

- A computer 116
- in a small neighbourhood 115
- in a larger neighbourhood 60
- controlled by an ISP 69
- (on the Internet)

The small neighbourhood, the larger neighbourhood, the ISP, and the Internet, could all be considered computer networks in their own right. Therefore, for a message to the same 'larger neighbourhood', the message would be passed up towards one of those intermediary computers in the larger neighbourhood and then back down to the correct smaller neighbourhood, and then to the correct computer.

Getting the message across

Now that we are able to deliver messages the hard part is over. All we need to do is to put stuff in our messages in a certain way such that it makes sense at the other end.

Letters we send in the real world always have stuff in common - they are written on paper and in a language understood by both sender and receiver. I've discussed before how conventions are important for networks to operate, and this important concept remains true for our messages.

All parts of the Internet transfer messages written in things called 'Packets', and the layout and contents of those 'packets' are done according to the 'Internet Protocol' (IP). You don't need to know these terms, but you do need to know that these simple messages are error prone and simplistic. You can think of 'packets' as the Internet equivalence of a sentence - for an ongoing conversation, there would be many of them sent in both directions of communication.

Getting the true message across

All those who've played 'Chinese whispers' will know how messed up ('corrupted') messages can get when they are sent between many agents to get from their origin to their destination. Computer networks aren't as bad as that, but things do go wrong, and it's necessary to be able to automatically detect and correct problems when they do.

Imagine you're trying to correct spelling errors in a letter. It's usually easy to do because there are far fewer words than there are possible word-length combinations of letters. You can see when letter combinations don't spell out words ('errors'), and then easily guess what the correct word should have been.

It really does work.

Errors in messages on the Internet are corrected in a very similar way. The messages that are sent are simply made longer than they need to be, and the extra space is used to "sum up" the message so to speak - if the "summing up"

doesn't match the message an error has been found and the message will need to be resent.

In actual fact, it is often possible to logically estimate with reasonable accuracy what was wrong with a message without requiring resending.

Error detection and correction can never be perfect, as the message and "summing up" part could be coincidentally messed-up so that they falsely indicate nothing went wrong. The theory is based off storing a big enough "summing up" part so that this unfortunate possibility is so unlikely that it can be safely ignored.

Reliable message transfer on the Internet is done via 'TCP'. You may have heard the term 'TCP/IP': this is just the normal combination of 'IP' and 'TCP', and is used for almost all Internet communication. IP is fundamental to the Internet, but TCP is not - there are in fact other 'protocols' that may be used that I won't be covering.

Names, not numbers

When most people think of an 'Internet Address' they think of something like 'www.ocportal.com' rather than '69.60.115.116'. People relate to names with greater ease than numbers, so special computers that humans need to access are typically assigned names ('domain names') using a system known as 'DNS' (the 'domain name system').

All Internet communication is still done using IP addresses (recall '69.60.115.116' is an IP address). The 'domain names' are therefore translated to IP addresses behind the scenes, before the main communication starts.

At the core, the process of looking up a domain name is quite simple - it's a process of 'homing in' by moving leftwards through the name, following an interrogation path. This is best shown by example - 'www.ocportal.com' would be looked up as follows:

- Every computer on the Internet knows how to contact the computers (the 'root' 'DNS servers') responsible for things like 'com', 'org', 'net' and 'uk'. There are a few such computers and one is contacted at random. The DNS server computer is asked if they know 'www.ocportal.com' and will respond saying they know which server computer is responsible for 'com'.
- The 'com' server computer is asked it knows 'www.ocportal.com' and will respond saying they know which server computer is responsible for 'ocportal.com'.
- 'The 'ocportal.com' server computer is asked if it knows 'www.ocportal.com' and will respond saying that it knows the corresponding server computer to be

'69.60.115.116'.

Note that there is a difference between a server computer being 'responsible' for a domain name and the domain name actually corresponding to that computer. For example, the 'ocportal.com' responsible DNS server might not necessarily be the same server as 'ocportal.com' itself.

As certain domain names, or parts of domain names, are very commonly used, computers will remember results to avoid doing a full interrogation for every name they need to lookup. In fact, I have simplified the process considerably in my example because the looking-up computer does not actually perform the full search itself. If all computers on the Internet did full searches it would overload the 'root DNS servers', as well as the DNS servers responsible for names like 'com'. Instead, the looking up computer would ask it's own special 'local DNS server', which might remember a result of a partial result, or might solicit help (full, or partial) from it's own 'local DNS server', and so on - until, in a worst case scenario, the process has to be completed in full.

Domain names are allocated by the person wanting them registering the domain name with an agent (a 'registrar') of the organisation responsible for the furthest right-hand part of the domain name. At the time of writing a company named 'VeriSign' (of which 'Network Solutions' is a subsidiary) is responsible for things like 'com' and 'net'. There are an uncountable number of registrars operating for VeriSign, and most domain purchasers are likely not aware of the chain of responsibility present - instead, they just get the domains they want from the agent, and deal solely with that agent and their web host (who are often the same company). Domains are never purchased, but rather rented and exclusively renewable for a period a bit longer than the rental period.

