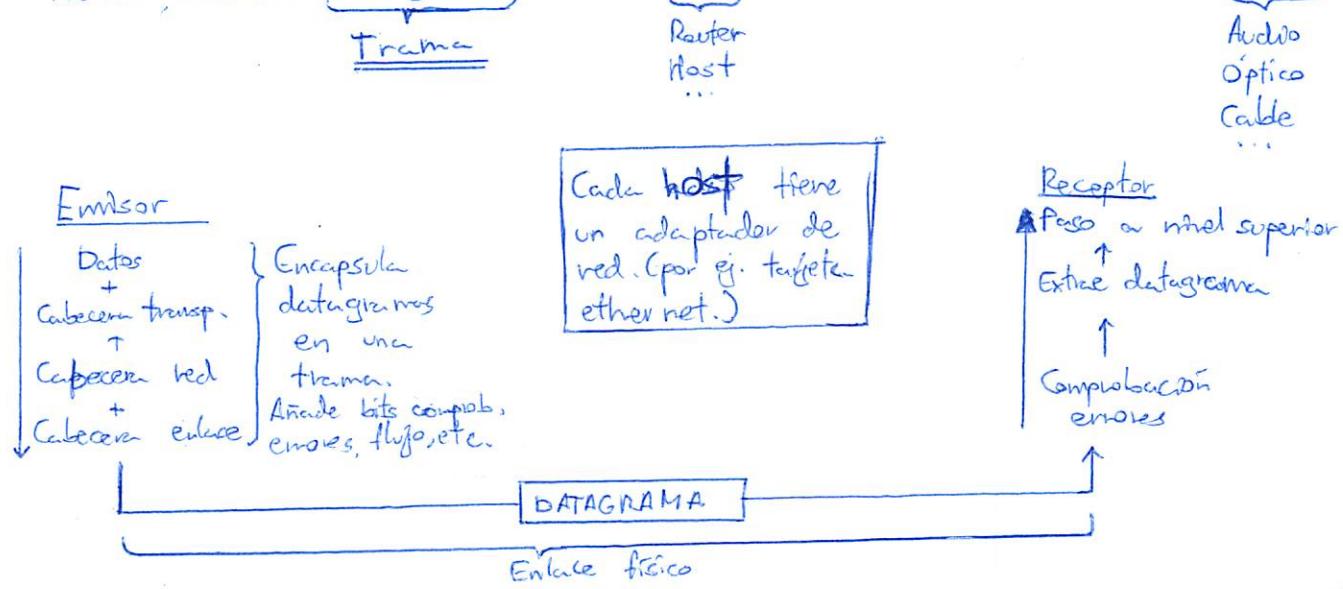


⑤ Capa de Enlace

Misión: transmitir datagramas de un nodo a otro a través de un enlace.



① Detección de errores

Los errores son causados por el ruido del medio físico por donde se transmiten los bits. El receptor debe detectar la presencia de errores, y en función de eso, retransmitir o descartar.

La detección de errores no es 100% fiable.

Al datagrama el emisor le añade un campo EDC. (Detección y corrección de errores).
El receptor comprobará que el datagrama y el EDC son congruentes.

② Mecanismos para la detección de errores

③ Comprobación de paridad: suma todas las 1's en mod 2. Detecta ^{impares} un único error de bit.

Ejemplo: Datagramas 0111 0001 1010 1011 ; Bit de paridad= 1. Trama = 0111 0001 1010 1011 1

④ Paridad bidimensional: detecta errores en filas y columnas, poniendo en filas las tramas

	bit ₁	bit ₂	...	bit _m	Cálculo paridad filas
trama 1	t _{1,1}	t _{2,1}	...	t _{r,1}	p ₁ -trama hor. 1
trama 2	t _{1,2}	t _{2,2}	...	t _{r,2}	p ₂ -trama hor. 2
...
trama n	t _{1,n}	t _{2,n}	...	t _{r,n}	p _n -trama hor. n
	p _{1,trama}	p _{2,trama}	...	p _{n,trama}	

Ejemplo

1	0	1	0	1	1
1	1	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	0

Emisor:

Trama + parh + parv

1	0	1	0	1	1
1	0	1	1	0	0
0	1	1	1	0	1
0	1	1	0	1	0

Receptor

⚠ Paridad vertical emisor
 \neq
 Paridad vertical receptor.
 ¡Hay error!

③ Checksum similar al de internet.

④ CRC: Código de redundancia cíclico.

Se realiza una correspondencia entre tramas y polinomios.

D: polinomio asociado a datos

G: polinomio generador estandar. (lo conocen emisor y receptor).

Emisor envia trama con $D + G$. Receptor intenta dividir $(D+G)/G$.

Resto = 0 \rightarrow No hay error.

Resto $\neq 0 \rightarrow$ Error.

Ejemplo:

$$D = D_{inv} \cdot G = 101110000; \quad G = 1001;$$

$$\begin{array}{r} D' = \\ \hline G \end{array}$$

$$\begin{array}{r} 0010101110000 \\ 1001 \\ \hline 001010000 \end{array} \} \text{XOR}$$
$$\begin{array}{r} 1001000 \\ \hline 1001 \end{array} \} \text{XOR}$$
$$\begin{array}{r} 0011000 \\ 1001 \\ \hline 00100 \\ 1001 \\ \hline 01010 \\ 1001 \\ \hline 001 \end{array} \} \text{R}$$

$R \neq 0 \Rightarrow \Delta$ Error.

Ejemplo 2.

$$D' = 001011000110110110010011$$
$$G = X^6 + X^5 + X^3 + X^2 + X + 1 \Leftrightarrow 1011011$$

$$\begin{array}{r} 001011000110110110010011 \\ \hline 1011011 \\ \hline 000000010110110110010011 \\ \hline 1101111 \\ \hline 01110100010011 \\ \hline 1101111 \\ \hline 0000111010010011 \\ \hline 1101111 \\ \hline 0000000 \end{array} \} R$$

$R=0$. Éxito ✓

• Enlaces de acceso múltiple y protocolos

Los enlaces de difusión el medio (enlace) es compartido por los host.

Si solo se tiene un canal y hay dos o más nodos transmitiendo, se pueden producir interferencias.

• Colisión: un nodo recibe 2 o más señales a la vez.

• Alg. distribuido: determina cómo los nodos comparten el canal.

La comunicación sobre cómo compartir el canal va sobre el propio canal.

• Protocolos de acceso múltiple

1) Partitionar el canal: dividir el canal en partes y asignar una parte a cada nodo.

2) Acceso aleatorio: permitir colisiones y poder recuperarnos de ellas.

3) Toma de turnos: hay un mecanismo de sincronización que asigna turnos variables a los nodos.

Generar datos crc:

$$D = 10001110;$$

$$G(x) = x^3 + 1 = 1001$$

datos add. pág. grados 3 ✓ $g(x)$

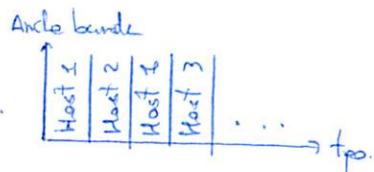
$$\begin{array}{r} 10001110 \ 000 \ 1001 \\ \hline 00001111 \end{array}$$

• Protocols MAC - Partición de canal

1) TDMA (División por tiempo).

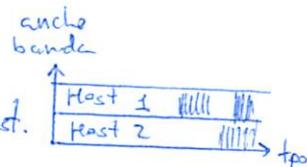
Los nodos acceden al canal por bandas (slots), de longitud fija.

Los slots no utilizados quedan inactivos (es ineficiente).



2) FDMA (División por frecuencia).

Se divide el ancho de banda y se asigna una porción a cada host por frecuencias.



3) Div. Código: asignamos un código a cada nodo. Al multiplicarlo por otro distinto, da 0.

Ej.: User 1 envía 00. Códigos: 0101. Envía por cable. ①

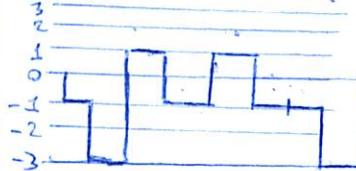
$$\begin{array}{r} 0101 \\ \times 0000 \\ \hline 0000 \end{array}$$

0101 0000 => Resultado

② Envía resultado:

$$\begin{array}{l} \text{User 1} \quad 0101 \quad 0101 \\ 0 \Rightarrow +1 \quad +1 \quad \text{[onda positiva]} \\ 1 \Rightarrow -1 \quad -1 \quad \text{[onda negativa]} \end{array}$$

③ Se acopla con señales de otros usuarios.



④ Multiplicamos el código de user 1 por la señal de todos los users.

$$\begin{array}{r} 0101 \\ \times 0101 \\ \hline 0000 \end{array}$$

-1 + 3 + 1 + 1 + 1 = 4; 4/4 = +1 V; ≈ 0 [Datos] 1 + 1 - 1 + 3 = 4; 4/4 = +1 V; ≈ 0 [del user 2].

• Protocols MAC - Acceso aleatorio

Pueden darse colisiones ya que no hay coordinación.

Deberemos detectar colisiones y poder recuperarnos de ellas.

1) ALOHA banurado

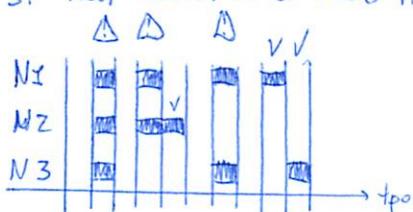
Envío tramas de L bits. Para transmitir una trama, tarda L/R segundos.

El tpo. está dividido en slots de igual tamaño.

