

2. Modelos de referencia. Arquitectura OSI, TCP/IP. Protocolos HTTP y DNS

Ahora vamos a revisar conceptos de arquitectura TCP/IP. Para explicar el concepto arquitectura TCP/IP, vamos a utilizar esta la analogía del proceso que ocurre en un transporte aéreo.

Arquitectura TCP/IP



Vamos a asumir que una persona llega a un aeropuerto, un aeropuerto de origen, en este aeropuerto origen realiza la compra de un pasaje en la aerolínea por donde quiere viajar, ya está facultado para hacer el registro de su equipaje. Registra su equipaje, su equipaje de bodega después van a pasar algunos controles de seguridad, pero finalmente hará el proceso de embarque al avión. El avión se llenará con el resto de los pasajeros y finalmente podrá hacer el despegue hacia la ruta de destino en el proceso de conducción del avión desde el aeropuerto origen al aeropuerto de destino.

Arquitectura TCP/IP



Aquí pueden ocurrir muchas situaciones. Básicamente la existencia del controlador aéreo permite que la ruta se vaya trazando de manera progresiva, por ejemplo: puede haber una ruta ya previamente definida entre la ciudad de origen y la ciudad destino, pero también los controladores de tráfico, los controladores aéreos, van validando la ruta o van redefiniéndola a medida que el avión va pasando por el trayecto hasta llegar al aeropuerto de destino. Cuando llegue aeropuerto de destino, tenemos un proceso pues también muy interesante y es que la aeronave llega al aeropuerto de destino y lo primero que tiene que hacer es por supuesto aterrizar, después de aterrizar ya se puede hacer el proceso siguiente que es el proceso de desembarcar después de desembarcar los pasajeros probablemente pues ya van a hacer el reclamo de su equipaje y van a recoger su equipaje y finalmente una persona cuando ya termina por su servicio puede hacer la calificación del servicio. Si nos damos cuenta las actividades que se hacían en el aeropuerto origen tiene un contraparte en el aeropuerto de destino, el proceso de

despegue tiene un contraparte que es el proceso de aterrizaje el de embarcar es el de desembarcar, el de registrar equipaje y acá es reclamar equipaje, el de comprar el pasaje pues con el de calificar el servicio.

Arquitectura TCP/IP



Entonces en este caso es una arquitectura compuesta por capas donde cada actividad que se hizo en el aeropuerto origen tiene un contraparte en el aeropuerto de destino y cada una de ellas es necesaria para poder facilitar la siguiente o para poder permitir la siguiente. Yo no podría registrar el equipaje sin antes haber comprado un pasaje, no podría embarcar sin antes haber registrado el equipaje y por supuesto haber comprado el pasaje. Entonces cada actividad está vinculada al inmediatamente anterior y en el aeropuerto de destino también necesito primero aterrizar para poder desembarcar, desembarcar para haber retirar el equipaje, y después reclamar equipaje para poder tener posibilidad de calificar el servicio de manera completa. Perfecto ahora vamos a revisar con esta analogía cómo está compuesto internet. Resulta que internet está compuesto de una manera muy similar y también está compuesto por capas. En la arquitectura TCP IP que es la que nos interesa. Vamos a explicarla aquí, la arquitectura o **modelos TCP IP está compuesto por capas y en este caso son cinco capas**: la capa **aplicación** (1) **transporte** (2), **red** (3), **enlace** (4) y **física** (5). Estas cinco capas están en todos los equipos tipo host, tipo servidor.

