

Redes Computadores

Modelo de capas

Es un modelo de referencia que proporciona una referencia común para mantener la consistencia dentro de todos los tipos de protocolos y servicios de red (**no está pensado para ser una especificación de implementación**)

El modelo de referencia OSI de ISO está formado por 7 capas o niveles con la finalidad de dividir el proceso de comunicación en un conjunto de funciones independientes entre si

Diseñadas con arreglo a los siguientes principios:

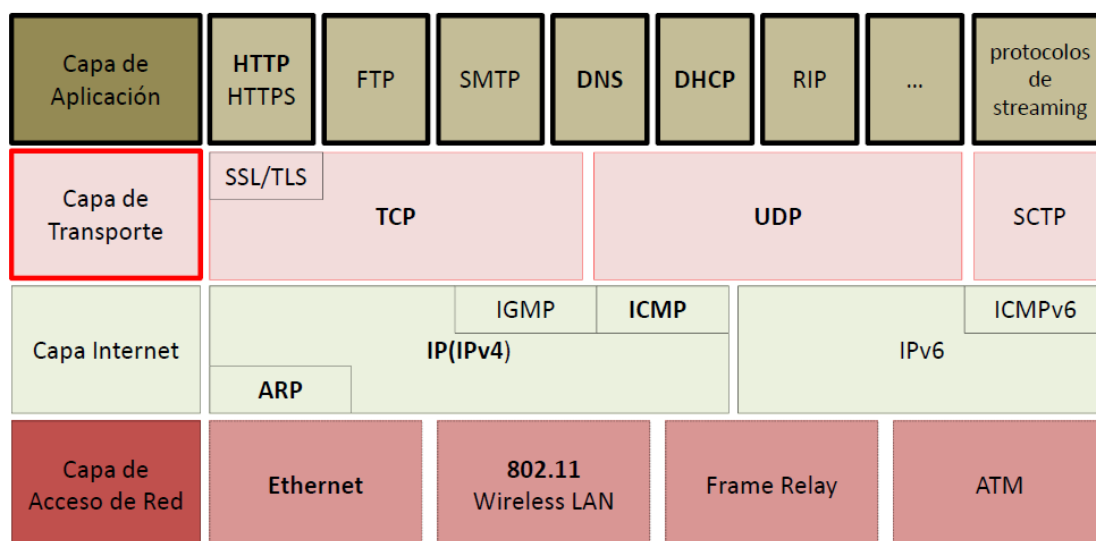
- Una capa se creará en situaciones en las que se requiera un nivel diferente de abstracción.
- Cada capa deberá realizar una función bien definida.
- La función que realiza cada capa deberá seleccionarse tomando en cuenta la minimización del flujo de información.
- El número de capas será suficientemente grande como para que funciones diferentes no estén en la misma capa y suficientemente pequeño para que la arquitectura no sea difícil de manejar.

(El modelo OSI no es una arquitectura de red puesto que no especifica los protocolos que deben usarse en cada capa)

OSI

Internet

Aplicación	Proporciona a los programas de aplicación un medio de acceso a los recursos a la red.	Aplicación
Presentación	Formato de los datos. Traduce, comprime y encripta los datos	
Sesión	Mantiene el control del enlace entre computadoras que transmiten datos	
Transporte	Asegura la entrega de los datos entre procesos que han establecido una sesión y que se ejecutan en diferentes nodos	Transporte
Red	Entrega los “paquetes” y hace enrutamiento	Internet
Enlace	Inicia, mantiene y libera el enlace, transmisión confiable sin errores	Acceso a la red
Física	Transmite datos binarios sobre un medio proporcionando las especificaciones eléctricas	



El modelo TCP/IP es un estándar abierto donde se especifican las definiciones del estándar de los diferentes protocolos

Un proceso de comunicación completo incluye estos pasos:

