Resumen Tema 2

Capa de red

Autor: @BlackTyson

Índice

1. Introducción

1.1. Funciones y servicios de la capa de red en TCP/IP

- El objetivo es la **interconexión de redes** con independencia de la tecnología.
- **Conmutación**: acción de **cursar tráfico entre los nodos** de la red.
- **Encaminamiento**: Encontrar la mejor ruta
- OSI realiza el control de congestión en esta capa

2. Conmutación

Es la acción de cursar tráfico para establecer un camino que permita transmitir información de extremo a extremo.

Tecnologías

- Circuitos
- Paquetes

2.1. Conmutación de circuitos

Servicio orientado a conexión que exige un establecimiento de conexión previo

Pasos

1. Conexión
2. Transmisión
3. Conexión

Ventajas

- Uso permanente de recursos dedicados
- Comunicaciones en **tiempo real**
- No hay pelea por acceder al medio
- Circuito fijo implica no cambios de encaminamiento
- Simplicidad en la gestión de los nodos intermedios

Desventajas

- **Retraso de establecimiento.**
- Uso ineficiente de recursos.
- Circuito fijo implica no reajuste de la ruta de comunicación.

Esquema de flujo

1. **Establecimiento de conexión:** Se solicita una conexión entre las estaciones origen y destino. Una vez confirmada la conexión, el circuito se reserva para la transmisión. Este paso implica:
   - **Retardo de procesamiento:** Tiempo necesario para procesar la solicitud en cada nodo.
   - **Retardo de propagación:** Tiempo requerido para que las señales viajen entre nodos.
2. **Transmisión de datos:** Con el circuito establecido, los datos se transmiten de manera continua y garantizada a través del camino dedicado.
3. **Cierre de conexión:** Una vez finalizada la transmisión, se envía una solicitud de cierre que libera los recursos del circuito reservado.

2.2. Conmutación de datagramas

Características

- **Transmisión independiente:** Los paquetes se envían de forma **autónoma**, cada uno con la información necesaria para alcanzar su destino.
- **No orientada a conexión:** No se establece un canal fijo entre el origen y el destino, por lo que cada paquete se maneja por separado.
- **Almacenamiento y reenvío:** En cada nodo de la red, los paquetes se almacenan temporalmente mientras esperan ser reenviados al siguiente nodo en la ruta.
- **Direcciones en las cabeceras:** Cada paquete incluye información clave en su cabecera, como las direcciones de origen y destino, para que pueda ser dirigido correctamente.
- **Rutas variables:** Los paquetes pueden seguir diferentes rutas dependiendo de las condiciones de la red, lo que puede resultar en que lleguen al destino de forma desordenada.

Esquema de flujo:

- Los paquetes (**paquete 1, paquete 2, paquete 3**) salen de la estación origen hacia el primer nodo.
- En cada nodo, hay un **tiempo de procesamiento**, durante el cual los paquetes se almacenan temporalmente y luego se reenvían.
- A medida que los paquetes atraviesan la red, pueden llegar a los nodos intermedios y al destino en un orden diferente al que fueron enviados debido a las rutas independientes que siguen.

Conmutación de paquetes con circuitos virtuales

- **Uso en redes ATM:** Este método se utilizaba en redes ATM, aunque actualmente está en desuso para redes troncales.
- **Orientada a conexión:** Se configura una conexión virtual entre el origen y el destino antes de la transmisión. Este proceso consta de tres pasos principales:
  1. **Conexión:** Se establece un circuito virtual entre los nodos involucrados.
  2. **Transmisión:** Los paquetes se envían siguiendo el circuito establecido.
  3. **Desconexión:** Al terminar la transmisión, el circuito virtual se cierra.
- **No hay asignación exclusiva de recursos:** A diferencia de la conmutación de circuitos, los recursos de la red no se reservan exclusivamente para una conexión.

3. Protocolo IP

3.1. Introducción y características (IPv4

- Especificado en RFC 791
- Protocolo de interconexión de redes
- Protocolo **salto a salto**
- Servicio **no** orientado a conexión, **no** fiable
  - No hay handshake entre las entidades
  - No realiza control de errores, flujo o congestión.
- El paquete IP se denomina datagrama (cabecera + datos)
- Protocolo **"best-effort"**, los datagramas se pueden perder, duplicar, retrasar o llegar desordenados.
- **Gestiona la fragmentación**, adaptando el tamaño del datagrama a las diferentes MTUs de las subredes necesarias.