Modelo TCP/IP

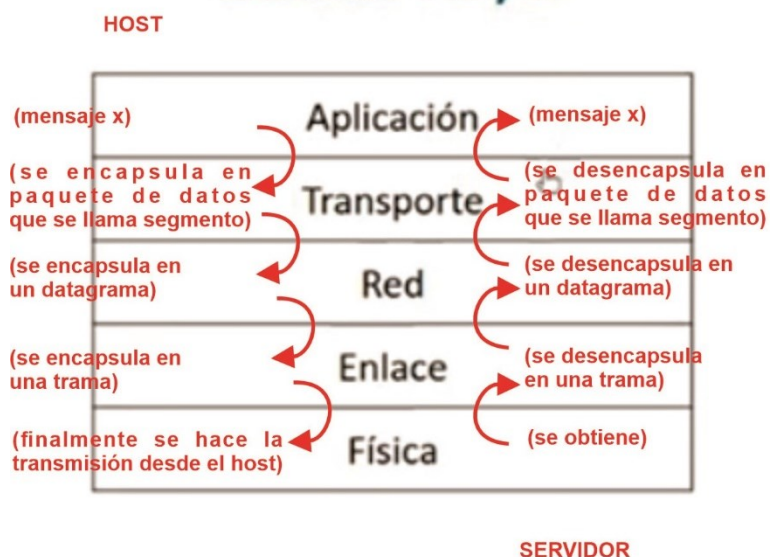
Aplicación	(1)
Transporte	(2)
Red	(3)
Enlace	(4)
Física	(5)

Modelo OSI

Aplicación
Presentación
Sesión
Transporte
Red
Enlace
Física

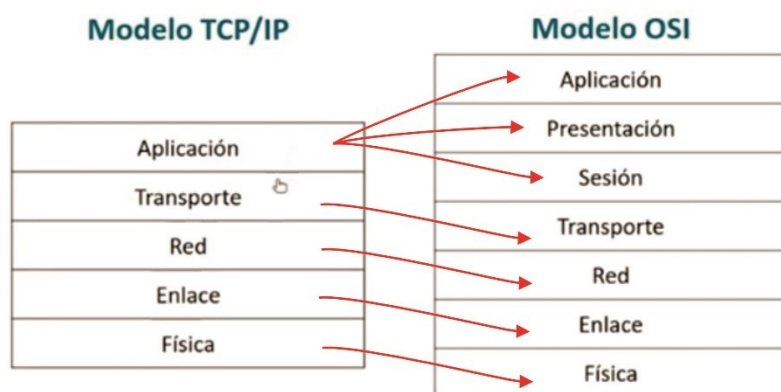
Los hosts, los equipos los laptops, por ejemplo, los servidores también implementan estas cinco capas y cuando existe una comunicación entre un host y el servidor pues existe también un proceso de comunicación justamente que se hace en un lado, donde yo tengo aquí un mensaje por que se encapsula realmente, el nombre es encapsulación y se encapsula en una unidad de datos, en un paquete que sé que se llama segmento aquí en la capa de transporte y luego ese segmento se encapsula en un datagrama en la capa de red, luego el datagrama se encapsula en una trama en la capa de enlace y finalmente se hace la transmisión desde el host. En el servidor que

Modelo TCP/IP



ocurre el proceso es exactamente el contrario, lo que llega a la capa física cierto se obtiene para obtener una trama en la capa de enlace, eso se desencapsula para obtener un datagrama en la capa de red. Luego, la capa de red se lo pasa a la capa de transporte, la capa de transporte desencapsula para obtener un segmento y finalmente la capa de transporte le pasa a la capa aplicación el segmento y la capa de aplicación lo que hace es a capturar también para obtener el mensaje. De esa manera el mensaje de acá, el mensaje enviado el mensaje se envía a través de todas estas capas y también se encapsula a través de estas capas y se desencapsula en el destino a través de esas mismas capas para lograr obtener el

mensaje de destino, el mensaje pues es recuperar el mensaje enviado en el destino. Esa es la forma como funciona el modelo TCP IP, son cinco capas, las cinco capas pues implementan en el host, se implementan en el server, muy similar a como lo veíamos en la analogía donde cada actividad de encapsulación en el origen pues tiene un proceso de ese desencapsulación en el destino.



