Bueno, ya vamos avanzando en el mundo de las redes y adentrándonos a la teoría real. Nos falta mucho camino pero estamos acercándonos más y más.

Quería dar una mención especial a varios alumnos que no sólo están estudiando y tomando iniciativa sobre el tema, sino que también usan su creatividad para todo esto. Gracias y es hermoso ver como puedo ayudarlos.

Ya está. Comencemos con lo importante. **Modelos y protocolos**. Suena feo ¿Verdad? No suena nada interesante, pero en realidad sí lo es. Además de ser de vital importancia para nuestro entendimiento.

Vamos a empezar con los modelos. Se reconocen **2: modelo OSI y modelo TCP/IP**. Estos dos modelos **se dividen en capas** (aunque distintas, se usan para lo mismo). Veamos el modelo **OSI primero**.

Éste se divide en **7 capas**:

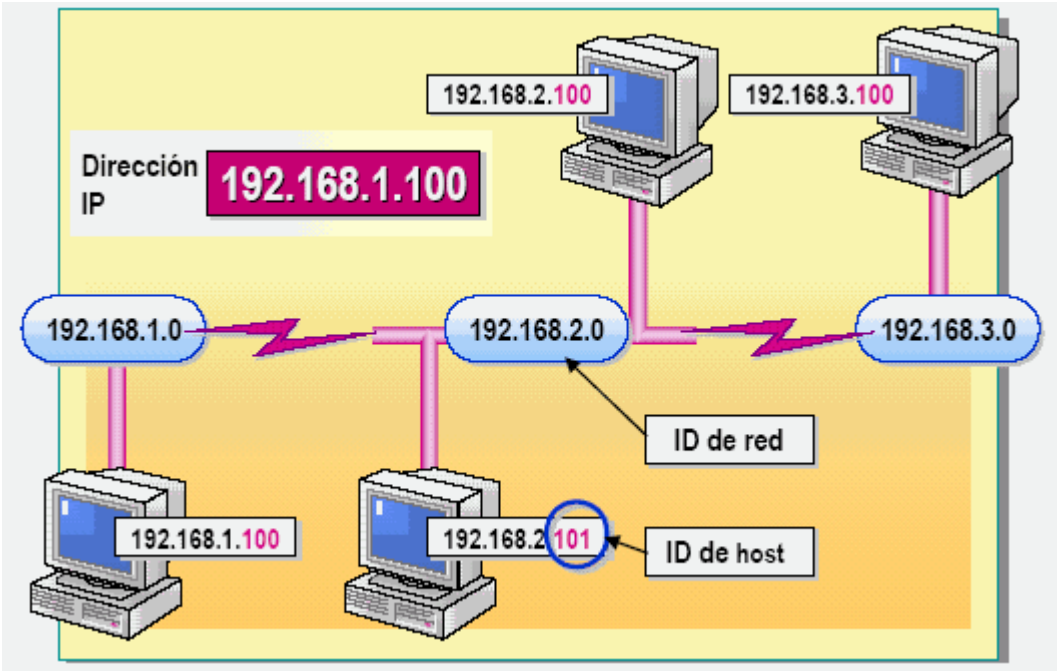
De abajo hacia arriba los numeramos de capa 1 (física) hasta la capa 7 (de aplicación).

Describamos una por una:

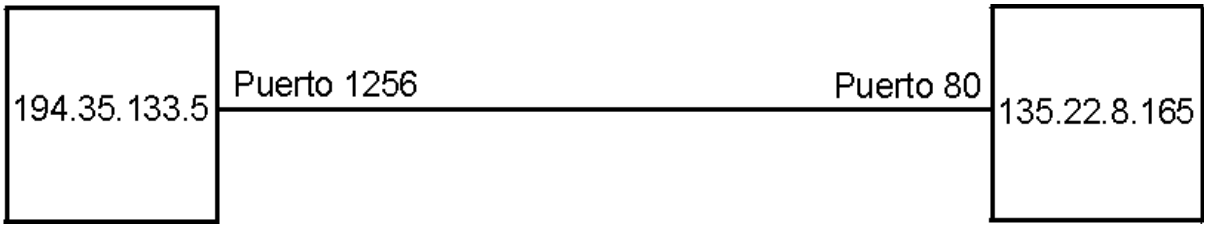
1-**Física**: Aquí sucede la **transmisión binaria** como **impulsos electrónicos** y es la parte donde los **cables** toman lugar.

