

# Resumen parcial 1

Federico Polidoro

## Contents

<b>Clase 1 - Ethernet/Internet</b>	<b>2</b>
Ethernet . . . . .	2
MAC . . . . .	2
Arquitectura de red . . . . .	3
Modelos . . . . .	3
Servicios a Nivel de Red . . . . .	3
Tecnología . . . . .	4
<b>Clase 2 - ARP/RARP</b>	<b>4</b>
Direcciones IP . . . . .	4
Ipv6 . . . . .	5
Arp . . . . .	5
Rarp . . . . .	6
<b>Clase 3 - ICMP - IP</b>	<b>6</b>
Paquetes . . . . .	6
Headers . . . . .	6
Mtu . . . . .	6
Fragmentacion Ipv4 . . . . .	6
ICMP . . . . .	7
<b>Clase 4 - Tcp/ip Core - Udp</b>	<b>7</b>
Udp vs Tcp . . . . .	7
Udp . . . . .	7
Puertos . . . . .	7
Tcp . . . . .	8
<b>Clase 6 - SubRedes</b>	<b>8</b>
Proxy Arp . . . . .	8
Mascaras . . . . .	8
Vlsm . . . . .	9

# Clase 1 - Ethernet/Internet

## Ethernet

- Entrega con el mejor esfuerzo  
El hardware no proporciona informacion sobre si el paquete fue recibido.
- Tecnologia de difusion  
Todas las estaciones comparten un canal
- Control contra colisiones  
Usa CSMA/CD

## Usa los medios de..

1. Utp (T)
2. Coaxial (C o un numero solo | CR)
3. Fibra Optica (F | S | L | E)

## Trama ethernet

preambulo	destino	origen	tipo	payload	CRC
8oct	6oct	6oct	2oct	64 a 1500 oct	4 oct

## MAC

Son las direcciones fisicas del hardware que se conecta a la red todos los terminos que te parezcan que se puedan referir a esto son tratados por ogial ( MAC, direccion fisica, hardware unicast addr)

Los tipos de direcciones pueden ser:

- Unidifucion,  
Pensa en cuando te conectas a un socket solo.
- Difusion,  
Como cuando mandas un paquete a 192.168.1.255 y el router lo envia por broadcast a todos los dispositivos conectados.
- Multidifucion,  
Envia a las direcciones especificadas

## Esquemas de direccionamiento

- Estatico,  
addr por el fabricante
- Configurable,  
addr por el usuario

- Dinamico,  
addr por el software en startup,

## **Arquitectura de red**

Su objetivo es abstraer los detalles de la red. esto lo logra a traves de aplicaciones de nivel de sistema operativo. o a nivel de red.

Para lograrlo a nivel de red se utilizan ruteadores. Estos son dispositivos que se encuentran entre la conexión de una red con otra, **Los ruteadores utilizan la red de destino no la maquina para rutear el paquete**

### **Propiedades**

- Conectividad fin a fin.
- Transparencia

## **Modelos**

### **OSI**

Las capas no definen el protocolo sino funciones de comunicación que pueden ser cumplidas por los protocolos

Las capas no necesitan saber como funcionan las demás solo como envian o reciben los datos entre ellas.

### **TCP/IP**

Usa menos capas que el modelo OSI

1. Capa de Acceso de red.
2. Capa de Internet.
3. Capa de Transporte.
4. Capa de Aplicación.

## **Servicios a Nivel de Red**

### **Sin Conexión de entrega de paquetes**

- Se rutea basandose en la dirección.
- Cada Paquete se Rutea por separado.
- No garantiza entrega.
- No garantiza orden de llegada.

### **Transporte de flujo confiable**

- Establece conexión entre las computadoras.
- Trabaja con acuse de recibo.

## Tecnología

- Internet no es un nuevo tipo de red,  
Es un metodo de interconexión.
- Redes WAN – Redes LAN,  
WAN : Mas lentas, Delay mayores, Altas tasas de errores  
LAN : Rápidas, Bajo Delay, Bajas tasas de errores

## Clase 2 - ARP/RARP

### Direcciones IP

Su objetivo es que cualquier computadora se pueda conectar con cualquier otra.

Las direcciones ipv4 tienen una longitud de 32 bits.