3.2. Direcciones IP

Fundamentos

- Internet adopta **direccionamiento jerárquico**, simplificando las tablas de routing.
- Las direcciones IPv4 tienen dos partes:
  - Identificador de subred
  - Identificador del dispositivo.
- Cada **subred** tiene un **identificador único** en la intranet o en **internet**.
- Cada **dispositivo** tiene un **identificador único** en la **subred**.
- La máscara de red es un patrón de 1s que determina qué **bits pertenecen al identificador** de subred
  - Dirección IP

    ```
    200.27.4.112 = 11001000.00011011.00000100.01110000
    ```

  - Máscara

    ```
    255.255.255.0 = 11111111.11111111.11111111.00000000
    ```

    Permite representarse de manera compacta (ej: /24)

Determinar identificador subred

Para determinar la dirección o identificador de la **subred** a la que pertenece una dirección IP, se utiliza una operación lógica **AND** entre la dirección IP y la máscara de subred. A continuación, se describe el proceso con un ejemplo:

- Dada la dirección IP: **200.27.4.112**
- Máscara de subred: **255.255.255.0**

Paso 1: Representación en binario
Cada octeto de la dirección IP y la máscara de subred se convierte a su equivalente en binario:

$$\text{IP:} \quad 200,27,4,112 = 11001000,00011011,00000100.\underline{01110000}$$

$$\text{Máscara:} \quad 255,255,255,0 = 11111111,11111111,11111111,00000000$$

Paso 2: Aplicación de la operación AND
Se realiza una operación lógica AND bit a bit entre la dirección IP y la máscara de subred:

$$\begin{array}{rl} & 11001000,00011011,00000100.\underline{01110000} \\ \& & 11111111,11111111,11111111,00000000 \\ \hline \text{Resultado:} & 11001000,00011011,00000100,00000000 \end{array}$$

Paso 3: Conversión a decimal
El resultado de la operación AND en binario se convierte nuevamente a su forma decimal para obtener la dirección de la subred:

$$\text{Subred:} \quad \mathbf{200},\mathbf{27},\mathbf{4},\mathbf{0}$$

Elección de máscara

La máscara se selecciona según el **número de dispositivos** previstos en la subred, de manera que no se desaprovechen direcciones. Cada subred tiene un identificador único, ya sea en una intranet (direcciones privadas) o en internet (direcciones públicas).

Ejemplo:

$$\text{Dirección IP:} \quad 200,27,4,112 = 11001000,00011011,00000100,01110000$$

$$\text{Máscara:} \quad 255,255,255,0 = 11111111,11111111,11111111,00000000$$

El número de **dispositivos** se calcula con la fórmula:

$$\#\text{dispositivos} = 2^{\#\text{ceros}} - 2$$

Donde el -2 corresponde a:

- **Dirección de subred (000...0):** Reservada para identificar la subred.
- **Dirección de difusión (111...1):** Reservada para broadcast en la subred.

Ejemplo: Con una máscara de **/24** (8 ceros), se permite un máximo de **254 dispositivos**.

Rango de direcciones en la subred **200.27.4.0/24:**

- **200.27.4.0:** 11001000,00011011,00000100.**00000000** ⇒ Reservada (subred)
- **200.27.4.1:** 11001000,00011011,00000100.**00000001** ⇒ Dispositivo #1
- ...
- **200.27.4.254:** 11001000,00011011,00000100.**11111110** ⇒ Dispositivo #254
- **200.27.4.255:** 11001000,00011011,00000100.**11111111** ⇒ Reservada (difusión)

Direcciones públicas y privadas

- Direcciones públicas
  - Identificador único en internet
  - Asignadas de manera centralizada
- Direcciones privadas
  - Identificador único en intranet
  - Se pueden repetir en distintas intranets.
  - Las asigna el usuario según su criterio