2-**Enlace de datos**: Aquí sucede todo lo relacionado con las **MAC** y las **tarjetas de red**.

3-**Red**: **Direccionamiento lógico**. Todo lo que viene con el direccionamiento **IP** y la determinación de la ruta a utilizar por el medio.



4-**Transporte**: **Direccionamiento** de **puertos**. Destino y partida.



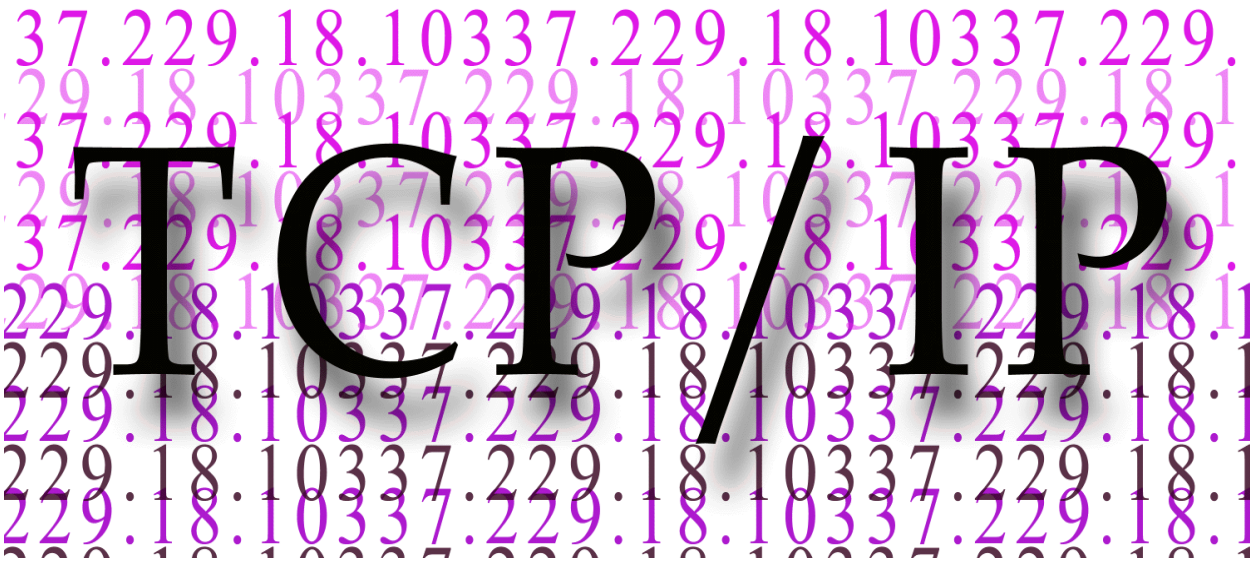
5-**Sesión**: **Controla** y **mantiene** una **sesión** establecida entre dos nodos.

6-**Presentación**: Esta capa, actúa como **traductor**. Para poder entender el mensaje y lograr un mensaje entendible para el receptor. Como si fuese el que cifra y el que descifra los datos (cada dispositivo tiene alguna forma de presentar las capas en si mismo)

7-**Aplicación**: Es el que le da a una **aplicación**, la **posibilidad** de **acceder** a los **servicios** de red, definiendo los protocolos de utilización.

Este modelo está bueno para representar varias cosas luego y para separar las administraciones. Sobre todo cuando varían entre las primeras 4 capas.

El otro modelo utilizado es el **TCP/IP.**



Este es mas compactado. Tiene **4 capas** solamente y se dividen de esta manera:

-**Acceso al medio**: donde juntamos las **primeras dos capas del modelo OSI**

-**Internet**: Muy similar a la **capa 3** del OSI.

-**Transporte**: **Capa 4** del OSI.

-**Aplicación**: Las **capas restantes** del OSI.“

"¿Nosotros vamos a tener que asegurar cada una de las capas que componen los modelos?”

Jajaja. **Sí Manolo.** Lo lamento si parece un arduo trabajo, pero te aseguro que separarlos en capas te ayuda mucho a la rapidez y al entendimiento.

“Bueno pero hay dos cosas que no entiendo. Los protocolos, y los puertos. ¿Qué es todo eso?”