- Son similares a las direcciones mac
- Unicas por host en el internet
- Busca un ruteo eficiente
- Se dividen en dos partes:  
Host y Red

### Características deseables

- Compacta
- Universales
- Independientes del hardware

### Direcciones A,B y C

class	netid	hostid
A	8bit	24bit
B	16bit	16bit
C	24bit	8bit

### Direcciones

Estas representan una conexión a la red mas no una computadora.

- Cuando el hostid = 0 nos referimos a la red (192.168.1.0)
- Cuando el hostid = 255 representa la ip de broadcast (192.168.1.255)

Al rutear se usa la netid

### **direcciones especiales**

- 0.0.0.0 Este host
- red y todo 1 broadcast
- 127.0.0.1 loopback

## **Ipv6**

### **Limitaciones v4**

las ipv4 ya estan agotadas. y no se pueden diferenciar los datagramas udp de los tcp cuando estan encriptados

### **Funcionalidad**

- Nuevo header,  
minimiza el procesamiento
- Más direcciones,  
Busca eliminar la necesidad de tener nat
- IPsec,  
Provee Integridad y auth de datos
- ICMPv6,  
Mejora las interacciones entre nodos

### **Direcciones**

Permite  $6.6 * 10^{23}$  direcciones.

y los tipos de direcciones pueden ser:

- Unicast,  
hacia un dispositivo.
- Multicast,  
para grupos, nodos o interfaces.
- AnyCast,  
Unicast a varios dispositivos

## **Arp**

Para que dos maquinas se comuniquen deben de conocer las direcciones fisicas. El usuario lo que conoce es la direccion ip. por lo que arp en base a una direccion ip te da una direccion mac. Utiliza un broadcast a los dispositivos para que el que tiene la ip coincidente con la del destino responda con su direccion mac.

**No** es utilizado por ipv6.

Para evitar hacer muchas peticiones se almacena una tabla ARP y normalmente se eliminan las entradas cada 20 mins.

### Rarp

Es lo contrario a arp, con una direccion mac te devuelve una direccion ip. usado en maquinas sin disco duro, muy antiguo. Reemplazado por Dhcp.

## Clase 3 - ICMP - IP

### Paquetes

Tambien conocidos como datagramas son un servicio no orientado a la conección el cual consiste de un header y un payload.

### Headers

Es de una longitud de 20bytes en **ipv4**, mientras que **ipv6** es de 40bytes.

### Comparacion

En ipv6 se quito la checksum, la flag, el header lenght.

### Mtu

\*M\*aximun \*T\*ransfer \*U\*nit

Consiste en el maximo tamaño que puede tener un datagrama en octetos.

- Link Mtu,  
define el limite en la capa de datos.
- Path Mtu,  
define el limite con lo maximo que pueden ofrecer los nodos intermedios entre el emisor y receptor.
- Mtu min ipv6,  
1280 bytes

### Fragmentacion Ipv4

Los Routers dividen los datagramas en fragmentos que se rutean individualmente. En ipv6 no se fragmentan.

un caso donde se emplea la fragmentacion es cuando un router "a" Tiene un Mtu de 4000 y otro "b" tiene uno de 400 y se quiere enviar un paquete de 4000 octetos. lo que sucede es que el paquete se tiene que fragmentar en 10 fragmentos que se rutean del router "a" al "b". Donde el router "a" se encarga del fragmentado.

## **Reensamblado**

Se realiza en el destino final cuando llega el primer fragmento. si el datagrama no se puede completar este se descarta.

## **ICMP**

Es un mecanismo de reporte de errores utilizado por los routers e informan solo al origen. (internamente funcionan sobre ip)

## **Clase 4 - Tcp/ip Core - Udp**

### **Udp vs Tcp**

Udp	Tcp
No orientado a la conexion	Orientado a la conexion
Mejor Esfuerzo	Lento pero Confiable

### **Udp**

Es un protocolo que no maneja conexiones ni tiene interes en que los datos lleguen en forma,

La responsabilidad de la correccion de errores corresponde a la aplicacion.

Trabaja en la capa de transporte.

El maximo tamaño de un datagrama es de 65535 bytes. el header ip es de 20/40 bytes y el udpo es de 8 bytes

### **CheckSum**

Es opcional pero recomendada, lo de opcional viene porque udp esta pensado en que las implementaciones operen con la menor cantidad de trabajo computacional.

### **Socket**

Un define un punto extremo de la conexión en el caso de udp/tcp es la dirección ip + puerto.