Clases

Tradicionalmente, se definieron cinco clases de direcciones IP, cada una con un rango específico y un propósito particular. A continuación, se detallan sus características y rangos:

Rangos:

- **Clase A (/8):** 0,0,0,0 → 127,255,255,255
  - 128 redes.
  - Cada red puede tener hasta 16,777,216 **hosts**.
- **Clase B (/16):** 128,0,0,0 → 191,255,255,255
  - 16.384 redes.
  - Cada red puede tener hasta 65,536 **hosts**.
- **Clase C (/24):** 192,0,0,0 → 223,255,255,255
  - 2.097.152 redes.
  - Cada red puede tener hasta 256 **hosts**.
- **Clase D:** 224,0,0,0 → 239,255,255,255
  - Reservada para **multicast**.
- **Clase E:** 240,0,0,0 → 255,255,255,255
  - Reservada para usos futuros.

Distribución de bits: Cada clase se define según los bits más significativos (**MSB, Most Significant Bits**) del primer octeto. La distribución es la siguiente:

- **Clase A:** 0 (1 bit) seguido por 7 bits para la red y 24 bits para el host.
- **Clase B:** 10 (2 bits) seguido por 14 bits para la red y 16 bits para el host.
- **Clase C:** 110 (3 bits) seguido por 21 bits para la red y 8 bits para el host.
- **Clase D:** 1110 (4 bits) seguido por 28 bits para multicast.
- **Clase E:** 11110 (5 bits) seguido por 27 bits reservados.

Reglas especiales:

- # Dispositivos en una red = $2^{\#\text{ceros}} - 2$ (se resta 2 debido a las siguientes reservas):
  - **Host** $00\ldots0$**:** Identifica la red, no se utiliza como dirección de dispositivo.
  - **Host** $11\ldots1$**:** Dirección de difusión (*broadcast*) para todos los dispositivos de la red.
  - **127.0.0.0:** Reservado para **loopback** o autobucle.

Reserva de direcciones privadas (RFC 1918): Ciertas direcciones están reservadas para uso privado en redes internas y no se enrutan en Internet. Estas son:

- **Clase A:** 10,0,0,0 → 10,255,255,255 (1 red privada clase A).
- **Clase B:** 172,16,0,0 → 172,31,255,255 (16 redes privadas clase B).
- **Clase C:** 192,168,0,0 → 192,168,255,255 (256 redes privadas clase C).

3.3. NAT

Características

- Es un método para **reasignar un espacio de direcciones IP** (privadas a públicas) modificándolas mientras se transmiten a través del router
- Usa la **tabla de traducciones**
- No se instalan servidores detrás de un NAT con direcciones privadas porque los harían inaccesibles desde el exterior.

Ejemplo

1. **Paso 1: Solicitud del cliente.** El cliente con dirección IP privada `10.0.0.3` intenta conectarse al servidor web público `128.32.32.68`. Envía un paquete TCP con una solicitud `SYN`, indicando el puerto de origen `1049` y el puerto de destino `80`, utilizando su dirección interna como dirección de origen (`SADDR = 10.0.0.3`).
2. **Paso 2: Traducción en el NAT interno.** El dispositivo NAT, ubicado en la red interna (`10.0.0.1`), intercepta el paquete.
   - Añade una nueva entrada en su tabla de traducción, asociando la dirección y el puerto internos del cliente con la dirección y el puerto externos del NAT.
   - Modifica el paquete:
     - Sustituye la dirección de origen `SADDR` por la dirección pública del NAT (`24.1.70.210`).
     - Cambia el puerto de origen `SPORT` a uno asignado por el NAT (`40960`).
     - Recalcula el chequeo de integridad (`CKSUM`).
3. **Paso 3: Envío del paquete al servidor.** El paquete modificado se envía a través de Internet al servidor web `128.32.32.68`. El servidor recibe el paquete con dirección de origen `24.1.70.210` y puerto `40960`, y responde con un paquete de retorno que incluye `SYN` y `ACK`.
4. **Paso 4: Respuesta del NAT.** Cuando el paquete de respuesta del servidor llega al NAT:
   - El dispositivo NAT consulta su tabla de traducción y encuentra la entrada correspondiente a la dirección de destino `24.1.70.210` y puerto `40960`.
   - Reescribe el paquete para restablecer las direcciones y puertos originales:
     - Dirección de destino (`DADDR`) pasa a ser `10.0.0.3`.
     - Puerto de destino (`DPORT`) pasa a ser `1049`.