Los nodos están sincronizadas con los slots, y comienzan a transmitir al comienzo de estos.

Si dos tramas colisionan en un slot, los nodos detectan la colisión.

Si hay colisión: el nodo tiene una prob. p de enviar la trama en el siguiente slot. (Hasta que la envíe sin colisiones).



Eficiencia de ALOHA

$N = \text{nº de nodos que transmiten}$
 $p = \text{prob. de transmitir}$

Prob. de que un nodo envíe y el resto no lo hagan

$\underbrace{\qquad\qquad\qquad}_{N-1}$

① Prob. de que al menos un nodo tenga éxito: $N \cdot p \cdot (1-p)^{N-1} = p[\text{exito}]$

② Buscar una p^* que maximice la función. Derivar la prob. de que al menos 1 nodo tenga éxito respecto a p , y la igual a 0 para hallar máximos.

$$\frac{d p[\text{exito}]}{dp} = 0; p^* = \frac{1}{N} \quad \left. \begin{array}{l} \text{? 1 de cada } N \\ \text{nodos debe tener éxito.} \end{array} \right\}$$

③ Para muchos nodos, vemos el límite. $\lim_{N \rightarrow \infty} \left(1 - \frac{1}{N}\right)^{N-1} \leq e^{-1} \approx 0.37 \%$ de éxito.

④ $p[\text{colisiones}] = 1 - p^* - p[\text{no hay nodos transm.}] = 1 - (0.37 \cdot 2) = 0.26$.

• Aloha puro

No hay slots y los nodos no están sincronizados. Por lo tanto no hay puntos fijos de comienzo. Por consiguiente, la prob. de que haya colisiones aumenta.

$$P[\text{Exito de al menos un nodo}] = p \cdot (1-p)^{N-1} \cdot (1-p)^{N-1}$$

El boli azul ha muyerto ::

$$\text{Si hayamos } \lim_{N \rightarrow \infty} \approx e^{-N} \frac{1}{2} e^{-1} \approx 0.18$$

• CSMA

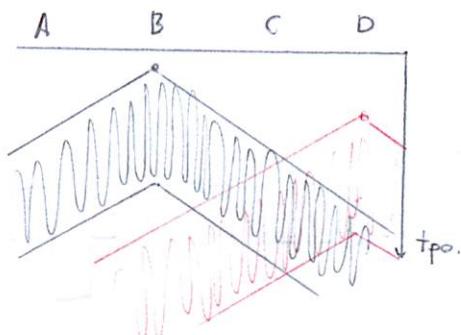
Acceso por sondeos. Si el canal parece desocupado, transmite.

Si, el canal parece ocupado, se espera un tpo. aleatorio.

Puede haber colisiones. Puede que un nodo vea el canal como libre aunque un nodo haya empezado a transmitir, debido al retardo del canal. (Tpropagación).

Para D, el canal estaba vacío en el momento en el que empezó a transmitir.

Pasado el tpo. se produce la colisión y se pierde el paquete.



• CSMA/CD

En el momento en el que se detecta colisión, se deja de transmitir. Así se reduce el tpo. perdido por el canal.

Cables → fácil

R. inalámbricas → difícil.

• Protocolos MAC - Toma de turnos

MAC particionado - inefic. con carga baja (dependencia canal) } infusión! → Toma de turnos.
MAC acceso aleatorio - inefic. con carga alta (colisiones). }

• Sondeo

Un nodo necesita coordinar a los nodos esclavos e indicar cuándo debe transmitir cada nodo. Problema: latencia, centralización.

• Paso de token

Los nodos se van pasando el token. El que tenga dicho token, puede transmitir.

• Direcciones MAC

Identifican la interfaz de red. Permite enviar tramas de una interfaz a otra, físicamente conectadas.

Son de 48 bits y vienen grabados en la tarjeta de red.

Formato de direcciones:

$$\text{Ex.: } I = 1A-2F-BB-76-09-AD$$

$$\text{Difusión} = FF-FF-FF-FF-FF-FF$$

El IEEE asigna MAC's a los vendedores de HW.
MAC → Son portables (No dependen de la subred).

• ARP - protocolo de resolución de redes

Permite determinar la MAC de un host, conociendo su IP.

Cada nodo tiene una tabla ARP, con esta forma:

IP	MAC	TTL	} TTL: tiempo tras el cual el mapeo se blinda
...	

ARP funciona en la misma subred. (No fuera).

Ejemplo

A quiere transmitir a B pero no sabe su MAC.

1) A construye un paquete ARP { IP origen: IP-A, MAC origen: MAC-A }

IP destino: IP-B (Conocida).

MAC destino: FF-FF-FF-FF-FF-FF (difusión).

(No hay que configurar nada).

2) El pkt. llega a todas las nodos. (Por ser difusión).

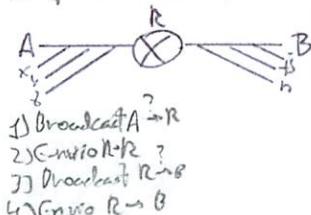
3) B recibe el PKT y responde diciendo cuál es su MAC.

4) A almacena en su tabla ARP { IP-B, MAC-B, TTL }

ARP es
plug and play

Ejemplo 2

A quiere transmitir a B, estando en otra red.

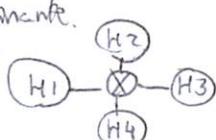


- i) A averigua MAC de R, y lo guarda en su tabla ARP. (ARP-Broadcast).
- 2) A envía datagrama a R con IP-destino = B.
- 3) R averigua MAC de B, y lo guarda en su tabla ARP. (ARP-Broadcast).
- 4) R envía datagrama a B.

• Ethernet

Tecnología LAN dominante.

Topología estandar:



Características

- ① Protocolo no orientado a conexión; no hay handshaking.
- ② No fiable; no hay ACK's. Puede haber datagramas perdidos.
- ③ Usa CSMA/CD para colisiones.

Trama ethernet	8 bytes	6 bytes	6 bytes	2 bytes	2 bytes	On/Off bytes	4 bytes
Preámbulo	MAC Destino	MAC origen	Tipo	Data...	CRC		

• Preámbulo: usado para la sincronización de teléfonos emisor-receptor

• CRC: calcula error sobre resto de trama (menos preámbulo).

• Tipo: indica protocolo de capa superior. (Normalmente IP).

Algoritmo CSMA/CD sobre Ethernet

- 1) La interfaz recibe destagrate y crea trama para enviarla. No colisión: éxito
 Desocupado: transmiso inmediatamente. { Colisión: 3)
 2) La interfaz escucha el canal. Ocupado: espera a que este libre.

3) Si el transmisor detecta otra colisión, aborta y envía señal de congestión.

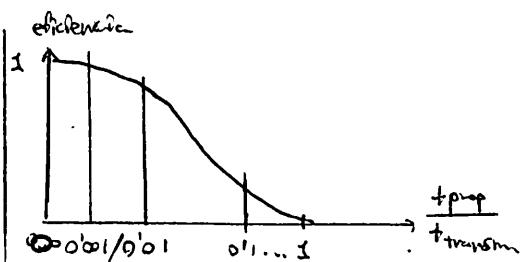
4) Para abortar, algoritmo de espera (exponential backoff):

- 1) $H \in \text{man}(n, 10)$
 - 2) $K \in \text{random}(0, \dots, 2^{m-1})$
 - 3) $t_{\text{espera}} = 512 \text{ bits} \cdot K = 512 \mu\text{s} \cdot K$

Objetivo: que los nodos que han colisionado no vuelvan a transmitir a la vez.

• Eficiencia

$$\text{eficiencia} = \frac{t_{\text{trans}}}{t_{\text{trans}} + 2t_{\text{prop}} \cdot c + t_{\text{prop}}} = \frac{1}{1 + 6^{1/4} \cdot \frac{t_{\text{prop}}}{t_{\text{trans}}}}$$



• Codificación Manchester

Transiciones:

+V ↓ → 1
-V

+V ↗ → 0
-V

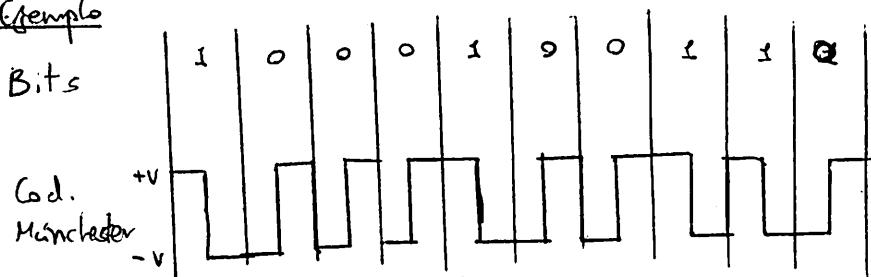
Permite sincronicidad
de roles.

Ejemplo

Bits

6d.
M. 5

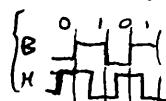
144



Ventajas: autosincro. en conversiones donde no se modula la sincronización. (Conversiones asincrónas)

Desventajas: consume el doble de ancho de banda de la señal de datos.

Detección de errores: si hay una ausencia de transición esperada



• Concentrator • Hub

Equivale a un bdc de datos para correr.
Muchas colisiones y poca seguridad.

• Switch

Switch Almacena y reenvia tramas Ethernet. 2) Reenvia por los enlaces la trama.

1) Torna tierra y hierba MZ

2) Recorta por los enlaces la trama.

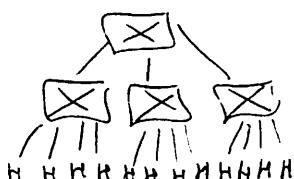
Use CSMA/CD

• Transparency

• **Transparencia**
• **Play-and-play:** aprende ^{por donde} como se resuelven conforme llegan tramas y solicitudes.