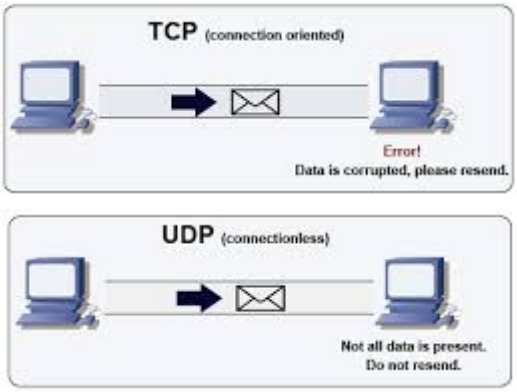
Bueno empezemos con los **protocolos**.

Para comunicarte con una persona, necesitas acordar una forma de hablar, un lenguaje, una velocidad, un tema de conversación y hasta una actitud. Todo esto se dice que es un **protocolo de comunicación**. Y lo mismo pasa con las computadoras, que tenemos un **lenguaje**, un **cifrado**, un **origen** y un **destino**, una velocidad de transferencia de datos, etc.

Pero en las redes, se pueden clasificar en **dos protocolos distintos**.

TCP: el cual se utiliza para las comunicaciones que no se quiere perder información. Es decir que **no importa si la velocidad cambia,** pero **sí importa la integridad** de la información. Se utiliza para chats de texto por ejemplo, donde no importa si tarda en llegar el mensaje,pero sí importa que esté completo. Imaginen que sino, uno puede llegar a recibir algunas letras o la mitad del mensaje.

UDP: que es lo contrario a TCP. Es decir que **no importa demasiado la integridad, sino que más importa la velocidad de transferencia.** Más o menos, los sistemas intentan reconstruir los paquetes de información dañados y los interpreta como puede. Cuando veamos la electrónica del asunto, veremos que diferencia puede haber entre una cosa y otra. Muy usado en el mundo de la comunicación audiovisual como las videoconferencias. Porque no es tan necesario que cada fotograma de imagen llegue perfecto, sino que es más vital que la comunicación sea en tiempo real. Si no imaginen que estarían hablando con minutos de diferencia y cada vez aumentando ese valor.



Ambos se utilizan en la comunicación de los puertos. Pero aunque podamos clasificarlos así, cada

uno puede variar en longitud de datos, velodidad de transferencia, etc.

“Woow, pará cerebrito. ¿Puertos?”

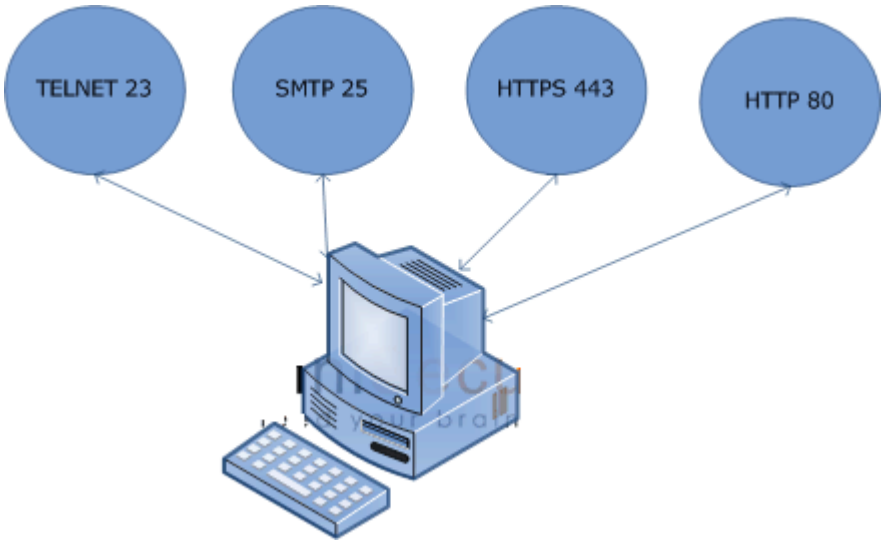
Aaah claro, jamás te conté sobre los puertos. Bueno, imaginen a una computadora conectada a internet. **Los programas utilizan, para comunicarse con otro dispositivo, un puerto. La pc tiene 65535 puertos lógicos**, y por lo tanto, 65535 **posibilidades** de **comunicación**.