Hay un modelo adicional que es el modelo OSI, es un modelo teórico que es muy similar al modelo de TCP/IP, sin embargo, tiene unas particularidades, las particularidades básicamente se dan porque la capa de aplicación del modelo TCP/IP tiene una equivalencia a tres capas en el modelo OSI. En el modelo OSI están estas tres capas, pero a nivel funcional hacen pues exactamente lo mismo de lo que haría la capa aplicación del modelo

TCP/IP. El resto de las capas son muy similares en términos de implementación y de funcionalidad, la capa de transporte tiene su par, la capa de red también tiene su par, la capa de enlace tiene su respectivo par en ambos modelos y claramente la capa física también su respectivo par en ambos modelos. **El modelo OSI es un modelo un modelo teórico**, es importante conocerlo porque es la forma como se conciben los sistemas. Pero en la práctica el modelo TCP/ IP es el modelo que realmente se utiliza, es el modelo que está implementado y es el que tenemos pues actualmente en nuestra arquitectura de internet hoy en día.

Vamos a realizar cada una de las capas del modelo de TCP IP para entender qué hace cada una de esas capas, ¿cuál es esa funcionalidad? ¿cuál es ese proceso de encapsulado que hemos mencionado previamente que se hace en cada una de las capas?

Modelo TCP/IP



- **Capa de aplicación:**
 - Incluye protocolos como HTTP que permite la navegación por internet.
 - Los aplicativos hacen uso de estos protocolos para intercambiar información con servidores u otros aplicativos pares.
 - Los paquetes de esta capa se llaman: *mensaje*.
- **Capa de transporte:**
 - Incluye los protocolos de transporte más comunes: TCP y UDP.
 - Las diferencias radican en la garantía de entrega, el control del flujo, la segmentación, el mecanismo de control de congestión.
 - Los paquetes de esta capa se llaman: *segmento*.
- **Capa de red:**
 - Los paquetes de esta capa se llaman: *datagramas*.
 - Incluye el protocolo IP y múltiples protocolos de enrutamiento.
 - Los protocolos de enrutamiento definen la forma como los datagramas llegan a un equipo de destino.

La primera capa, la capa más de más arriba, es la capa de aplicación, en esta capa aplicación hay muchos protocolos principalmente quizás el que más utilizamos hoy en día es el protocolo HTTP que es el que utilizamos por ejemplo para poder navegar el internet, para poder conseguir una descarga, para poder acceder por ejemplo a un portal web.

Las aplicaciones por ejemplo Google Chrome o Mozilla hacen uso de los protocolos que están aquí justamente en la capa de aplicación. Aquí está HTTP y una aplicación como Google Chrome pues hace uso de sus protocolos de la capa aplicación para poder establecer una comunicación de origen a destino, en este caso por ejemplo de una laptop a un servidor de destino de Facebook o de YouTube. En este caso vamos a, como lo hemos mencionado previamente, las unidades de datos en la capa de aplicación se llama mensajes. Luego viene la capa de transporte, la capa de transporte como su nombre lo indica tiene una funcionalidad muy importante y es transportar ese mensaje, ese mensaje que viene de la capa de arriba, de la capa de aplicación.

Para eso tenemos dos protocolos en este caso hay muchos más, pero los más importantes son: protocolo TCP y el protocolo UDP. Cada uno de ellos tiene unas funcionalidades especiales y unas diferencias, de manera general, lo vamos a ver más adelante, pero de manera general, el protocolo TCP permite que la comunicación sea fiable porque garantiza que el destinatario reciba el mensaje que está enviando el emisor el protocolo UDP no, el protocolo UDP simplemente envía, pero no garantiza la recepción del mensaje. Las diferencias entre ambos, como lo hemos mencionado, pues están en la garantía de entrega del mensaje y el control de la congestión, segmentación mecanismos de controles también son diferentes en ambos casos y en la capa de transporte pues ya le vamos a llamar, en este caso, la unidad la vamos a llamar segmento. Vamos ahora con la capa de red, la capa red es la tercera y en este caso primero que todo, a las unidades de datos en la capa de red le vamos a llamar datagramas y en la capa de red también existen muchos protocolos, pero principalmente el más importantes es el protocolo IP, el protocolo IP es el más importante. Es el protocolo que permite la comunicación de un origen a un destino, en este caso los mensajes se van a encapsular en segmentos, los segmentos se van a encapsular en datagramas, datagramas IP en este caso porque pues asumimos que estamos utilizando el protocolo IP. En la capa de red también hay algo muy importante y es que se implementan diferentes protocolos de enrutamiento, los protocolos de enrutamiento permiten justamente que un equipo envíe datos hacia un equipo, un equipo a envía datos hacia un equipo y hay muchos elementos intermedios, muchos elementos intermedios y también posibles rutas que permiten que el mensaje llegue de A a B, entonces esos elementos intermedios se rigen por unos protocolos de enrutamiento que garantiza que el mensaje pues tomen la ruta más adecuada, que se vaya por la ruta más adecuada, por ejemplo, aquella que le permita garantizar un ancho de banda o un throughput óptimo para el tipo de aplicación que se está usando de un origen a un destino.