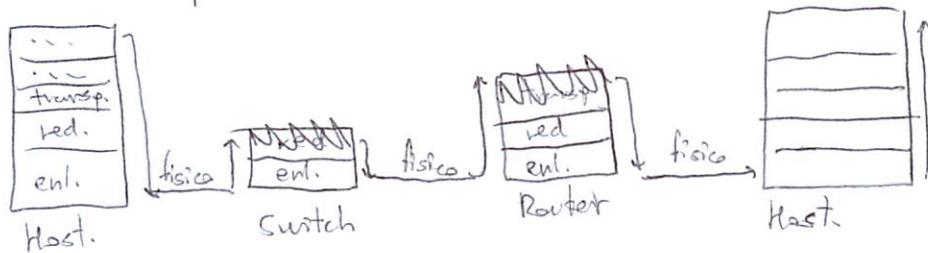
• Table switch: almacena {MAC, interfaz, TRL} para poder servir las tramas ethernet.

Puede realizarse en función de switches:



• Routers vs Switches

Routers - capa de red - Mantienen tablas de enrutado y emplean algoritmos de enrutamiento.
 Switches - capa de enlace - Mantienen tablas de switch y emplean algoritmos de autoaprendizaje.



• VLAN

Virtual Local Area Network.

Dentro de un switch podemos definir que bocas van a una VLAN y cuáles van a otra.

También podemos definirlo por MAC de host.

• PPP

Protocolo de enlace punto a punto

Emitor enlace. Receptor

Más simple { No requiere control de acceso al medio
 No requiere direcc. MAC explícita.

- Orientado a conexión; antes de enviar tramas, hay handshake y auth. de conexión.
- No hay control de flujo ni recuperación de errores.

Formato de la trama

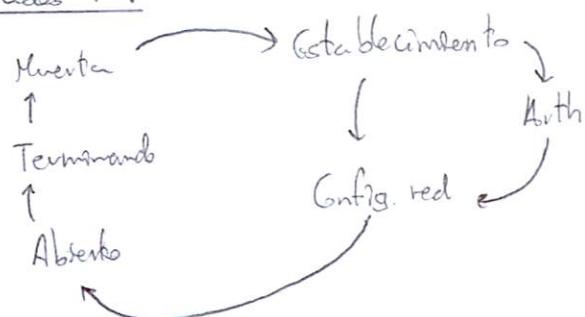
1 byte	1 byte	1 byte	1 o 2 bytes	2 o 4 bytes	1 byte
Flag	Dirección	Control	Protocolo	Info	Check

- Flag: indican inicio y fin de trama.
- Dirección: por lo general es fija. Siempre FF.
- Control: por lo general no usado.
- Protocolo: código protocolo copia su valor.
- Check: CRC.

Problema de Flag

¿Qué pasa si los datos tienen sec. bits = flag?
 Carácter de escape

Estados PPP



• Redes de Computadoras

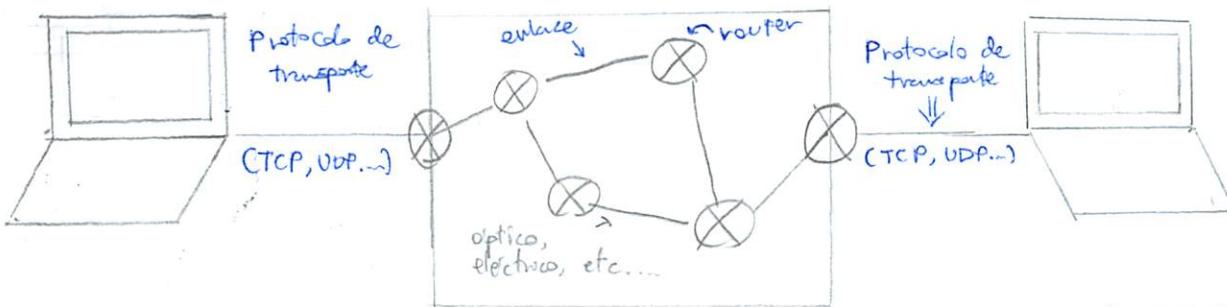
④ La capa de Red

Los segmentos (datagramas) pasan del emisor al receptor usando la capa de red.
Indica nodo origen y destino.

- Datagrama: segmento de datos + cabeceras. {

HRED	HTRANSF.	DATOS
------	----------	-------

 } Indica proceso destinatario.
- La misión de la capa de red es conseguir que los datagramas lleguen a su destino.



- Router (⊗): 2 funciones
 - Atravesar un inter es dar un salto. (Hop).
 - Reenvío: toma un datagrama y en función de los datos de la cabecera, lo envía por un enlace u otro.
- Enrutamiento:
 - Cálculo de rutas: optimiza y averigua la ruta que debe seguir el paquete. (Optimización ruta en tiempo variable).
 - Nota: se puede utilizar un server dedicado a calcular las rutas y pasárselas a los routers. } Red SDN.

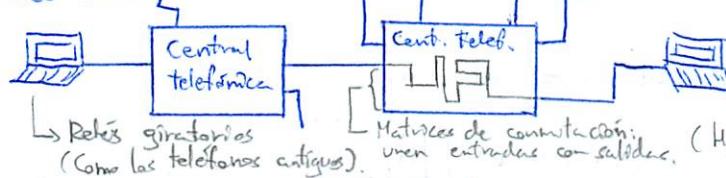
- Contafuegos: routers de la frontera con un sistema autónomo.
 - ↳ Deciden qué entra y qué no.
 - ↳ Dependen de la autoridad que gobierna sobre un conjunto de routers. } RIBIS.
- Existe cierta jerarquización de routers. (Países → Provincia → Barrio, etc.)

→ No es necesario que cada router tenga toda la información necesaria para enrutar, sólo la necesaria. (Se pierde la bala).

4. I.- Modelos de redes

• Conmutación de circuitos

- Más antiguo
- Es un circuito eléctrico que conecta emisor con receptor.



- Una vez establecido, es un circuito dedicado.
- Se establece físicamente un circuito de A a B.

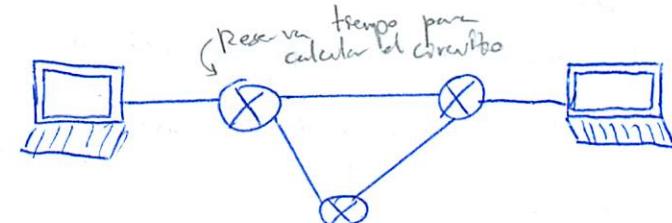
• Conmutación de circuitos virtuales

- Se debe establecer una conexión:

- ① A → D
- ② Se calcula un circuito y se le asigna una ID.
- ③ Cada paquete tiene una ID del circuito que sigue.
- ④ Cada x tiempo, se atende un paquete.

- No es la que usan en internet.

- La ruta se establece como combinación de interrupciones de entrada y salida.
- Puede ocasionar congestión.
Ej. Redes X.25.



Ventajas:

- Vel. cte.
- Retardo máx. garantizado
- Suministro de datos cte.

Interrupciones de entrada y salida

ID del circuito asignado por el router de entrada y salida

Desventajas:

- Ocupan recursos que igual no están utilizando.
(Rutas fijas.)
- Si cae el router, se debe establecer todas las conexiones.

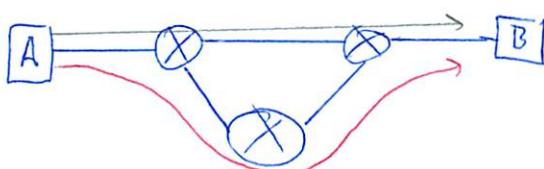
$$I_1, I_2, V_C, V_C$$

• Conmutación de datagramas:

- No hay establecimiento de conexión
- Cada datagrama se procesa de forma individual.

- No hay relación entre datagramas
- Los encamina en función de su destino.

} Dos datagramas con mismo destino no tienen por qué seguir el mismo recorrido.

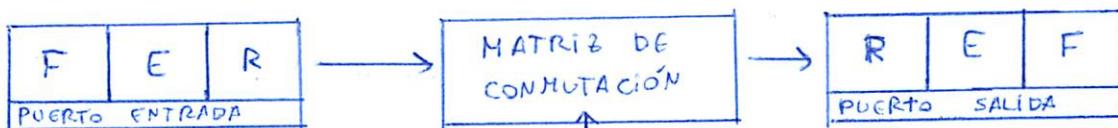


Ventajas

- Si un router se cae, "no pasa nada."
 - ↳ Se crea nueva ruta.
- No garantiza retardo, ni orden, ni no perdida.
- Pero si se fode se reconfigura (llega al destino correcto)
- Utilizan tablas de reenvío de datagramas.

• Destripando un router

Un router \otimes Ejecuta alg. de enrutamiento.
Reenvía datagramas desde puertos de entrada a puertos de salida.



F: convierte señal física en bits.

E: capa enlace: comprueba errores y lee cabecera

R: capa red: cola de datagramas. Se busca puerto de salida.

R: cola de datagramas de salida.

E: encapsula datagrama.

F: vierte información en el enlace físico.

Cada puerto tiene su propia memoria (buffer de datagramas) y procesador, por lo que las tramas se procesan muy rápido secuencialmente.

• Matriz de conmutación: envía datagramas del puerto de entrada al puerto de salida.

Tipos:

① Memoria del computador:

- 1) Llega el datagrama y se transfiere a un programa.
- 2) El programa del computador lo procese.
- 3) Lo saca a la salida.