- Creación de datos en la capa de aplicación del host origen
- Segmentación y encapsulación de datos a medida que pasan por el pila de protocolos host origen
- Generación de datos para transporte por los medios en la capa de acceso a la red del sistema
- Transporte de los datos a través de red (**compuesta por medios y por cualquier dispositivo intermediario**)
- Recepción de los datos en la capa de acceso a la red del host de destino
- Des-encapsulación y re-ensamblaje de los datos a medida que pasan por la pila en el dispositivo de destino
- Transmisión de estos datos a la aplicación de destino en la capa de aplicación del dispositivo final de destino

Capa de aplicación

La capa de aplicación no tiene en cuenta:

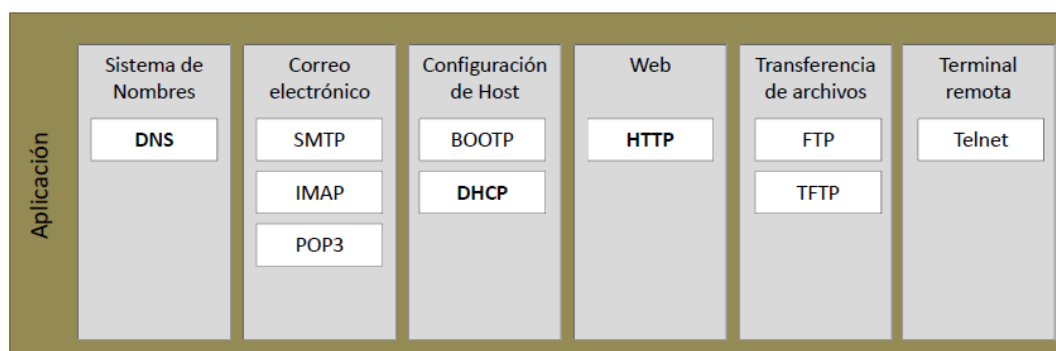
- el host destino
- el medio de transmisión
- el tamaño de los datos,
- la congestión en el enlace
- el tamaño de la red

Los servicios del nivel de transporte y de aplicación utilizan normalmente un modelo Cliente-Servidor

- **Cliente:** Realiza una petición o solicita el Inicio de una conexión
- **Servidor:** Espera recibir peticiones o solicitudes de conexión

Los protocolos de la capa de aplicación especifican el control y formato de los mensajes con la finalidad de Establecer las reglas para el intercambio de datos entre las aplicaciones del origen y del destino

- Especifican los tipos de mensajes que se envían entre origen y destino
- Definen la secuencia de mensajes
- Especifican cómo se estructuran los datos dentro de los mensajes



DNS – Servicio de Nombres de Dominio

Es un sistema de nomenclatura jerárquico descentralizado para dispositivos conectados a redes IP como Internet

Este sistema asocia información variada con nombre de dominio asignado a cada uno de los participantes.

Su función es traducir nombres inteligibles para las personas en identificadores binarios asociados con los equipos conectados a la red con el propósito de poder localizar y direccionar estos equipos mundialmente.

Ventajas:

- Cambio de la dirección numérica es transparente para el usuario, el nombre de dominio seguirá siendo el mismo. La nueva dirección simplemente se enlazada con el nombre de dominio existente y la conectividad se mantendrá.

Inconvenientes:

- Es necesario asegurarse de que no existan dos computadoras con el mismo nombre.
- Es necesario proporcionar una forma de convertir los nombres a direcciones numéricas

Los servidores DNS almacenan diferentes tipos de registros de recursos que utilizan estos para resolver los nombres.

- **Nombre de dominio:** nombre del host o del dominio DNS al que pertenece este recurso.
- **TTL:** tiempo que un servidor debe guardar en cache esta entrada de registro antes de descartarla (opcional).
- **Tipo:** identifica el tipo de registro.
- **Clase:** define la familia de protocolos en uso.
- **RData:** los datos del registro de recursos.

HTTP – Protocolo de Transferencia de Hipertexto

Capa de Presentación

Se encarga de mostrar los datos de forma que el dispositivo receptor pueda comprender.