Modelo TCP/IP



• Capa de enlace:

- Se encarga del movimiento de los paquetes de un nodo al nodo adyacente.
- Incluye protocolos como Ethernet, WiFi y PPP.
- Los paquetes de esta capa se llaman: *tramas*.

• Capa física:

- Se encarga del movimiento de los bits que conforman la trama de un nodo al nodo adyacente.
- Opera en función del medio de transmisión: cobre, fibra óptica, aire.

Pasemos a las últimas dos capas, en las últimas dos capas tenemos la capa de enlace y la capa física. En la capa de enlace tenemos una utilidad también muy importante y es la capacidad de comunicar equipos contiguos por ejemplo si yo tengo aquí un elemento intermedio que es contigo a otro elemento intermedio ósea son adyacentes, están seguidos, la comunicación entre esos dos elementos se hace principalmente utilizando la capa de enlace, utilizando protocolos de transporte entre ellos que se llama el protocolo Ethernet o bueno, en el caso por ejemplo de redes inalámbricas podría ser protocolo wifi también. En el caso de la capa de enlace, ya le vamos a llamar a las unidades de datos tramas. Finalmente tenemos la capa física en la que tenemos ya la conversión de bits a señales que pueden ser señales eléctricas, ópticas, o electromagnéticas.

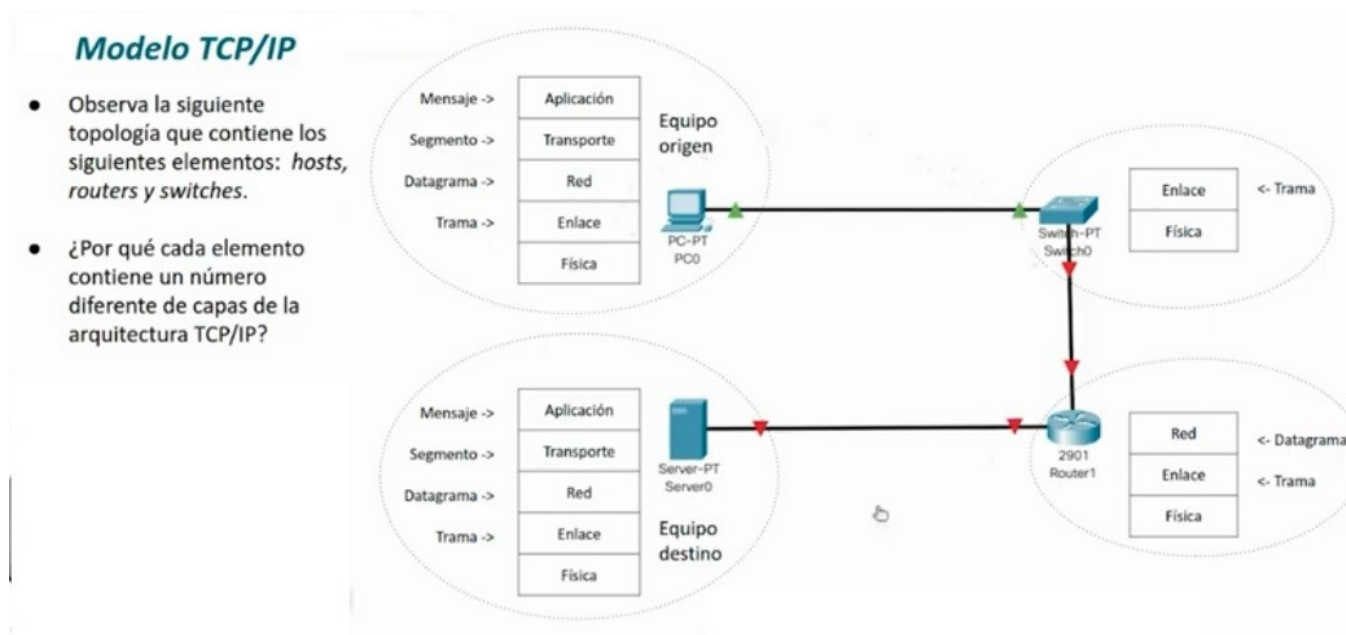