### **Puertos**

- Los menores a 1024 son **well known**. En linux los únicos procesos que pueden utilizar estos puertos son los que tienen acceso root.
- Los mayores a 1024 los puede usar cualquiera y pueden tener cualquier servicio.

## **Tcp**

Utiliza Ip para transmitir datos. Ofrece un servicio de transferencia de datos eficiente y confiable a los programas. Compensa las perdidas y retardos sin sobrecargar redes ni enrutadores.

### **Caracteristicas**

- Full Duplex (transfiere en ambas direcciones)
- Fiable

### **Confiable**

Tiene un handshake de 3 direcciones para asegurar la confiabilidad.

1. El receptor envia ACK cuando llegan los datos
2. El emisor inicia un timer con la emision.
3. Si no llega ACK se Reenvian los datos.

### **Ciclo Tcp**

1. Establecimiento de conexion.
2. Transferencia de datos.
3. Cierre de conexion.

### **Viaje en Redondo - Tiempo**

$$RTT = A * RTT + (1-A) NRTT$$

Siendo que NRTT sea el ultimo RTT medido

## **Clase 6 - SubRedes**

### **Proxy Arp**

Es cuando hay varias subredes y una red "original" y una maquina (M) en la red original la cual le manda una request ARP a la maquina (N) en la subred 1 al estar en otra red la request no le va a llegar. pero como el router de la subred 1 si se entera de la request ARP este le va a reenviar la request a las maquinas que tiene en la red, en caso de que una responda el router se va a hacer pasar por la maquina que respondio y le va a enviar la direccion mac de la maquina (N) a (M)

### **Mascaras**

Es una palabra de 32 bit donde los 1 representan los bits del netid y los 0 los del hostid. No necesitan ser contiguos pero se recomienda.

En una red tipo c tenemos que tener en cuenta que el ultimo octeto tenemos que definir cuantos bits vamos a usar para diferenciar las subredes

entonces si tenemos una mascara /24 y tenemos que hacer 6 subredes para 25 usuarios hariamos

calcular cantidad de host:

$$2^5 - 2 = 30$$

Calculamos la longitud de la mascara de red:

$$32 - 5$$

y como vamos a usar 5 bits para los hosts nos quedan 3 (osea 8-5) bits para definir hasta 8 subredes de 30 hosts.

## Vlsm

Con este metodo podemos subnetear para que se refleje la cantidad exacta de hosts necesarios en cada punto de la red. El apunte tambien lo describe como direccionamiento jerarquico.

### Pre requisitos

1. verificar que los enrutadores con los que se cuenta todos soportan protocolos que permiten VLSM
2. Los enrutadores deben de basarse en "la mayor coincidencia" Notese que mientras más grande es el numero de bits significativos compartidos entre redes menor es el grupo de ips a las que dirige. entonces la mejor ruta es la que cuenta con el /n mayor

### Ejemplo simple VLSM

En una empresa con 3 sucursales y una casa central unidas por una red wan

- Londres: 28 hosts.
- n. york: 12 hosts.
- Tokio: 12 hosts.
- Casa central: 60 hosts.

Se Asigna la direccion 192.168.10.0/24 Como solo necesitamos 3 subredes vamos a dejar 2 bit para direccionamiento

|#0|192.168.10.0/26|

#1	192.168.10.64/26
#2	192.168.10.128/26
#3	192.168.10.192/26

Entonces para #0:

1 <sup>er</sup> host	192.168.1.1
ultimo host	192.168.1.62
broadcast	192.168.1.63

Si tomamos las de #1 y las asignamos para la sede de londres tendriamos que como londres solo necesita 28 hosts podemos usar una mascara de /27 (32-5) sabiendo que  $2^5 = 32$ . esto nos deja con 3 bit de direccionamiento para poder usar los hosts que sobran en otra subred

#1.0: Londres: 192.168.1.64/27

1 <sup>er</sup> host	192.168.1.65
ultimo host	192.168.1.94
broadcast	192.168.1.95

La sede de n.york necesita 12 hosts y en #1.1 tenemos 32-2 hosts disponibles por lo que subnateamos otra vez con mascara de /28 #1.1.0

IP Red	192.168.1.96/28
1 <sup>er</sup> host	192.168.1.97
ultimo host	192.168.1.110
broadcast	192.168.1.111

y para la subred de tokio usamos #1.1.1

IP Red	192.168.1.112/28
1 <sup>er</sup> host	192.168.1.113
ultimo host	192.168.1.126
broadcast	192.168.1.127