- Trabaja más el contenido de la comunicación que cómo se establece la misma.
- Se encarga de la representación de la información, de manera que los datos lleguen al destino de manera reconocible aunque los equipos puedan tener diferentes representaciones internas.
- Desempeña tres funciones principales:
 - o Codificación o formateo de datos
 - o Encriptación o cifrado de datos
 - o Compresión de datos

Capa de Sesión

Su función es proporcionar los mecanismos para controlar el diálogo entre las aplicaciones de los sistemas finales

Proporciona los siguientes servicios

- **Control del Diálogo:** iniciar los diálogos y mantenerlos activos
- **Recuperación:**
 - o Reiniciar sesiones que se interrumpieron o desactivaron durante un periodo de tiempo prolongado
 - o Proporcionar un procedimiento de puntos de comprobación, de forma que si ocurre un fallo se puede retransmitir los datos desde el último punto de comprobación y no desde el principio.

Capa de transporte

Funciones que realiza

En la capa de transporte se aceptan los datos de la capa de Aplicación y se preparan para el enviarlos a la capa de red.

Se encarga del transporte de los datos extremo a extremo

No saben que existen varias aplicaciones. Solo entregar la información al Dispositivo adecuado

Las redes tienen una limitación en la cantidad de datos que se pueden incluir en una simple PDU

Un host origen puede tener múltiples aplicaciones que se comunican a través de la red con una o más aplicaciones en el hosts destino.

Los servicios que ofrece pueden ser de dos tipos

- **Orientados a la conexión**
- **No orientados a la conexión**

Segmentación y encapsulación

- La capa de transporte divide los datos de aplicación en segmentos de un tamaño adecuado a los límites del medio de forma que se permita multiplexar datos de diferentes aplicaciones.
 - o En el host destino se deben direccionar los paquetes a la aplicación adecuada más aplicaciones en el hosts destino.
 - o Es responsabilidad de la capa de transporte mantener la comunicación múltiple entre estas aplicaciones. Para lograr esto, la capa de transporte asigna un identificador exclusivo (**número de puerto**) para la aplicación en ese host.
- En los protocolos de la capa de transporte se describen los servicios que segmentan estos datos de la capa de aplicación.

Re-ensamblaje

- Los segmentos de datos individuales también deben reconstruirse para generar los datos útiles para la capa de aplicación destino.
- La capa de transporte re-ensambla los datos antes de enviarlos a la aplicación o servicio de destino.
- Los protocolos en la capa de transporte describen cómo se utiliza la información del encabezado para reensamblar los segmentos y pasarlos a la capa de aplicación.

Seguimiento, acuse de recibo y retransmisión

- Es posible que algunos segmentos se pierdan o se corrompan a medida que se transmite a través de la red.
- Algunas aplicaciones necesitan que todos los datos enviados lleguen al destino en su condición original para que sean útiles mientras que otras son más tolerantes.
- Un protocolo de la capa de transporte puede implementar algún método para asegurar el envío confiable de datos haciendo un seguimiento de todos los paquetes enviados y reenviando cualquier dato del que no se haya recibido acuse de recibo
- En la capa de transporte del destino también debe rastrear los paquetes de datos a medida que se reciben y reconocer la recepción de los mismos.

Entrega

- Los datos pueden llegar en el orden equivocado, debido a que las redes pueden proporcionar múltiples rutas que pueden tener diferentes tiempos de transmisión.
- Los protocolos de la capa de transporte numeran y secuencian los segmentos de ese modo pueden re-ensamblar los mismos en el orden adecuado.

Control de flujo

- Los hosts de la red cuentan con recursos limitados, como memoria o ancho de banda, el control de flujo puede evitar la pérdida de segmentos en la red y evitar la necesidad de la retransmisión.
- Si la capa de transporte advierte que sus recursos están sobrecargados, algunos protocolos pueden solicitar que la aplicación que envía reduzca la velocidad del flujo de datos.