Supongamos que tenemos un navegador abierto. Bueno, éste va a tomar un puerto para poder comunicarse con lo que haya afuera. Claro que este puerto no debe estar ya ocupado por otro servicio. En ese caso, va a intentar conectarse con otra computadora y algún puerto que se le asigne para destino, le propondrá un protocolo y por último, si la conexión se acepta, la mantendrá y se comunicará.

Generalmente, por **estandar** hay muchos puertos que estan asignados para ciertos servicios que se inician. Por ejemplo: 80, http; 21, FTP; 25, telnet; etc.

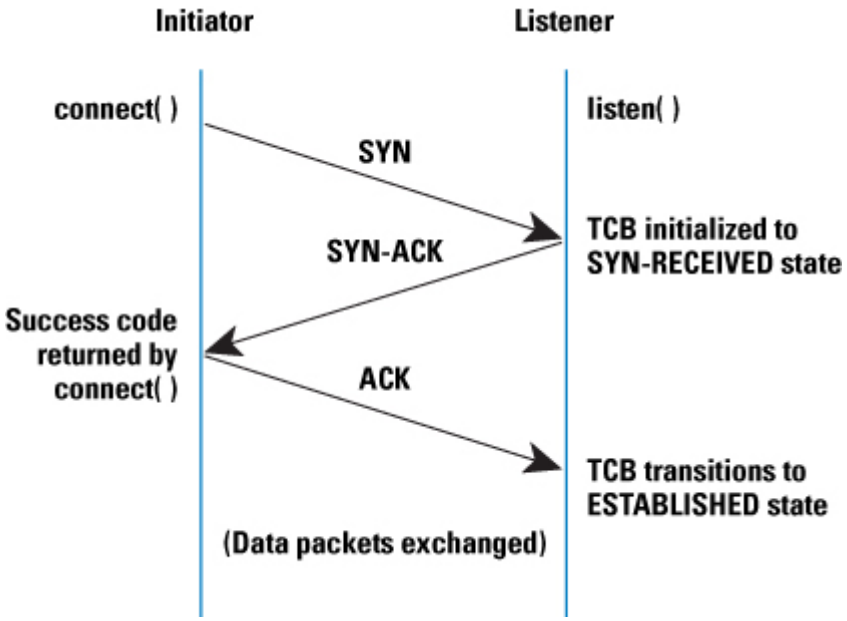
No es necesario que se los aprendan YA, sino que con la práctica verán las cosas que pueden hacer y estarán familiarizados con esto.

Para denotarlos en un destino o salida de una PC se puede hacer IP:puerto, "192.168.1.1:80" sería de la PC con la IP 192.168.1.1, y puerto 80.



“Intentando de llegar más a fondo. ¿Cómo es eso de que propone un tipo de protocolo?”

Claro, para las conexiones TCP, se usa lo que se conoce como el **three handshake** o los tres apretones de manos. Digamos que se envía un paquete de información (que se llama **SYN**) diciendo “hola estoy acá y quiero una conexión con vos”. A esto, si el servicio tiene ganas de conectarse, dice “bueno te acepto, hagámoslo con estos parámetros” (es decir que le envía un paquete de información con un **SYN+ACK**) y luego el mismo que quería la conexión le envía un **ACK** para confirmar ésto. La conexión queda hasta que, generalmente, se envíe un paquete con un “**FIN**” que finaliza la sesión. Si la conexión no es aceptada por el segundo nodo, se le envía un **RST** que rechaza la conexión. Quizás ahora no lo entiendan demasiado, pero luego verán que es pan comido.



“¿Y para el UDP?”

Bueno, el UDP **no realiza esta comprobación** ni nada, simplemente manda paquetes de información sin parar como loco xD. Es por esto que el UDP **no es demasiado fiable.** Aunquemuchas veces para esto, se establece primero una conexión TCP que haga de control y luego un puerto UDP que hace ese tipo de cosas.

Bueno aunque esto fue demasiada información en poco tiempo, ya nos familiarizaremos con todo y lo veremos a fondo. Pero intenten de retener la información ya que lo vamos a usar en los tutoriales que siguen. Y sobre todo, para el examen!

Cualquier cosa pueden mandarme mail a: r0add@hotmail.com

Para donaciones, pueden hacerlo en bitcoin en la dirección siguiente:

1HqpPJbbWJ9H2hAZTmpXnVuoLKkP7RFSvw

Roadd.