   Finalmente, el paquete es entregado al cliente con las direcciones internas restauradas.

3.4. Asignación de direcciones

La asignación de direcciones IP busca **minimizar las IPs no utilizadas** y simplificar el encaminamiento. Para lograrlo, es crucial elegir adecuadamente las máscaras de subred para ajustarlas al número necesario de dispositivos.

- En el caso mostrado, se trabaja con:
  - **3 subredes corporativas** (por ejemplo, direcciones privadas `192.168.0.0/24`):
    - Cada subred requiere 30 dispositivos.
    - Máscara seleccionada: `/27` (5 bits para los dispositivos, lo que permite hasta 30 hosts).
  - **1 subred de acceso** (por ejemplo, una dirección proporcionada por el ISP `150.214.190.0/30`):
    - Requiere 2 dispositivos (por ejemplo, un router y el ISP).
    - Máscara seleccionada: `/30` (2 bits para dispositivos).

Detalles técnicos:

- **Máscara de subred:** Determina el número de direcciones disponibles en la subred.

  Ejemplo: 192,168,0,0 con máscara /27 = 255,255,255,224

  Bits para dispositivos: $2^{\#\text{bits ceros}} - 2$. En este caso, 5 bits $\Rightarrow 30 dispositivos$.

- **Direcciones:**
  - Dirección de red (todos ceros en los bits de dispositivo).
  - Primera dirección asignable: `192.168.0.1`.
  - Última dirección asignable: `192.168.0.30`.
  - Dirección de difusión (todos unos en los bits de dispositivo).

Criterios en la asignación

1. **Reducir el número de direcciones IP** desperdiciadas ajustando máscaras.
2. **Reducir las filas en las tablas de encaminamiento** haciendo subredes agregables.

Reglas para asignación

Hay que ordenar las subredes de mayor a menor tamaño. Para cada subred:

1. Determinar **máscara en función del tamaño**.
2. Elegir **identificador** de red **no usado** previamente.
3. **Cuidar agregabilidad** de las subredes bajo prefijos fácilmente acumulables.

3.5. Encaminamiento

- Consiste en **encontrar el mejor camino** para llevar la información
- Se realiza **paquete a paquete** y **salto a salto**, en función de la IP destino y de las tablas de encaminamiento.
- En cada salto hay un store & forward
- Existe el **modo directo y modo no directo**.
- Cada dispositivo tiene una tabla de encaminamiento.
- Un **router** está en **varias redes**.
- Un **host** está en **una sola red**.
- En caso de **conflicto** en la tabla, se elige la ruta con la **máscara más larga**.
- El default no es obligatorio pero simplifica las tablas

Traducción de direcciones (NAT)

El datagrama necesita del NAT para modificar el camino en ciertos puntos.

Proceso de encaminamiento por datagrama

1. Se extrae la dirección destino **IP_DESTINO**
2. por cada entrada I se calcula

```
IPi = IP_DESTINO & MASCARA_i
```

3. Comprobación IPi
   - Si IPi == Di (campo destino de la tabla) de manera **única**
     - Routing directo: Reenvía al destino final
     - Routing indirecto: Reenvía al salto siguiente
   - Si hay **varias coincidencias** se elige el destino con la **máscara más larga**
   - Si **no hay coincidencias** encontramos un **error**.

Para encapsular el datagrama se consulta la tabla ARP para obtener la MAC. Si no se conoce se envía un broadcast con el protocolo ARP para obtenerla.