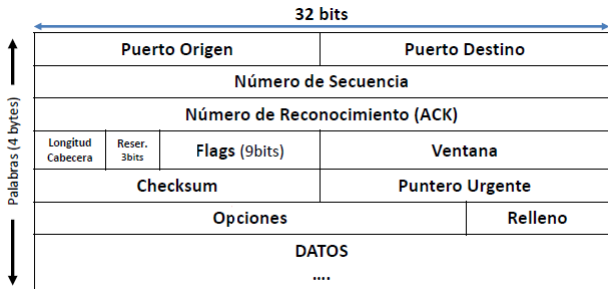
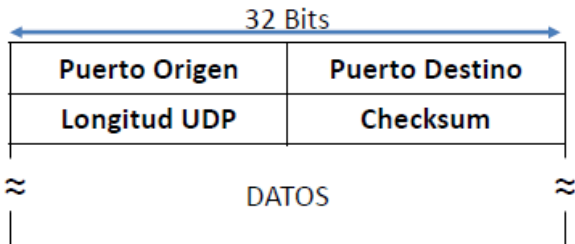
Protocolos TCP / IP

Los dos protocolos más importantes de la capa de transporte son:

- **Protocolos Protocolo de Control de Transmisión TCP**
- **Protocolo de Datagramas de Usuario UDP**

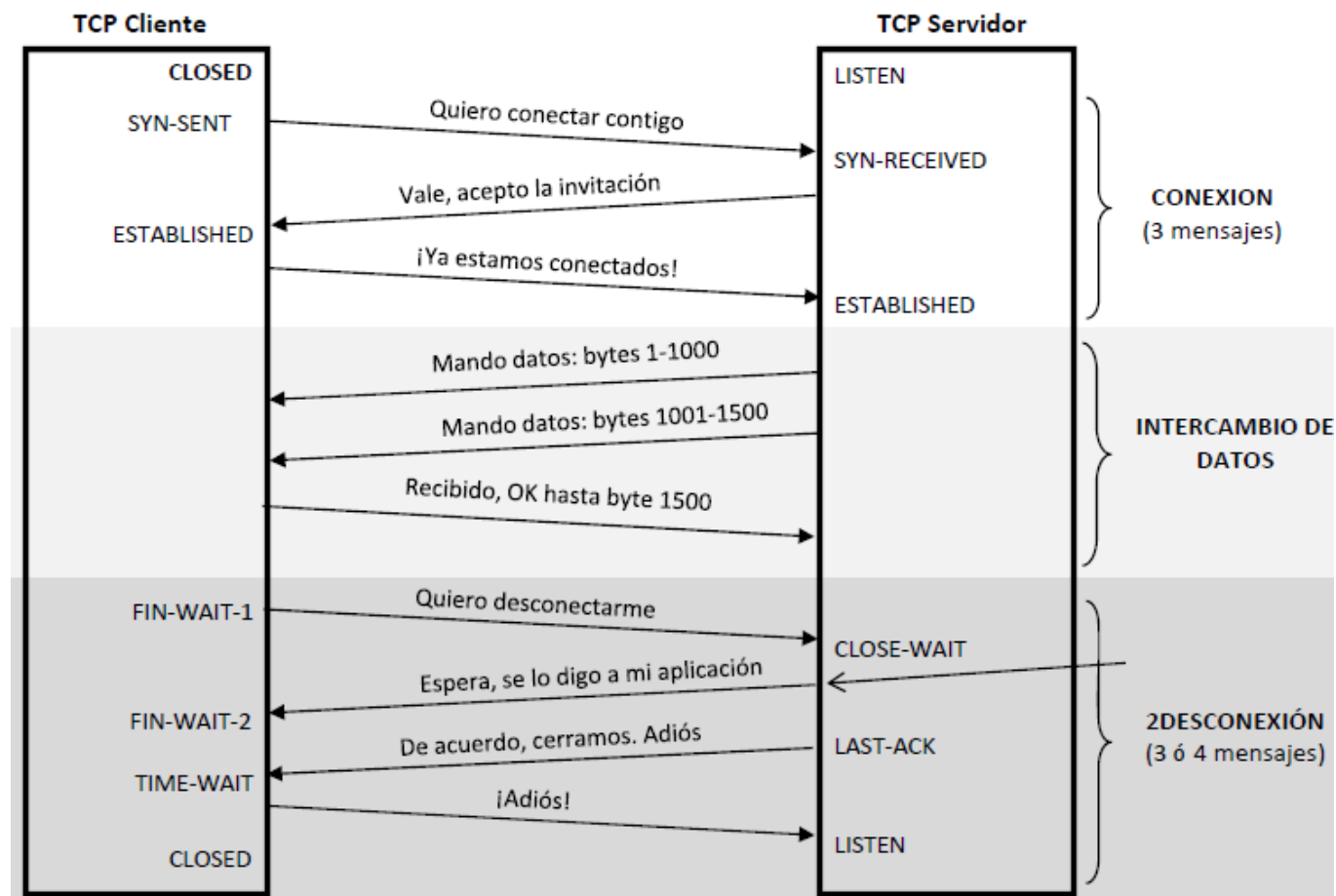
Tanto TCP como UDP mantienen un seguimiento de las diversas aplicaciones que se comunican.

