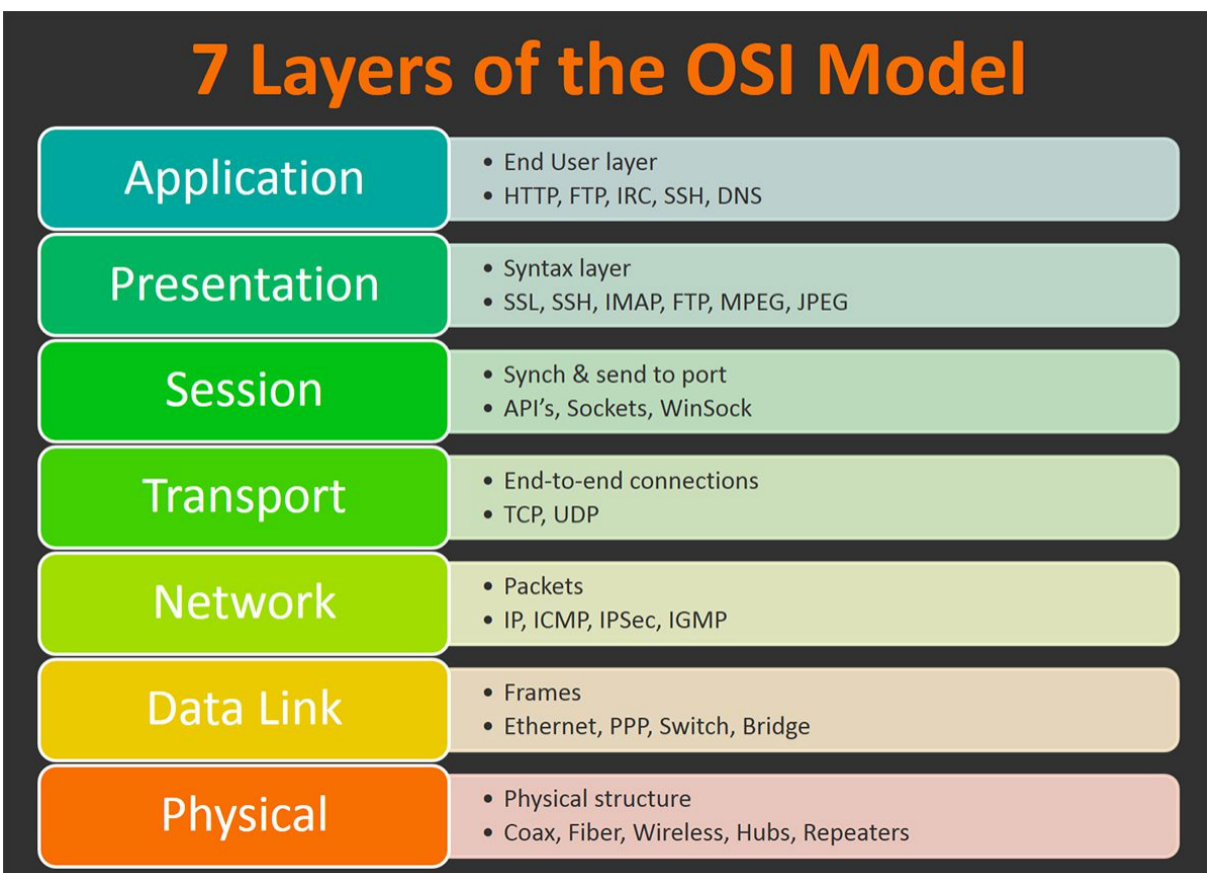


How the internet works Continued

Last week we talked briefly on how the internet works. In this lecture we will cover some more of the basics on how things work.

The internet is made up of layers:



The physical layer is the actual machines,

computers and devices that we connect to the internet.

The datalink layer is the linkage, the actual infrastructure that the earth is entangled in. Physical (fiberoptic) cables that connect continents together, then down the line to data centres, institutions, all the way down to your neighbourhood, house, router and device. These fiberoptic cables send light.

The network layer is responsible for routing data via the best possible physical path. This is based on a number of factors. Best physical path, congestion of data, priority of service

The Transport layer ensures the complete and reliable delivery of data packets. When data is transmuted it is split up into multiple data packets and delivered piece by piece. Each piece includes a header which shows

where it is from, where it is going and what order to piece it together. This layer also provides mechanisms for errors, flow control, congestion mitigation, duplication checks.

The Session layer is the first layer that deals with software and manages sessions between servers to coordinate communication. A session is any time there is an interactive data exchange between 2 entities within a network. For example when there is an HTTPS session that allows a user to visit a website and browse for a specific time period. This layer is what opens, closes, re-opens a session as well as authentication and authorization of communication between specific applications and servers.

The Presentation layer converts data

formats between applications and networks. This can include data conversion, data compression, encryption and decryption. For example: Data transferred from an encrypted application is formatted and encrypted at this layer before it is sent across the network. At the receiving end it is decrypted and formatted back to how it is intended.

The Application layer interacts directly with end users to provide support for email, data sharing, file transfers and so on. For example: FTP, SMTP, DNS.

What is a server?

A server is a machine that can host resources and allow clients to access said resources. Physically, servers are tall racks of machines. They run operating systems and you as a developer can choose what

OS you want to work with. You must keep in mind that depending on what you want to do will correlate with what server you will choose. For the most part, linux servers are the norm.

What is a client?