} Comutación en memoria.
LENTO

② Comutación por bus:

Todas las entradas y salidas conectadas al bus.

Problema: secuenciamiento obligatorio. (de 1 en 1).

} Suficientemente rápido.
④ Usada.

③ Comutación por malla:

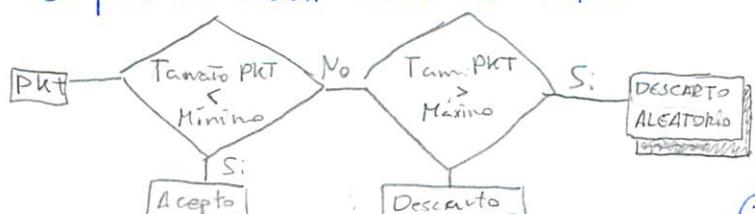
Mapa cartesiano comutador.

} La más rápida y
la más cara.

• Retardos y pérdidas de datagramas

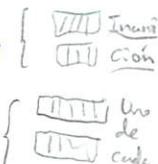
- Bloques HOL: un datagrama encolado en cabecera de la cola impide progresar el resto.
- El buffer del puerto de salida es la causa más frecuente de pérdidas de datagramas.
↳ Puede descender [viejos] → [nuevos].
- Soluciones de retrasos

• Control de congestión preventivo: (RED). Random Early Discard
↳ Se pone un umbral. Pasado ese umbral, se descarta aleat.



R

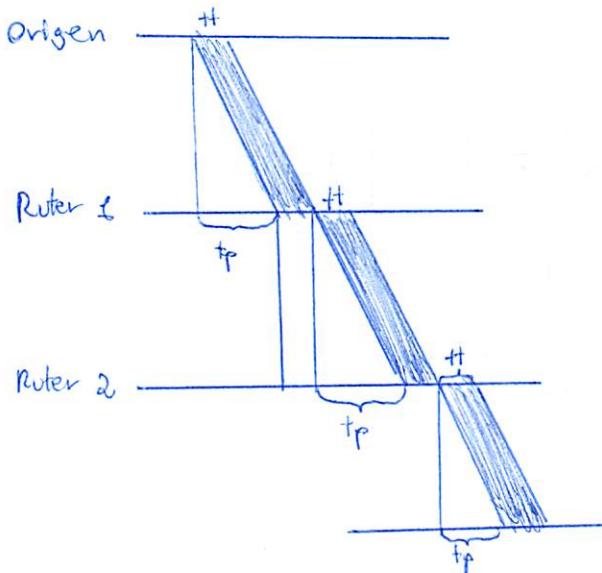
- Cola FIFO:
- Colas de prioridades
- Colas round robin:



(3)

• Reenvío del router (Almacenamiento y reenvío).

Un router no trabaja con un datagrama hasta que este no esté completo.



$$\begin{aligned}
 H &= \text{nº saltos} \\
 T &= t_{\text{procesamiento total}} \\
 K &= \text{tamaño búfer} \\
 T &= H(t_p + t_f) \\
 t_p + \frac{H}{3} + t_p + \frac{H}{3} + t_p + t_f \\
 H \cdot t_p + (H - L) \frac{H}{K} + t_f \\
 H \cdot t_p + H \cdot \frac{H}{K} + \frac{K-L}{K} \cdot H \\
 \text{Si } K \rightarrow \infty \Rightarrow H \cdot (t_p + t_f)
 \end{aligned}$$

Nota: suponemos que no hay tpo. procesamiento.

• Protocolo IP - Formato

Campos interesantes de la cabecera:

- ⇒ Versión: versión del protocolo. (IPv4, IPv6).
- ⇒ Long. cabecera: es variable.
- ⇒ Tipo - servicio: permite diferenciar entre distintos datagramas ip. No usado.
- ⇒ Despl. fragmentación: más adelante lo veremos.
- ⇒ Tpo. vida: nº de saltos restantes antes de morir. ($0 \rightarrow 255$). Para evitar bucles.
- ⇒ Protocolo capa superior: nº que identifica protocolo capa transporte. { 6 - TCP | 17 - UDP }
- ⇒ Checksum: ayuda a routers a detectar errores de bits.
Se ejecuta y recalcula en cada salto por el tpo. de vida. No desactivable.
- ⇒ Dir. IP (32 bits): origen y destino.
- ⇒ Opciones: permite ampliar la cabecera IP. No usado.

• Protocolo IP: gestión:

- Convenio de direccionamiento
- Formato de datagramas
- Convenios de manipulación de paquetes

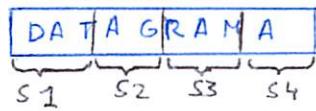
Fragmentación y reensamblado

Las tramas no pueden ser muy grandes, porque generarian errores.

o MTU: Bytes máximos de trama. Específico de router.

Si el tamaño de trama supera MTU, se fragmenta.

{ Errores, obligando a reenviar todo el datagrama de nuevo.
Iniciación en la red.



El header de estos nuevos fragmentos contendrá:

- Identificador: ID del datagrama original.
- Indicador:
 - I → no es el último.
 - } o último fragmento
- Desplazamiento indica el primer bit del fragmento respecto al datagrama dividido entre 8.
- Nueva longitud y checksum.

Ejemplo: datagrama de 4000 bytes. (3980 bytes de datos + 20 bytes header)

	Longitud	Indicador	Identificador	Desplazamiento
•	1500	I (hay más)	X	Ø
•	1500	I (hay más)	X	1480/8
•	1640	Ø (ultimo)	X	2960/8 Primer byte que lleva cada uno /8 El mismo de la cabecera original.

Consideraciones finales

Si los fragmentos se encuentran un router con MTU más pequeño, se fragmentan los fragmentos y vienen refragmentados.

Existe un bit de don't fragment: algunos routers no aceptan fragmentación.

Direccionalmiento IPv4

Interfaz: conexión entre la máquina - host o router - y el enlace físico.
Para identificar la interfaz de un router o host, se usa la dirección IP. (32 bits).

Ej. IP: 223.1.1.1 \Leftrightarrow 1101 1111 0000 0001 0000 0001 0000 0001
 223 1 1 1 1

Máscara de red: indica con 1's los bits que no podemos modificar de una red.

Ejemplo:

IP = 1010 1101. 0111 0101. 0101 0101. 1010 0101
 Máscara = 1111 1111. 1111 1111. 1111 1110. 0000 0000

Netmask
 $/23$
 1...1 \Rightarrow Parte de red.
 0...0 \Rightarrow Parte de host.

- Dirección de red: IP con bits de host a 0. (Ej. 192.168.1.0/24)
- Dirección de broadcast: IP con bits de host a 1. (Ej. 192.168.1.255/24).
 ↳ Envía paquetes a todos los hosts de la red.

Introducción: Arquitectura de clases

Clase	Rango	Máscara (Bits de red)	Bits de host
A	0.0.0.0 - 127.255.255.255	1111 0... /8	24
B	128.0.0.0 - 191.255.255.255	1111 10... /16	16
C	192.0.0.0 - 223.255.255.255	1111 110.. /24	8
D	224.0.0.0 - 239.255.255.255	1111 1110... /12	Direcciones multicast; usadas para crear grupos
E	240.0.0.0 - 255.255.255.255	1111 11110... /5	Investigación

Direcciones reservadas

0.0.0.0 \Rightarrow Designa algo irrealizable.
 255.255.255.255 \Rightarrow Broadcast global.
 127.x.x.x \Rightarrow Loopback (Propia máquina).

Direccionalamiento privado (hogares, instituciones...)

1 A \Rightarrow 10.0.0.0 \Rightarrow 10.255.255.255 (8R/24H)
 16 B \Rightarrow 176.16.0.0 \Rightarrow 176.32.255.255 (16R/16H)
 256 C \Rightarrow 192.168.0.0 \Rightarrow 192.168.255.255 (24R/8H)

Problema arg. clases:
 demasiado rígidas, no
 soportaba necesidades
 reales de usuarios.
 Solución: direccionalamiento
 sin clases: CIDR

• CIDR

En ruteamiento sin clúses: Red / Maíscaren. GJ

200.23.16.0 / 23
Red Museum

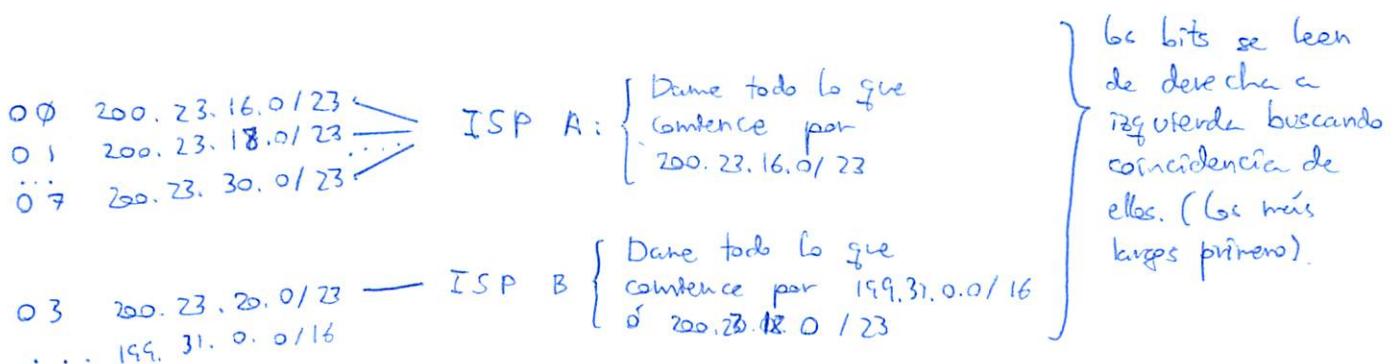
} las primeras 23 bits
 } son de red y no se
 } pueden modificar.
 Quedan $2^{32-23} - 2$
 } bits para hosts.