La diferencia entre ellos está en las funciones específicas que cada uno implementa.

	TCP	UDP
Características	<ul style="list-style-type: none"> - Protocolo Orientado a la Conexión - Implementa opciones avanzadas <ul style="list-style-type: none"> o Multiplexación de aplicaciones o Segmentación o Control de errores o Control de flujo o Control de congestión o Retransmisión de datos perdidos o Conexión/desconexión - utiliza recursos adicionales para: <ul style="list-style-type: none"> o Los acuses de recibo y las retransmisiones. o La re-organización de la información 	<ul style="list-style-type: none"> - No orientado a la conexión. - Solo implementa las opciones básicas: <ul style="list-style-type: none"> o Multiplexación de aplicaciones o Segmentación o Opcionalmente una comprobación de errores - Utiliza pocos recursos de red por lo que genera menos carga y produce una transferencia de datos mas rápida
Segmentación	El encabezado de los segmentos contiene un número de secuencia que permite que las funciones de la capa de transporte del host de destino re-ensamblen los segmentos en el mismo orden en el cual se transmitieron.	No se preocupa del orden en que se transmite la información, por lo que la aplicación de destino podría recibir la información en un orden distinto al que fue transmitida, ya que los mensajes pueden tomar rutas distintas a través de la red
Diseño de los Paquetes	<p>La información se divide en segmentos</p>  <ul style="list-style-type: none"> - Puertos de origen y destino son un número de 16 bits que identifica la aplicación origen y destino respectivamente - Checksum es un campo de control de errores. - Numero de secuencia SEQ permite establecer el orden de los mensajes para su re-ensamblaje - Longitud UDP indica el tamaño en bytes de todo el datagrama, incluyendo la cabecera y los datos. - Numero de reconocimiento AKT indica el número del primer byte que se espera recibir en el siguiente segmento. 	<p>La información se divide en datagramas</p> 

El receptor TCP coloca los datos del segmento en un búfer de recepción. Los segmentos se colocan en el orden de número de secuencia adecuado y se pasa a la capa de aplicación cuando se re-ensamblan.

Flujo de la comunicación TCP



Saludo a tres vías: El mecanismo de conexión utilizado por TCP se basa en el intercambio de tres mensajes

- El cliente envía al servidor una invitación a conectar
- Cuando recibe la invitación el servidor devuelve una respuesta al cliente aceptando la invitación.
- Al recibir la respuesta, el cliente considera establecida la conexión y envía un tercer mensaje de acuse de recibo.

Intercambio de datos:

El TCP reconoce datos por medio de la técnica de Piggybacking de modo que en vez de enviar un segmento de confirmación ACK individual dicha confirmación es incluida dentro del próximo paquete que se envía.