A client is software that (usually) connects to the server to perform actions. The client provide a user interface that allows users to carry out actions. It forwards these requests to the server, which carries out the action and returns a response.



Cpanel - This is a server side back end control panel for a hosting account. When you rent space on a server sometimes it will have this. Here you can manage email addresses, files, databases, domains, security, server side software (things like PHP), check metrics and so on. This tool is crucial for a webmaster. It can be accessed by going to `yoursite.com/cpanel` or by using the port `yoursite.com:2082`

Port - Ports identify and forward one thing to another when it arrives at a server. Like when we use port 2082 to go to the Cpanel the server knows which port/process we want to arrive at and directs us there.

WHM - Stands for Web Host Manager This is what we use to manage vded or fully dedicated servers. It is a powerful tool and if you don't know what you are doing you

could break an entire server hosting many websites. We will not get into this yet. For now you will not need to know anything to do with this as we are only learning about smaller scale web-mastery.

ISP - Internet Service Provider. Who supplies us our internet access. (Cogeco Rogers, Bell, Verizon so on).

Domain Name - This is what we purchase from a registrar so that we can provide a symbolic representation like an easily memorable name to a numerical number. If humans were good at remembering numbers we wouldn't need these, but because we use words, we use domain names and resolve them to IP addresses.

Top Level Domain (TLD) - A TLD is a the extension like .com .net .gov .ca .org and so on. This used to be limited but now there

are all kinds. From countries to things and even .wtf is a TLD now. Some of these are restricted. Like when we use a .ca you have to be a Canadian to do so.

Second Level Domain (SLD) - Directly below a TLD in: example.com "example" is the SLD.

Sub Domain - If you own a domain name (example.com) you can make a whole new site within that domain using a sub domain (help.example.com). Depending on the shared hosting you have (and if you have access to cpanel) the procedure for this might differ. Can also be thought of like a third level domain. example.co.uk or example.on.ca

DNS - Domain Name System. This is a hierarchical naming system. It serves as a kind of phone book for the internet,

resolving domain names into IP addresses.

DNS Records - DNS records basically are a list of records that point things where to go Zone Records, Resource Records and so on.

A list of most common records is provided below:

- Address Mapping records (A) - Points the domain to an IP address
- IP Version 6 Address records (AAAA) Same as A record but uses a IPv6
- Canonical Name records (CNAME) Specify an alias domain name
- Mail exchanger record (MX) Resource for mail exchanger. You can port your emails through your own server or you can use google and point to that for example. (later lesson)
- Name Server records (NS) Points to a Name Server. NS1 NS2

IP - An Internet Protocol Address is something that every machine has. Your computer, phone, server, printer, robot and android (coming soon) all have one. Even though every machine has a unique IP, every ISP assigns a private IP to a network. So if at home you have 2 computers and 5 phones on that network, all of the machines on the network will share 1 private IP address. These private IP addresses prevent people from getting in to the network.

EG 69.89.31.226

Note Websites can be reached and viewed with ip addresses instead of the URL (unless the webmaster forces the IP to resolve to the URL in which case when you enter a website by its IP it will go to the site

and switch the IP to the domain name in the browser automatically)

URL - Uniform Resource Locator. It is a reference (address) to a resource (file on a server, website)

There are 2 parts to a URL:

A Protocol Identifier - http://

A resource Name - yoursite.com

FTP - Stands for File Transfer Protocol. This is how we have been connecting to our shared hosting accounts. It is just a protocol. It can also be used in place of www. in a browser.

Name Server - A name server is permanently connected to the internet and translates domain names to IP addresses. If

you have a vded or dedicated server you can make your own name server.

An example of a name server is:

NS1.EXAMPLE.COM
NS2.EXAMPLE.COM

[.htaccess](#) - htaccess is a configuration file for apache servers. Some shared hosting accounts allow the use of these and some do not. If you run a vded or dedicated then you can implement these at will. To do so, make a file in the public_html folder and name it ".htaccess" Then, in the text of that file you can enter commands. .htaccess files can also be places into sub directories. An .htaccess file is not a file extension. Rather, it is the full name of the file. Once placed into the root or subdir it will take effect, though sometimes you may have to wait a little bit for the server to clear its

cache. This can be advanced and if you don't know what you are doing you can cause internal server (500) errors which will break your site until fixed.

[php.ini](#) or [php5.ini](#) - This is another configuration file for running applications that require PHP. In here we can set upload size limits, memory limits and all kinds of other things.

[SEO](#) - Search Engine Optimization. This is how google and other search engines index our sites. SEO is comprised of the information on our sites (which can be written for SEO), the meta info (description in the meta tags):

```
<head>
<meta charset="UTF-8">
<meta name="description"
content="Description would go in here. The
```

contents would describe with 160 characters about the content within the page">

<meta name="keywords" content="In here you would add keywords separated by a comma. Its not as important as the description">

<meta name="author" content="The name of the author of the page content">

</head>

And the off site SEO (sites linking to your site) and how relevant the content is in the page, to the content on your page as well as many other aspects.

SSL - Secure Sockets Layer. This creates an encrypted link between a server and a browser. If you ever have a site that takes payments you might want to deploy this. To do so you have to purchase one from a provider (godaddy and the like). Prices vary

but don't be fooled by high-ballers. You can get a cheap one and use that.

SSH - Secure Shell allows you to connect via command line to your server. In mac you can do this using terminal.