Sub red: red de una red. Interfaz de dispositivo con misma parte de dirección IP.

¿Cómo obtiene un ISP direcciones? Se lo compra al ICANN.

- Asigna direcciones
- Gestiona DNS
- Asigna dir. dominio

La red tiene un descentralamiento jerárquico



- DHCP: Dynamic Host Configuration Protocol.

Obtiene dinámicamente una dirección que proporciona un servidor cuando se conecta a la red.

Además, { - Permite renovar una IP previamente asignada.
- Permite reutilizar IPs.
- Configuración cómoda de host máquinas

Faces

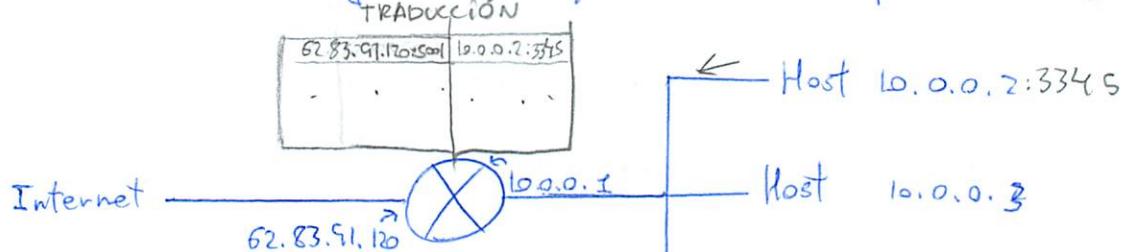
- Opcional:

 - ① Descubrimiento DHCP: Cliente ($0.0.0.68 \rightarrow 255.255.255.255$) $\xrightarrow{\text{DHCP}}$
 - ② Oferta DHCP: Servidor envía una posible IP por broadcast a cliente
 - ③ Solicitud DHCP: Cliente solicita tomar esa IP como propia. (por broadcast.)
 - ④ ACK DHCP: Servidor confirma a cliente la asignación de IP.
↳ También asigna gateway, netmask, y DNS's propias.

DHCP
UDP
IP

• Nat: Network Address Translation

Como nos quedamos sin direcciones IP, se reservan redes de carácter privado. NAT es el mecanismo que traduce y comunican redes públicas y privadas.



Todos los datagramas que salen de la red local tienen la misma IP de origen, asignada por el ISP. Tienen diferentes puertos.

La traducción se realiza con una tabla de traducción que tiene el router, utilizando los puertos como filtro.

- NAT transversal: técnica que permite descubrir la IP privada del host que ofrece un servicio.
P2P, Webcast...

• ICMP: Internet Control Message Protocol

Lo utilizan hosts y routers para enviarse info. de control de nivel de red. (No datos).

Viaja en un datagrama sin cabeceras en la capa de transporte.

Va dirigido al SCO del host.

Tipo	Cod. tipo	Checksum
Datos: puede contener la IP del router que ha localizado el error.		

- Traceroute: permite ver routers del camino y retraso. Se va incrementando el TTL en 1 paso a paso. Si en un router el TTL es 0, el router devuelve ICMP con info. de TTL caducado. Así sabe su IP.
 - Don't fragment: si un paquete es muy grande, un router puede enviar un ICMP avisando de que debe fragmentar.
 - Congestión: un router puede enviar un ICMP al emisor para avisar de que está congestionado. Puede ser contraproducente (gen. más congestión).
- Windows: ping
Linux: ping

IPv6

Createdo por el problema de agotamiento de direcciones IPv4. (32 bits).
 ↳ Además de complejidad en routers, tablas de enrutamiento enormes, etc..

Características

- Direcciones de 128 bits con máscaras 128 bits.
- Cabecera de tam. fijo (40 bytes) sin checksum.
 ↳ Extensiones en datos.
- ICMP revisado y IPsec por defecto. (ICMPv6)
- No soporta fragmentación en routers. Si hay que fragmentar, el router avisa al emisor para que lo envíe fragmentado y lo descarta.

Cabecera

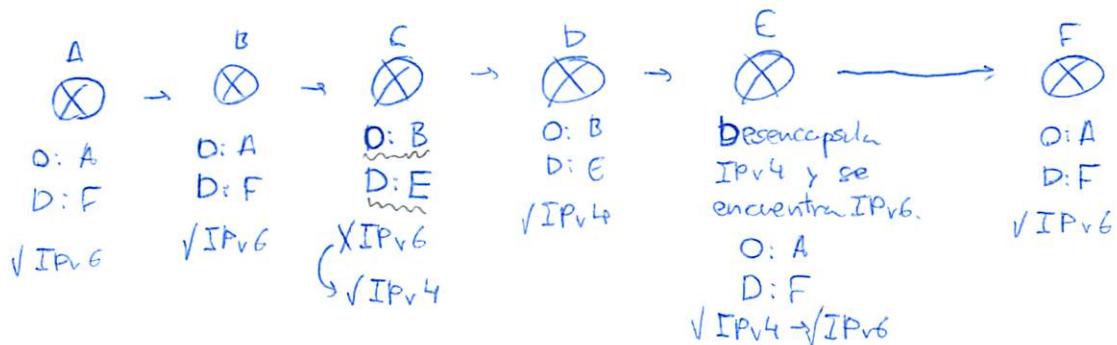
- Dirección origen y destino. (2 x 128 bits).
- Versión
- Prioridad
- Etiqueta flujo
- Long. carga: tamaño datos.
- Siguiente cabecera: código que indica el tipo de cabecera del siguiente nivel.
 También indican extensiones de cabecera.
- Lím. de saltos (TTL).

Formato dirección IPv6

Ejemplo: fe80:0:c9e5:b063:99da::1d33 % 9

Netmask?

- Extensión de fragmentación: guarda info sobre la fragmentación. (Despl., indicador, etc.).
- Transición IPv4 → IPv6: Si un nodo no entiende IPv6, lo encapsula en IPv4.



• Algoritmos de enrutamiento

Encuentran la ruta de menor coste.

2 tipos { Globales: todos los routers conocen costes y topología (Link-State)
Descentralizados: el router conoce vecinos y coste a ellos. (Vector-Distancia).

Además pueden ser { Estáticas: las rutas tardan en cambiar.
Dinámicas: las rutas cambian rápidamente.

• Algoritmo Link-State

Algoritmo global, calcula iterativamente la ruta de coste mínimo de un origen al resto.

$c(x, y) \rightarrow$ coste de enlace de x a y .

Problema de oscilación de congestión.

$D(v) \rightarrow$ valor actual de la ruta desde origen a v .

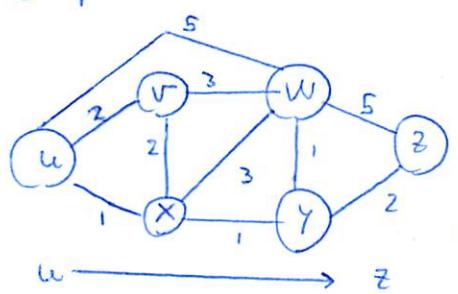
Complejidad $O(n^2)$

$P(v) \rightarrow$ nodo predecesor de origen en v .

$\hookrightarrow O(n \log n)$.

N : nodos de los que conocemos su camino de coste mínimo.

Ejemplo



Paso	v	w	x	y	z	Descripción
0	2 u	5 u	1 u	∞	∞	Todo x.
1	2 u	4 x	\times	2 x	∞	Todo y.
2	2 u	3 y	\times	\times	4 y	Todo w.
3	\times	3 y	\times	\times	4 y	Todo z.
4	\times	\times	\times	\times	4 y	Todo z.
5	\times	\times	\times	\times	\times	N!: u, x, y, v, w, z

• Algoritmo vector-distancia

Calcula todas las rutas y se queda con la mínima.

$$d_x(y) = \min_{v \in N} \{ c(x, v) + d_v(y) \}$$

El coste del camino mío de x a y es lo que cuesta llegar al vecino + lo que el vecino dice que le cuesta llegar a y .

Ejemplo para el grafo anterior

$$\begin{aligned} d_u(z) &= \min \{ c(u, v) + d_v(z), = 2 + 5 \\ &\quad c(u, x) + d_x(z), = 1 + 3 \\ &\quad c(u, w) + d_w(z) = 5 + 3 \} \end{aligned} \quad \left. \begin{array}{l} \min = 4 \Rightarrow \text{Lo envía a } x. \end{array} \right.$$

Cada nodo:

• Espera mensajes de vecinos

• Recalcula estimados asincrónicamente

• Si cambian costes, avisa a vecinos.

Problema: si aumentan los costes, puede ocurrir un desfase de información que crea un bucle de enrutamiento.

El paquete va rebotando poco a poco a la vez que se actualizan las tablas lentamente.

Enrutamiento jerárquico

Internet = Red de redes. Cada admin. controla su red.
Los routers están organizados en regiones: sistemas autónomos.

En cada AS { Se ejecuta el mismo protocolo de enrutamiento.
Hay un router gateway que conecta un AS con otro AS.
Los routers gateway usan BGP.

En cada router de un AS hay { Tabla enrut. INTRA-AS: tabla para enrutar dentro del AS.
Tabla enrut. INTER-AS: tabla para enrutar a otros AS.

Protocolo RIP

Protocolo de enrutamiento interno de un AS. Usa el algoritmo vector distancia.
Utiliza un máximo de 15 saltos.