El número ACK menos uno indica el último byte reconocido, por lo que se emplea para saber el número de bits que han sido recibidos correctamente

- Todos los segmentos intercambiados en una conexión TCP excepto el primero tienen puesto el flag ACK.
- La presencia del flag ACK NO incrementa el número de secuencia.

El número SEQ en el encabezado de cada paquete.

- Durante la configuración de la sesión, se establece un número de secuencia inicial (ISN).
- Este número de secuencia inicial representa el valor de inicio para los bytes de esta sesión que se transmitirán a la aplicación receptora.
- A medida que se transmiten los datos durante la sesión, el número de secuencia se incrementa en el número de bytes que se han transmitido.

Desconexión:

El mecanismo de desconexión utilizado por TCP se basa en el intercambio de cuatro mensajes

- Cuando uno de los participantes no tiene más datos para enviar envía un segmento con el flag FIN activado
- El otro participante envía un ACK para acusar de recibo del FIN desde el cliente y continúa enviando sus datos.
- Cuando ya no tiene más datos para enviar, envía un FIN al para terminar la sesión.
- El que había solicitado cortar la conexión en un principio responde con un ACK para dar acuse de recibo del flag FIN

Es preciso tener en cuenta que tanto el cliente como el servidor pueden cerrar la conexión

Gestión de segmentos perdidos

TCP cuenta con métodos para gestionar pérdidas de segmentos.

Entre estos está un mecanismo para retransmitir segmentos con datos sin acuse de recibo.

- Un servicio de host de destino que utiliza TCP GENERALMENTE sólo da acuse de recibo de datos para bytes de SECUENCIA CONTINUOS.

– Por ejemplo, si se recibieron los segmentos con números de secuencia de 1500 a 3000 y de 3400 a 3500, el número de acuse de recibo sería 3001. Esto es porque hay segmentos con números de secuencia del 3001 al 3399 que no se han recibido.

- Cuando el TCP en el host de origen no recibe un acuse de recibo después de un determinado período de tiempo, éste regresará al último número de acuse que recibió y volverá a transmitir los datos desde dicho punto.

- En la actualidad, los hosts pueden emplear también una función opcional llamada ACUSES DE RECIBO SELECTIVOS. Si ambos hosts admiten el Acuse de recibo selectivo, es posible que el destino reconozca los bytes de segmentos discontinuos y el host sólo necesitará retransmitir los datos perdidos.

Control de flujo

- El campo VENTANA en el encabezado del TCP especifica la cantidad de datos que se pueden transmitir antes de que se deba recibir un acuse de recibo.

- El tamaño inicial de la ventana se determina durante el arranque de sesión por medio del enlace de tres vías.

- Ejemplo: Si el tamaño inicial de la ventana para una sesión TCP se establece en 3000 bytes. Cuando el emisor transmite 3000 bytes, espera por un acuse de recibo de los mismos antes de transmitir más segmentos para esta sesión. Una vez que el emisor tiene este acuse de recibo ya puede transmitir 3000 bytes adicionales.

Durante el retraso en la recepción del acuse de recibo, el emisor no enviará ningún segmento adicional para esta sesión.

En los períodos en los que la red está congestionada o los recursos del host receptor están ocupados, la demora puede aumentar. A medida que aumenta esta demora, disminuye la tasa de transmisión efectiva de los datos para esta sesión.

La disminución de la velocidad de los datos ayuda a reducir la contención de recursos.

- Otra forma de controlar el flujo de datos es utilizar TAMAÑOS DE VENTANA DINÁMICOS.

- Cuando los recursos de la red son limitados, TCP puede reducir el tamaño de la ventana para lograr que los segmentos recibidos sean reconocidos con mayor frecuencia.

- Esto reduce de forma efectiva la velocidad de transmisión porque el origen espera que se de acuse de recibo de los datos con más frecuencia.