3.6. Sistemas autónomos

Conjunto de redes administradas por una **única autoridad** que define el routing interno. Características:

- **Facilitan la administración** y aumentan la **escalabilidad**.
- En cada sistema existe **un router exterior** responsables de **informar** a los **otros** sistemas autónomos.
- Cada SA se identifica por un entero de 16 bits

3.7. Intercambio automático de tablas

Niveles

Se definen dos niveles:

- **Protocolos interiores (IGP)**: Actualizan las tablas dentro del SA. El **administrador tiene libertad de elección** para el protocolo de intercambio de tablas. Ejemplos:
  - RIP
  - OSPF
  - HELLO
- **Protocolos exteriores (EGP)**: Facilita el intercambio de información entre SAs. Lo usan solo los **routers exteriores** y se denomina **BGP**

RIP (Routing information protocol

- Protocolo de la **capa de aplicacion**. Opera sobre **UDP** en el puerto **520**.
- Adopta **algoritmo vector-distancia** usando como **métrica** el **número de saltos**.
- Cada router RIP **recibe y envía** de manera **periódica** vía **multi-cast** los vectores distancia a sus posibles destinos.
- Se selecciona como **siguiente salto** el vecino con **menor coste**, actualizando la métrica aumentando 1 post-salto.
- Las noticias **tardan en propagarse** (DESVENTAJA)
- Existen problemas como la **convergencia lenta** y la **cuenta al infinito**.

OSPF (Open Short-Path First)

- **Divide** el SA en **áreas independientes** para permitir **escalabilidad**.
- Está basado en el **estado del enlace** (coste **1/velocidad enlace**)
- **Informa a todos** los routers del área sobre el coste de los vecinos.
- **Crea un grafo** de la topología con esta información.
- Se **aplica el algoritmo de Dijkstra** a dicho grafo.
- Define **rutas alternativas** y **balanceo de carga**.
- **Minimiza la difusión** eligiendo **un solo router** designado

3.8. Formato de datagrama IP

El formato del datagrama IP incluye dos secciones principales: la **cabecera** y los **datos**. A continuación, se describen los campos presentes en la cabecera y su función:

- **V (4 bits):** Versión del protocolo IP. Identifica si el paquete pertenece a IPv4 o IPv6.
- **LC (4 bits):** Longitud de la cabecera. Especifica el tamaño de la cabecera en múltiplos de 4 bytes.
- **TS (8 bits):** Tipo de servicio. Define la prioridad del paquete en la red.
- **Longitud total (16 bits):** Indica el tamaño total del datagrama, incluida la cabecera y los datos (máximo 65.535 bytes).
- **Identificación (16 bits):** Etiqueta única utilizada para fragmentación y reensamblaje de paquetes.
- **I (3 bits):** Indicadores:
  - Bit DF (*Don't Fragment*): Indica si el paquete puede ser fragmentado.
  - Bit MF (*More Fragments*): Señala si existen más fragmentos de este paquete.
- **Desplazamiento (13 bits):** Posición del fragmento dentro del datagrama original.
- **TTL (8 bits):** Tiempo de vida (*Time To Live*). Indica el número máximo de saltos antes de descartar el paquete.
- **Protocolo (8 bits):** Especifica el protocolo de la capa superior al que deben entregarse los datos (e.g., TCP, UDP, ICMP).
- **Comprobación (16 bits):** Campo de verificación que detecta errores en la cabecera.
- **Dirección IP origen (32 bits):** Dirección del dispositivo que envió el paquete.
- **Dirección IP destino (32 bits):** Dirección del dispositivo que debe recibir el paquete.
- **Opciones (tamaño variable):** Opcional. Permite funcionalidades adicionales como enrutamiento específico o registro de tiempo.
- **Relleno (tamaño variable):** Asegura que la cabecera termine en un múltiplo de 4 bytes.
- **Datos:** Contenido transportado por el datagrama, correspondiente a las capas superiores (e.g., TCP, UDP).

Nota: La cabecera básica tiene un tamaño mínimo de 20 bytes, pero puede incrementarse si se incluyen opciones adicionales. El diseño de la cabecera permite el transporte eficiente y la fragmentación de datos, adaptándose a diferentes redes.

3.9. Fragmentación

- Tamaño máximo es de **2^{16}-1**
- Es necesario **adaptarse a la MTU** de la subred
- El **ensamblado** se hace **al final**
- El fragmento se especifica en el desplazamiento de la cabecera.
- Mediante los indicadores de la cabecera (I) se indica si puede fragmentarse el paquete o no.