Cada router tiene su tabla RIP de enrutamiento que contiene { vector de distancias
tabla de reenvío.

Cada 30 seg., se intercambian entre vecinos Mensajes de respuesta RIP

Sistema de máx. 25 subredes con distancia del emisor a ellas.

Si no se escucha un anuncio pasado 180 seg. { -El vecino se declara muerto.
-Se invalidan sus rutas.
-Se envían nuevos anuncios a vecinos.
El fallo se propaga por la red.

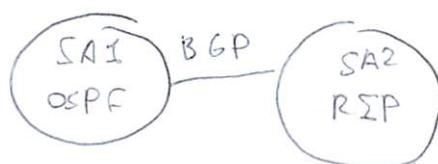
RIP viaja sobre UDP en port 520. Dirección multicast (Solo un grupo del total de hosts en red).

Comandos { -Hello: avisa que sigue vivo.
-Update: actualiza rutas.
-Request: pide rutas a vecinos.

1 versión: sin máscara de red variable.
2 versión: con máscara de red variable.

OSPF - Intra SA - Linkstate (Dif. Kafra)
RIP - Intra SA - Vector dist. (Mín. Recursivo)

BGP - Inter SA



• OSPF

Alg. público. Usa link state. (Dijkstra).

Los anuncios:

- Tienen 1 entrada / vecino.
- Se envían por todo el AS cada 30 minutos. + mensajes cortos de actualización.
- Usan IP

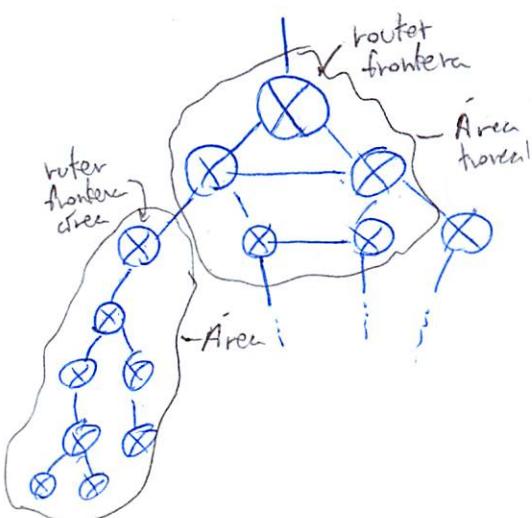
Características:

- 1) Es seguro: permite autenticación de datagramas.
- 2) Permite varias rutas de igual coste. (No como en RIP).
- 3) Métricas variadas y arcos orientados (coste ida ≠ coste vuelta).
- 4) Soporta multicdifusión.

Comandos:

- Hello! : msg. para decir que estás akt.
- Linkstate update : envía info.
- " " request : pide info.
- " " ACK : Confirma info
- Database description: metadatos y topología.

Jerarquía:



Jerarquía de 2 niveles

{ Área troncal
Área local

Cada nodo conoce la topología del su área, pero sólo el camino más corto al resto de áreas

R.Frontera: conectan a otros AS

R.Frontera de áreas: calculan distancias a redes en su área

BGP

Protocolo INTER AS. Permite:

Saber qué subredes se alcanzan por las AS vecinas.
Propagar esa info por los routers de su sist. autónomo.
Permitir a las subredes decir que existen y dónde se encuentran.

Funciona entre pares de nodos fronterizos por TCP. (puerto 179).

e BGP: routers externos.
i BGP: routers internos.

Formato de anuncios:

- Open: abre conexión.
- Update: actualiza conexión. 2 elementos
 - AS-PATH: lista de SAs que ha atravesado.
 - NEXT-HOP: indica el router atravesado.
- Keep alive
- Notification

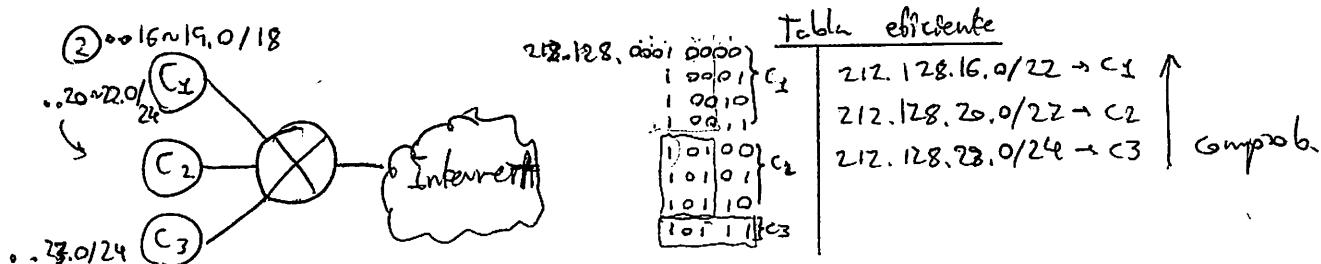
Ej. Redes - T4

① a) MTU Max. Transm. Unit. El máximo tamaño de datagrama que puede pasar por un enlace.

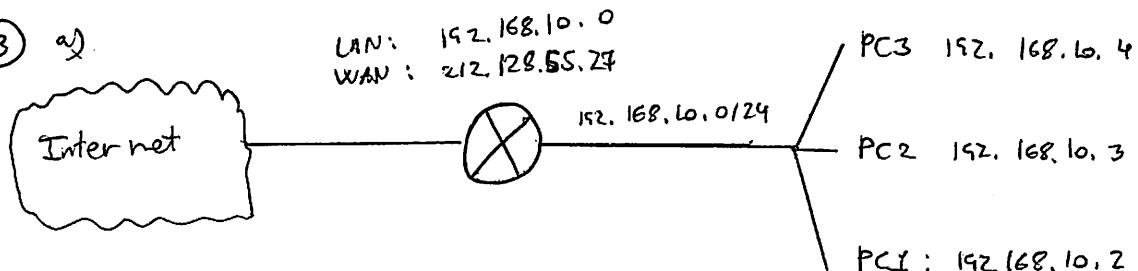
b) Si: MTU ↑ : ↑ prob. errores, mucho desvío y reenvío de PCTP; inefficiencia
 Si: MTU ↓ : mucho fragmentación y sobrecarga de procesamiento de datagramas. Ideal.
 Tamaño medio: ideal.

c) Se envía ICMP con bit de don't fragment y tamaño más grande posible.

A la que se encuentre un enlace con MTU < tamaño, el router enviará paquete ICMP diciendo que eso no pasa y se debe enviar otro datagrama con tamaño \leq MTU, y así en bucle.



③ a)



b) Tabla NAT

WAN

	IP Destino	IP Origen	Puerto D	Puerto O	IP best.	IP Origen	Puerto best.	Puerto Orig.
PC1	213.134.43.166	192.168.55.27	12500	80	213.134.43.166	192.168.10.2	80	3357
PC2	"	"	12501	"	"	192.168.10.3	"	3368
PC3	"	"	12502	"	"	192.168.10.4	"	3359
PC4	"	"	12503	"	"	"	"	"

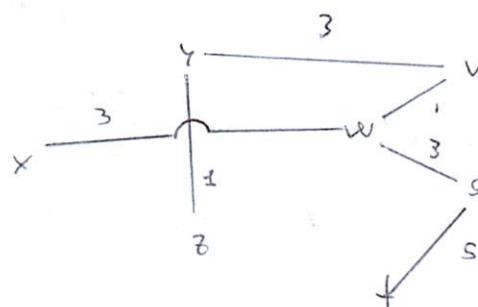
c) Que alguien se ha conectado por http a 192.168.10.1, y seguidamente esa misma app web envia datos por 3357, 3358. O que 2 personas se han conectado y se ha creado un nuevo socket para cada una de estas personas.

Paso	x	y	z	w	s	t	v	Nota
0	∞	∞	∞	s, 3	x	s	5	Tono w
1	w, 6	∞	∞	x	x	s	5	Tono v
2	w, 6	v, 7	∞	x	x	s	5	Tono f
3	w, 6	v, 7	t, 9	x	x	x	x	Tono x
4	x	v, 7	t, 9	x	x	x	x	Tono y
5	x	x	y, 8	x	x	x	x	Tono z

c) Tabla reversa de s
Para llegar a s
Se usa el enlace

x	s, w
y	s, w
z	s, w
w	s, w
t	s, t
v	s, w

b)



⑤ M bits por L equipos con almacenamiento y reenvío.

$$\text{Vel transmisión} = R$$

t_{prop.} cte.