- El host receptor del TCP envía el valor del tamaño de la ventana al TCP emisor al inicio de la sesión.

- Si el destino necesita disminuir la velocidad de comunicación debido a su memoria de búfer limitada, puede enviar un valor más pequeño del tamaño de la ventana al origen como parte del acuse de recibo.

- Después de períodos de transmisión sin pérdidas de datos o recursos limitados, el receptor comenzará a aumentar el tamaño de la ventana. Esto reduce la sobrecarga de la red, ya que se requiere enviar menos acuses de recibo. El tamaño de la ventana continuará aumentando hasta que haya pérdida de datos, lo que producirá una disminución del tamaño de la misma.

- Estas disminuciones y aumentos dinámicos del tamaño de la ventana representan un proceso continuo en TCP que determina el tamaño óptimo de la ventana para cada sesión del TCP. En redes altamente eficientes, los tamaños de la ventana pueden ser muy grandes porque no se pierden datos. En redes donde se tensiona la infraestructura subyacente, el tamaño de la ventana probablemente permanecerá pequeño.

Son técnicas que permite sincronizar el envío de información entre dos dispositivos que la producen y la procesan a distintas velocidades.

Control de congestión.

– Concepto más amplio que el control de flujo Es un conjunto de técnicas flujo. para detectar y corregir los problemas que surgen cuando no todo en una red puede ser cursado, con los requerimientos de retardo, u otros, necesarios desde el punto de vista de la calidad del servicio. Por tanto, es un concepto global, que involucra a toda la red, y no sólo a un remitente y un destinatario de información, como es el caso del control de flujo.

Capa de RED

Define cómo transportar datos entre dispositivos que no están conectados localmente en el mismo dominio de difusión

- La capa de red se encuentra en todos y cada uno de los hosts y Routers de la red.

El Router

- Es un dispositivo que proporciona conectividad a nivel de red.
- Su función es interconectar subredes enviando paquetes de una red a otra
- Tienen configurada la dirección IP de cada uno de los interfaces.
- Cada interface tiene configurada la máscara de red empleada en la red a la que está conectada
- Tienen configurada una Tabla de enrutamiento en la que se identifica la red destino y su máscara para cada entrada

La **puerta de enlace (Gateway)** cuando un host quiere enviar un paquete a un dispositivo que no está en la misma red el Router actúa como puerta de enlace hacia la red destino

- Es necesaria para enviar un paquete fuera de la red local.
- Es una interfaz del Router conectado a la red local.
- Tiene una dirección de red que pertenece a la red de los hosts origen.
- Los hosts son configurados para reconocer esta dirección como puerta de enlace.

La tabla de enrutamiento

- almacena información sobre redes conectadas y remotas.
 - o Las redes conectadas están directamente unidas a una de las interfaces del Router. Estas interfaces son las puertas de enlace para los hosts en las diferentes redes locales.
 - o Las redes remotas son redes que no están conectadas directamente al Router.
- El administrador de red puede configurar las rutas a estas redes manualmente o automáticamente a través de protocolos de enrutamiento

La capa de red realiza cuatro funciones básicas:

- **Direccionamiento**

Proporcionar un mecanismo para que los datos lleguen a los dispositivos finales correctos.

- **Encapsulación**

La capa de red recibe el mensaje de la capa de transporte y agrega un encabezado que contendrá la dirección del host destino y del host de origen.

El paquete se envía a la capa de enlace de datos a fin de prepararse para el transporte a través de los medios.

- **Enrutamiento**

El Router toma una decisión de reenvío para cada uno de los paquetes que le llegan a una de sus interfaces.

Para llevar a cabo esta tarea necesita conocer una ruta hacia esa red.

- o Si no existe una ruta a una red de destino, el paquete no puede reenviarse.
- o En cada Router sólo se indica el Router del siguiente salto, no el Router final.

Utiliza las rutas que tiene previamente definidas en su tabla de enrutamiento para asignar la dirección de red destino del siguiente salto y reenvía el paquete hacia esta dirección.