3.10. Diferencias entre IPv4 e IPv6

Característica	**IPv4**	**IPv6**
Longitud de la dirección	32 bits	128 bits
Espacio de direcciones	4.3 millones	Casi ilimitado
Formato	Decimal	Hexadecimal
Configuración	Manual o DHCP	Autoconfiguración
Seguridad	IPsec opcional	IPsec obligatorio
Fragmentación	Routers pueden fragmentarse	Solo emisor fragmenta
Encabezado Paquetes	Complejo y variable	Simplificado y fijo
QoS	Limitado (TOS)	Optimizado (Flow Label)
Compatibilidad	Limitado	Incompatible
Optimización moviles	Ineficiente	Mejor rendimiento

4. Asociación con capa de enlace (ARP)

4.1. Direcciones MAC

- Tras consultar tabla, se **envía** el datagrama a la **dirección MAC** (por ethernet y wi-fi)
- Su formato es de **6 bytes** y están expresadas en **hexadecimal**: HH-HH-HH-HH-HH-HH
- Son **únicas**
- Existe definida una dirección de difusión.
- Protocolo: **ARP** (Address Resolution Protocol). Obtiene la **dirección MAC** a partir **de la IP**
- Protocolo **RARP**. Obtiene la **dirección IP** a partir **de la MAC**

5. Protocolo ICMP

- Protocolo de señalización que **informa sobre errores** IP
- Suele ir hacia el origen del datagrama
- **Se encapsula en IP**
- Cuenta con una cabecera de 32 bits donde se incluye la cabecera del diagrama

6. Autoconfiguración de la capa de red (DHCP)

El protocolo **DHCP** (*Dynamic Host Configuration Protocol*) se utiliza para asignar direcciones IP de forma automática en una red. Está especificado en los RFC 2131 y 3396 y utiliza el protocolo UDP, con los siguientes puertos:

- **Puerto 67:** utilizado por el servidor DHCP.
- **Puerto 68:** utilizado por el cliente DHCP.

Proceso de asignación de direcciones

El proceso de asignación de direcciones IP mediante DHCP sigue los siguientes pasos:

1. **DHCP Discover:** El host (cliente) envía un mensaje de difusión (*broadcast*) para buscar un servidor DHCP disponible. Este mensaje tiene las siguientes características:
   - Origen: 0.0.0.0, puerto 68.
   - Destino: 255.255.255.255, puerto 67.
   - Contenido: Identificador único (*ID*) del cliente.
2. **DHCP Offer:** El servidor DHCP responde al cliente con una oferta de configuración, incluyendo una dirección IP disponible. Este mensaje tiene las siguientes características:
   - Origen: Dirección IP del servidor, puerto 67.
   - Destino: 255.255.255.255, puerto 68.
   - Contenido:
     - Dirección IP ofrecida (*SudirIP*).
     - Tiempo de vida (*lease time*).
     - Identificador único (*ID*).
3. **DHCP Request:** El cliente selecciona una de las ofertas recibidas y solicita la dirección IP al servidor. Este mensaje tiene las siguientes características:
   - Origen: 0.0.0.0, puerto 68.
   - Destino: 255.255.255.255, puerto 67.
   - Contenido:
     - Dirección IP solicitada.
     - Tiempo de vida.
     - Identificador único (*ID*).
4. **DHCP Ack:** El servidor confirma la asignación de la dirección IP al cliente. Este mensaje tiene las siguientes características:
   - Origen: Dirección IP del servidor, puerto 67.
   - Destino: 255.255.255.255, puerto 68.

- Contenido:
  - Dirección IP asignada.
  - Tiempo de vida.
  - Identificador único (*ID*).

Esquema de funcionamiento

El flujo del proceso puede resumirse de la siguiente manera:

1. El cliente envía un mensaje **DHCP Discover** en busca de un servidor DHCP.
2. El servidor DHCP responde con un **DHCP Offer**, ofreciendo una dirección IP.
3. El cliente solicita la dirección ofrecida mediante un mensaje **DHCP Request**.
4. El servidor confirma la asignación con un **DHCP Ack**.

Este proceso permite la configuración automática y dinámica de los dispositivos conectados a una red, simplificando la administración de direcciones IP.