M = bits datos

L = n° saltos

K = tam fragmento

h = cabecera

R = bits/seg.

a) Tamaño total bits a transmitir

$$\text{Bits} = \left\lceil \frac{M}{K} \right\rceil \cdot (K + h)$$

$\left\{ \begin{array}{l} \text{Nº frag.} = \text{tam.fragmentos} \\ \text{Redondeo} \\ \text{permiso} \end{array} \right.$

$$\text{b) } t_{\text{ps. total}} = \underbrace{(L \cdot t_p)}_{\text{ret. paquetes}} + \underbrace{\left(\frac{K + h}{R} \right) \cdot \left[(L - 1) + \frac{M}{K} \right]}_{\text{Lo que tarda en enviar 1 PKT}}$$

Nº paquetes que envió + n° saltos de rebote que duró - inicial

$$c) \text{Derivo y } K = \sqrt{\frac{M \cdot h}{L - 1}}$$

$$\text{Aplicéndolo } K \text{ para } \left\{ \begin{array}{l} M = 1 \text{ MB} \\ L = 5 \text{ saltos} \\ h = 64 \text{ B} \end{array} \right\} K = 4 \text{ kB.}$$

→ 1 hay más
Fijo último

⑥ Fragmentación PKT's.

h = cabecera ~ 20 bytes

MTU = 536 bytes

PKT = 1684 bytes. $\Sigma d = 7$;

Datos = $1684 - 20 = 1664$;

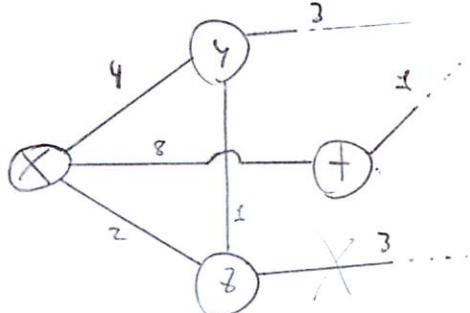
$$\text{Fragmentos} = \frac{\text{Datos}}{\text{MTU-Cabecera}} = \left\lceil \frac{1664}{516} \right\rceil = 4 \text{ PKT's.}$$

Long.	Id	Indicador	Desplaz.
536	7	1	0
536	7	1	516/8
536	7	1	1032/8
136	7	∅	1548/8

4

Ex. Redes T4 - P4 II

7)



$d_Y(u) = 3$	table X
$d_X(u) = 4$	
$d_Z(u) = 3$	$d_X(u) = 4$
$d_Z(u) = 2$	
$d_+(u) = 1$	$d_+(u) = \infty$
$d_+(u) = 8$	

La buena
→

Suponiendo ej.
de vector dist.
por Z.
en.

d) $d_Y(u)$ debería valer 5

$$\begin{array}{c} d_Y(u) \\ \sim \\ d_Z(u) \end{array} \quad \begin{array}{c} 1 \\ \sim \\ 6 \end{array}$$

$$a) d_X(u) = \min \left\{ \begin{array}{l} c(x, y) + d_Y(u), \\ c(x, z) + d_Z(u), \\ c(x, +) + d_+(u) \end{array} \right\} = \min \left\{ \begin{array}{l} 4+3, \\ 8+1, \\ 2+3 \end{array} \right\} = 5 = \text{Enrutado por } Z.$$

Registrarse que para llegar a u debe hacerlo por Z porque es lo más barato.

b) No, p.g. para Y, Z, + sigue siendo más barato enviar directamente a u.

$$c) d_Y(u) = \min \left\{ \begin{array}{l} c(x, y) + d_Y(u), \\ c(x, z) + d_Z(u), \\ c(x, +) + d_+(u) \end{array} \right\} = \min \left\{ \begin{array}{l} 4+3, \\ 2+5, \\ 8+1 \end{array} \right\}$$

Enrutado por Y.

Por otra parte, si Z pierde su conexión con u, se ejecutaría el algoritmo distanciar en Z. Z vería que su mejor opción es enrutar por Y, de forma que el coste sería 4.

De esta forma, $d_Z(u) = 4$, no ∞ .

De esta forma, X seguirá enrutiando por Z.

$$d_X(u) = 6 = c(x, z) + d_Z(u) =$$

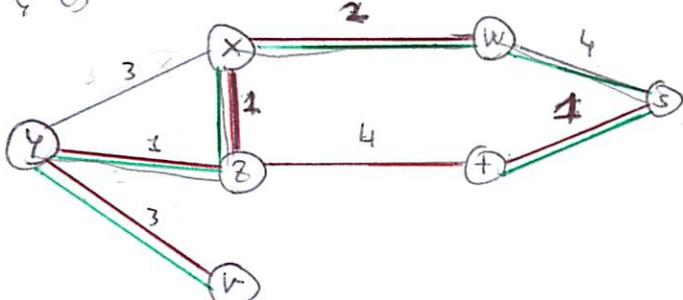
$$d_Z(u) = c(z, z) + c(z, y) + d_Y(u)$$

$$d_Z(u) = 2 + 1 + 3 = 6$$

planteo (solución)

$d_X(u)$ usando Z < $d_X(u)$ usando Y.

8) a) Y b) Nodo origen → Y.
c)



w → resto.

Tabla enrutamiento Y

Para llegar a	Se usa enlace	Tabla V
X	Y, Z	Y, X
Z	Y, Z	W, Y
+	Y, +	+
w	Y, Z	W, S
+	Y, Z	S
s	Y, Z	U, X
v	Y, Z	U, X

d) Paso	1	Y	X	w	+	s	z	Nota	V
0		∞	w 2	X		w 4	∞	Toma X	∞
1	X 5					w 4	8 3	Toma Z	∞
2	Z 4					w 4	X	Toma S	∞
3	Z 4	X					X	Toma Y	∞
4		X	X				X	Toma +	Y 6
5	X	X	X				X	Toma V	Y 6
6	X	X	X				X	Fini.	X

Ej. T6 Redes

- ① Distancia Hamming: ~~redes~~ n° bits que se deben cambiar para convertir una cadena de bits en otra.

Por ejemplo, la distancia Hamming para cambiar $011010 \xrightarrow{\text{3}} 001001$ es 3.
Por tanto, para que un mensaje erróneo pase inadvertido al receptor, deben cambiar H bits.

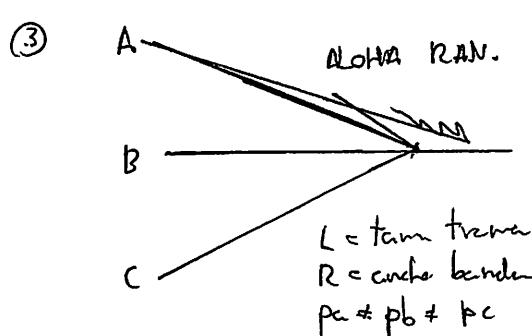
Sea p la prob. de que cambie un bit, la prob. de q. pase inadvertido será $p \times p \times p \dots H \text{ veces} = p^H$.

- ② a) DATOS: 101100011011011

CRC: 0010011

b) Ver círculos

- c) Para que ademas podamos comprobar el checkeo de paridad, se debe cumplir $\sum g(x) \rightarrow g(1) = 0$. En este caso, $g(1) = \underbrace{1}_{\text{un}} + \underbrace{1}_{\text{un}} + \underbrace{1}_{\text{un}} + \underbrace{1}_{\text{un}} + \underbrace{1}_{\text{un}} = 0 + 0 - 0 = 0$
Podemos ver el checkeo de paridad

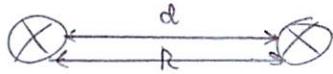


- a) $P_a \cdot (1-p_b) \cdot (1-p_c)$ } transmite a y no transmite b.
b) $P_a \cdot (1-p_b) \cdot (1-p_c) + P_b \cdot (1-p_a) \cdot (1-p_c) + P_c \cdot (1-p_a) \cdot (1-p_b)$ } que transmite alguno
y los demás no.
c) Tasa med. transm = $\frac{R}{t_m +}$

$$r = \frac{e}{f} ; f = \frac{e}{r}$$

1

- ④ R = vel transmisió
 $t_{prop} = t_{prop. propagació}$
 $t_{trans} = t_{trans. transmisió}$
 $d = \text{distància}$
 $v = \text{vel. prop.}$
 $L = \text{long. transm.}$



$$\left. \begin{array}{l} a) t_{prop} = \frac{d}{v} \\ t_{trans} = \frac{L}{R} \end{array} \right\} \eta = \frac{I}{1 + 2e \cdot \left(\frac{\frac{d}{v}}{\frac{L}{R}} \right)} = \frac{I}{1 + 2e \cdot \left(\frac{d \cdot R}{v \cdot L} \right)}$$

$$\eta = \text{ANAL} \frac{I}{1 + \frac{2e \cdot d \cdot R}{v \cdot L}} ;$$

$$\cancel{\frac{2e \cdot d \cdot R}{v \cdot L}} = \frac{I}{\eta} - I ; \left[\frac{I}{\eta} - I \right] \cdot v = \frac{2e \cdot d \cdot R}{L} ;$$

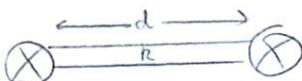
$$a) L = \frac{2 \cdot e \cdot d \cdot R}{\left[\frac{I}{\eta_0} - I \right] \cdot v}$$

$$b) d = \frac{L \cdot v \cdot \left[\frac{I}{\eta} - I \right]}{2 \cdot e \cdot R}$$

~~PERÒ?~~ BIEN?

$$\left| \begin{array}{l} R = 100 \cdot 10^6 \\ v = 2 \cdot 10^8 \\ L = 1500 \\ \eta = 0'7 \end{array} \right| \frac{1500 \cdot 2 \cdot 10^8 \left[\frac{I}{0'7} - I \right]}{2 \cdot e \cdot 100 \cdot 10^6} \Rightarrow d \geq d \leq \cancel{2 \cdot 10^8 \cdot 10^6} \\ d \leq 276'4939 \dots$$

- ⑤ R = vel transmisió
 t_{prop}, t_{trans}
 $d = \text{distància nodes}$
 $v = \text{vel. prop.}$
 $L = \text{long. transm.}$



$$a) \eta = \frac{I}{1 + 2e \cdot d/v} ;$$

$$\frac{1 + 2e \cdot d/v}{L/R} \cdot \eta = I ; I + (I + 2e) \cdot d/v \cdot R = L$$



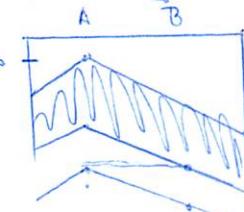
- ⑤ $\leftarrow d \rightarrow$
(A) — o — (B)
l bits.