Los campos de las entradas de las tablas de enrutamiento son:

- o **Destino y máscara de red:** Se utilizan para ver la coincidencia con la dirección destino
- o **Puerta de acceso:** Es la dirección IP que utiliza el host local para reenviar datagramas IP
- o **Interfaz:** Es la dirección IP del interfaz del propio equipo por el que se alcanza la puerta de acceso
- o **Métrica:** indica el costo del uso de una ruta (número de saltos al destino IP).
 - Cualquier destino en la subred local está a un salto de distancia
 - Cada enrutador que se atraviesa en la ruta es un salto adicional.
 - Si existen varias rutas al mismo destino con diferentes métricas, se selecciona la ruta con menor métrica.

Protocolo de enrutamiento:

- El host origen envía, por la red local, el paquete al Router puerta de enlace.
- El Router examina la dirección destino del encabezado paquete (**sin modificarla**) y busca en la tabla de enrutamiento la máscara que coincida con la dirección de destino
 - o Si no coincide ninguna descarta el paquete
 - o Si encuentra una o más rutas válidas hacia el mismo destino escoge aquella con la máscara más larga y reenvía el paquete al Router del siguiente salto que especifica dicha ruta.
 - o Cuando la ruta de destino no está representada por ninguna ruta en la tabla de enrutamiento se utiliza la ruta predeterminada (**en redes IPv4 se usa la dirección 0.0.0.0**)
 - o Si la red destino está conectada directamente al Router, el paquete se reenvía directamente al host destino.
- El enrutamiento se hace paquete por paquete y salto por salto. Cada paquete es tratado de manera independiente en cada Router a lo largo de la ruta.

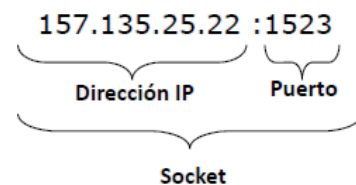
Capa de Enlace

Su función es preparar los paquetes de la capa de red para su transmisión y controlar el acceso a los medios físicos

Otros conceptos:

Número de puerto: Para diferenciar los segmentos TCP o datagramas UDP dirigidos a cada aplicación, ambos protocolos cuentan con campos de encabezado que pueden identificar estas aplicaciones de manera exclusiva mediante una dirección local que indica la aplicación origen y destino del paquete:

- Tiene un tamaño de 16 bits por lo que su valor está comprendido entre 0 y 65535
- Podemos identificar tres rangos:
 - o **Puertos conocidos** [0 – 1023]: Se reservan para servidores de protocolos estándar (el puerto de HTTP es el 80)
 - o **Puertos registrados** [1024 – 49151]: Pueden ser usados por cualquier aplicación. Existe una lista pública en la Web de IANA donde se puede ver qué protocolo usa cada uno de ellos.
 - o **Puertos dinámicos o Privados** [49152– 65535]. Se asignan de forma dinámica a las aplicaciones cuando se inicia una conexión.
- La asignación del número de puerto depende de si se trata de un cliente o un servidor
 - o Los servidores tienen asignados números de puerto estáticos, en los clientes se eligen de forma dinámica un número de puerto para cada conversación.
 - El cliente debe conocer el número de puerto asociado al proceso en el servidor.
 - El puerto del cliente se crea de forma aleatoria.
 - o La capa de transporte destino mantiene un seguimiento del puerto cliente asociado a la aplicación que generó la solicitud y después la utiliza como número de puerto destino en la respuesta.



Socket: La combinación de un número de puerto de la capa de transporte y de una dirección IP de la capa de red identifica de manera única un proceso de red que se ejecuta en un dispositivo host específico. [Un par de sockets, identifica la conversación entre dos hosts.](#)

Los datos que se envía desde un origen hasta un destino se puede dividir en paquetes y entrelazar con los mensajes que viajan desde otros hosts hacia otros destinos. Es muy importante que cada paquete de datos contenga suficiente información de identificación para llegar al destino correcto.