Modelo TCP/IP



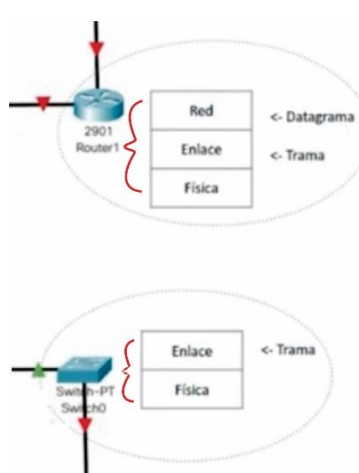
Entonces resumiendo tenemos cinco capas cada una de las capas tiene protocolos y también tiene nombre diferente para las unidades de datos de cada una de ellas. En el caso de la capa aplicación la unidad de datos se llama mensaje, en el caso de la capa de transporte se llama segmento, en el caso de la capa de red se llama datagrama y en el caso de la capa de enlace se llama trama. La capa física no tiene como tal una unidad de datos porque aquí ya lo que hace la capa física es convertir las tramas a bits prácticamente y enviarlas por el medio de transición. Este mismo modelo con cinco capas y con estos nombres de unidades de datos para cada una de ellas se implementa también como hemos dicho en el destinatario, en el equipo hacia dónde van dirigido los

mensajes.

Muy bien, vamos a resumir un poco los conceptos vistos hasta el momento en la figura siguiente: tenemos aquí una topología compuesta por un PC, un servidor, dos elementos intermedios: un Switch y un router.



Es muy importante que reconozcamos que tanto el PC0 como el server tienen las mismas cinco capas aplicación, transporte, red, enlace y física, y por supuesto cada una de sus capas con sus respectivos nombres de unidades de datos.