Meaningful dialogue

I've fully covered the essence of how messages are delivered over the Internet, but so far these messages are completely raw and meaningless. Before meaningful communication can occur we need to layer on yet another protocol (recall IP and TCP protocols are already layered over our physical network).

There are many protocols that work on the communications already established, including:

- HTTP - for web pages, typically read in web browser software
- POP3 - for reading e-mail in e-mail software, with it stored on a user's own computer
- IMAP4 - for reading e-mail in e-mail software, with it archived on the

receiving server
SMTP - for sending e-mail from e-mail software
FTP - for uploading and downloading files (sometimes via a web browser, although using special FTP software is better)
ICMP - for 'pinging', amongst other things (a 'ping' is the Internet equivalent to shouting out a 'are you there')
MSN Messenger - this is just one example of many protocols that aren't really standard and shared conventions, but rather ones designed by a single software manufacturer wholly for the purposes of their own software

I'm not going to go into the details of any of these protocols because it's not really relevant unless you actually need to know it.

The information transferred via a protocol is usually a request for something, or a response for something requested. For example, with HTTP, a client computer requests a certain web page from a server via HTTP and then the web server, basically, responds with the file embedded within HTTP.

Each of these protocols operates on more or more so-called 'ports', and it is these 'ports' that allow the computers to know which protocol to use. For example, a web server (special computer software running on a server computer that serves out web pages) uses a port of number '80', and hence when the server receives messages on that port it passes them to the web server software which naturally knows that they'll be written in HTTP.

For a client computer it's simpler - it knows that a response to a message it sent will be in the same protocol it initially used. When the messages are sent back and forth the server computer and client computer typically set up a so-called 'stream' (a marked conversation) between them. They are then able to associate messages to the stream according to their origin address and port number.

The World Wide Web

I've explained how the Internet works, but not yet how the 'World Wide Web' (the 'web') works. The web is the publishing system that most people don't realise is distinguishable from the Internet itself.

The Internet uses IP addresses (often found via domain names) to identify resources, but the web has to have something more sophisticated as it would be silly if every single page on the Internet had to have it's own 'domain name'. The web uses 'URLs' (uniform resource locators), and I'm sure you know about these as nowadays they are printed all over the place in the real world (albeit, usually only in short-hand).

A typical URL looks like this:

<blockquote><protocol>;//<domain-name_OR_ip-address>/<resource_identifier></blockquote>

For example:

<blockquote>http://www.ocportal.com/index.php</blockquote>

That said that's not really a full URL, because occasionally URLs can be much more complex. For example:

<blockquote><protocol>;//<user>:<password>@<domain/ip>;<port>/<resource_identifier></blockquote>

You can ignore the more complex example, because it's not really relevant for the purposes of this article.

HTTP is the core protocol for the web. This is why URLs usually start 'http://'. Web browsers almost always also support FTP, which is why some URLs may start 'ftp://'.

Typically the 'resource identifier' is simply a file on the server computer. For example, 'mywebsite/index.html' would be a file on the server computer of the same path, stored underneath a special directory. On Windows the "\" symbol is used to write out directory names, but as the web wasn't invented for Windows, the convention of the older operating systems is used.

We now have three kinds of 'Internet Address', in order of increasing sophistication:

IP addresses

Domain names

URLs

If a URL were put into web browser software by a prospective reader then the web browser would send out an appropriate request (usually, with the HTTP protocol being appropriate) to the server computer identified by the URL. The server computer would then respond and typically the web browser would end up with a file. The web browser would then interpret the file for display, much like any software running on a computer would interpret the files it understands. For the HTTP protocol, the web browser knows what to interpret the file as because the HTTP protocol uses something called a 'MIME type' to identify each kind of resource the server can send out. If the web server computer is just sending out an on-disk file then the web server computer works out the MIME type from the file extension (such as '.html') of the file.

An 'HTML' file is the kind of file that defines a web page. It's written in plain text, and basically mixes information showing how to display a document along with the document itself. If you're curious, try using the "View page source" function of your web browser when viewing a web page, and you'll see a

mix of portions of normal human text and short text between '<' and '>' symbols. The former is the document contents and the latter are the display instructions.

In newer versions of HTML there's a split between 'structuring' a document and 'displaying' a structure - in this case, another special technology named 'CSS' is added to the mix.

I've explained how typical web pages are just files on the disk of a server computer. Increasingly, things are slightly less direct. When you visit something like eBay, your web-mail, or an ocPortal-powered website, you aren't just reading files. You're actually interacting with computer software, and the web pages you receive are generated anew by that software every time a request is made. These kinds of systems are known as 'web applications' and are increasingly replacing the need to install software on your own computer (because it's so much easier just to use a web browser to access a web application on a server computer).