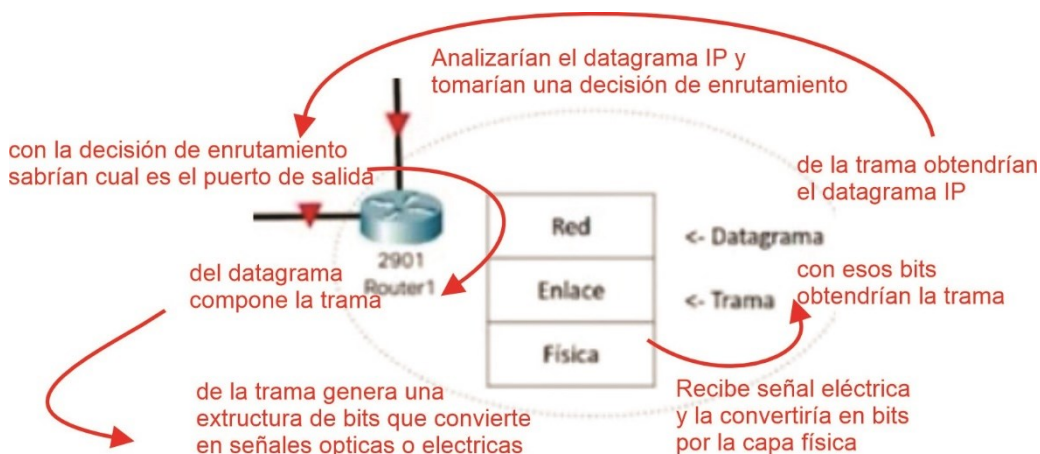
Sin embargo, notemos que los elementos intermedios no necesariamente tienen esas mismas cinco capas, aquí tenemos los elementos intermedios: en este caso por ejemplo un switch y un router. En el caso de switch solamente implementa dos capas: la capa física y la capa de enlace. En el caso del router implementa solamente tres capas: la capa física, la capa de enlace y la capa de red ¿por qué? porque hacer elementos intermedios solamente dependiendo de la utilidad de cada uno de ellos y de dónde se los ubique solamente les hace falta conocer la información por ejemplo este caso, en el caso del switch, la información a nivel de enlace para poder hacer el forward, no necesita conocer de ahí hacia arriba. Entonces no implementa las capas de más arriba de enlace. Solamente con la capa física y la capa de enlace puede hacer el proceso de forwarding, el proceso de reenvío, que se necesita pues como elemento intermedio.

Entonces el caso por ejemplo de este switch recibiría unos bits por la física, a partir de esos bits obtendría la trama, la compondría la trama, analizaría la trama. Y a partir de la trama sabría hacia dónde se tiene que reenviar la trama y de nuevo lo convertiría nuestros bits en señales ópticas o señales eléctricas y los enviaría hacia el destinatario. Solamente con la información de la trama ya puede hacer el reenvío de, en este caso pues del mensaje que está recibiendo.



Ahora en el caso de los routers: los routers en este caso si necesita información de la capa de red entonces recibirían por

ejemplo un mensaje, una señal eléctrica la convertiría en bits obtendrían la trama, a partir de la trama obtendrían el datagrama, el datagrama IP, analizarían el datagrama IP y tomarían una decisión de enrutamiento



porque es un router. Y esa decisión de enrutamiento le diría pues cuál es la interfaz de salida hacia dónde tiene que enviar ese datagrama ya sabiendo hacia donde tiene que enviarlo, de nuevo del datagrama compone la trama, de la trama genera una estructura de bits que son convertidos a señales eléctricas y finalmente pues haría el reenvío del datagrama.

Entonces esos elementos intermedios no necesitan tener las cinco capas incorporadas. En el caso por ejemplo los router, necesita tener hasta la capa 3 porque la información de enrutamiento la obtienen justamente a partir de la información que viene en el datagrama IP. En el caso de los switches la información de forwarding la obtienen a partir de la de la información que está contenida dentro de la trama que está justamente a nivel de la capa 2.

- ¿Por qué cada elemento contiene un número diferente de capas de la arquitectura TCP/IP?

Bueno la pregunta es **¿por qué cada elemento contiene un número diferente de capas de la arquitectura TCP IP?**

Porque cada elemento tiene un propósito diferente y por lo tanto pues cada elemento pues implementa las capas justas y necesarias para poder concluir su función, en el caso de los elementos intermedios únicamente funciones de reenvío, de forwarding en el caso de switch, de enrutamiento en el caso el router en el caso de equipo origen, equipo destino, ahí sí por supuesto se necesitan las cinco capas completas porque necesitan que el mensaje enviado desde el servidor pueda encapsularse a través de todas esas capas en el equipo de destino, desencapsularse en el equipo destino y finalmente obtener el mensaje el equipo en el equipo pues hacia dónde está llegando la comunicación. Si lo hacemos en el sentido contrario pues también, el mensaje se encapsularía en el equipo de origen y se desencapsularía en el equipo de destino, obteniendo también un mensaje pues acá igual al enviado desde el equipo de origen.