$$a) \text{Deben esperar } t_{prop.} + t_{trans.} + t_{prop} = 2t_p + t_{trans}$$

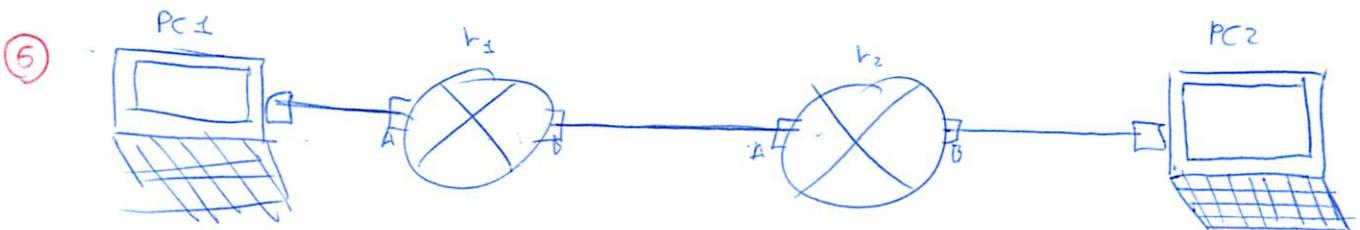
$$b) \text{Sólo un tpo. propagació.} = \frac{d}{v} \quad \cancel{\frac{2d}{v} + \frac{L}{R}}$$

~~¿ Hasta que detecta que A ya no emite más señal + 2tp.~~

~~Lo que debe esperar antes de sondar es la duración previsible de una trama, es decir, un tpo propagació.~~



$$c) \text{Eficiència de CSMA/CD: } \frac{t_{trans}}{t_{trans} + 2 \cdot t_{prop} \cdot e + 2t_p} = \frac{I}{1 + 6'44 \cdot \frac{t_{prop}}{t_{trans}}} = \frac{I}{1 + 6'44 \cdot \frac{\frac{d}{v}}{\frac{L}{R}}} = \frac{I}{1 + 6'44 \cdot \frac{d}{v \cdot L}}$$



- ① PC1 sabe que su IP es 212.100.1.10, y su gateway es 212.100.1.1. ~~PC1~~
PC1 sabe que tiene que enviar a PC2, cuya IP es 159.23.1.10.
PC1 No sabe ninguna MAC, excepto la suya.
- ② PC1 envía trama ARP con IP Origen PC1, IP Destino R1, MAC origen PC1, MAC destino FFFF-FFFF-FFFF
- ③ R1 responde a PC1 diciendo que su MAC es A1:A1:A1..:A1
- ④ Como PC1 ya conoce la MAC de su default gateway y no que su IP destino no está en su propia red, envía la trama con datos a R2:

IP origen: PC1

IP Destin.: R2

MAC Orig.: PC1

MAC Dest.: R2

- ⑤ R1 ve que no sabe llegar a ese IP, y hace broadcast preguntando quien lo tiene:

IP Origen: R1

IP Dest.: PC2

MAC Or.: R1

MAC Dest.: Broadcast

| Adelante R1 guarda en ARP < PC1 - MAC PC1 - TTL >

- ⑥ R2 recibe broadcast, ve que el destino está en su propia red, y hace broadcast:

IP Origens: R2

IP Dest.: PC2

MAC Orig.: R2

MAC Dest.: Broadcast

- ⑦ PC2 contesta dando su MAC a R2. R2 guarda PC2 en su tabla ARP. < PC2 - MAC PC2 - TTL >

- ⑧ R2 contesta a R1 diciendo que sabe llegar a ese IP.

- ⑨ R1 guarda PC2 en su tabla ARP. < PC2 - MAC R2 - TTL >

- ⑩ R1 envía Pkt de PC1 a R2. R2 guarda en ARP < PC1 - MAC R1 - TTL >

Origen PC1

Dest.: PC2

MAC Or.: MAC R1

MAC Dst.: MAC R2

- ⑪ R2 envía Pkt de PC1 a PC2. PC2 guarda en ARP < PC1 - MAC R2 - TTL >

Origen: PC1

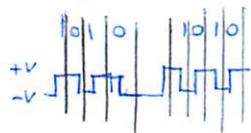
Dest.: PC2

MAC Or.: R2

MAC Dest.: PC2

⑦ a) En conexiones donde sea necesario sincronizar. Por ejemplo, en cuestiones de radio.

b) Aspecto.



⑧ True Star colisiones:

a) $m \in \{0, 1, 2, \dots, 15\}$; $m = 5$

$$k = \text{random}\{0 \dots 2^4\} = \{0, 1, 2, \dots, 16\}$$

$$\text{Prob. } k=7; \frac{1}{16}$$

b) $R = 100 \text{ Mb/s};$

$$\text{espera} = \frac{R}{P}$$



Universidad
de Alcalá

Redes de computadores

Problemas propuestos

Raúl Durán Díaz
Departamento de Automática
Universidad de Alcalá

ALCALÁ DE HENARES, ESPAÑA

Raúl Durán Díaz
Departamento de Automática
Universidad de Alcalá
E-28871 Alcalá de Henares, España
raul.duran@uah.es

El autor, consciente de la debilidad del pensamiento humano, ruega y agradece al piadoso lector que le comunique cualquier error que pueda encontrar en las siguientes páginas.

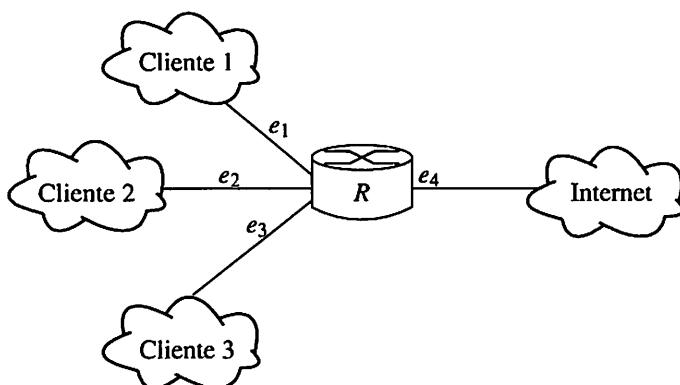
El presente documento es evolutivo. Los campos de revisión y fecha que aparecen a continuación indicarán cuál es la versión más moderna.

Revisión: 1.4
Fecha: 23 de marzo de 2018

Capítulo 4

La capa de red

1. Recordemos que en la cabecera de IPv4 existen dos *flags*, a saber, el flag MF (*more fragments*) y el DF (*don't fragment*). Contestar a las siguientes cuestiones.
 - a) Explicar qué significa el término MTU (*maximum transmission unit*), por qué existe, y qué es la MTU de una ruta.
 - b) Explicar qué influencia puede tener el valor de la MTU en el rendimiento de una red.
 - c) Explicar cómo se puede implementar un protocolo que permita a un nodo emisor de un datagrama averiguar la MTU de la ruta hasta el receptor de dicho datagrama. Suponer que se pueden generar las cabeceras a voluntad y están disponibles cualesquiera de los protocolos existentes a nivel de capa de red.
2. Un proveedor de servicios de internet (ISP) tiene tres clientes corporativos, todos ellos conectados al mismo *router R*, administrado por el ISP, según se ve en la figura. El ISP realiza la siguiente asignación de redes a cada cliente:



- Al primero le asigna las redes:
212.128.16.0/24, 212.128.17.0/24, 212.128.18.0/24,
212.128.19.0/24.
- Al segundo le asigna las redes:
212.128.20.0/24, 212.128.21.0/24, 212.128.22.0/24.

- Por fin, al tercero le asigna la red:
212.128.23.0/24.

Rellenar las entradas de la tabla enrutamiento de R de manera que los datagramas se enruten correctamente y se minimice el número de entradas de dicha tabla.

3. Supongamos un *router* NAT con una dirección IP de 212.128.55.27 en el lado WAN. El lado LAN está conectado a una red con prefijo 192.168.10.0/24 a la que también hay conectados tres equipos terminales con una interfaz de red cada uno.

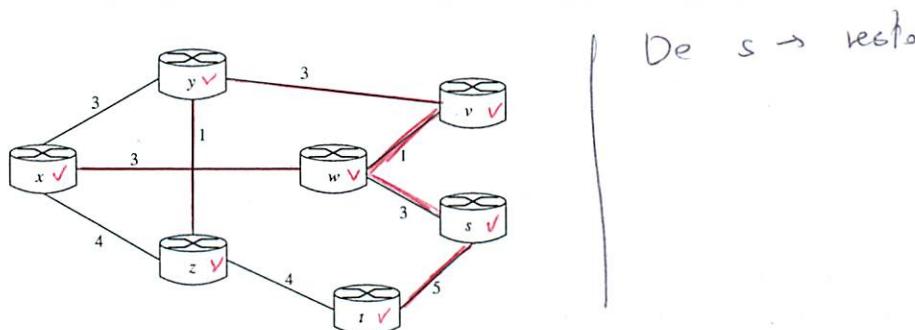
- Dar direcciones IP congruentes a todas las interfaces de la LAN.
- Supongamos que cada equipo terminal ha establecido una conexión HTTP con el servidor web www.uc3m.es, cuya dirección IP es 213.134.43.166. Se pide dar valores congruentes para las direcciones IP origen y destino, y puerto TCP origen y destino tanto en el lado LAN como en el lado WAN para tres parejas de datagramas, donde cada pareja va asociada a cada una de esas conexiones.
- Más tarde nos encontramos en la tabla NAT del *router* unas entradas como las siguientes:

Lado WAN	Lado LAN
212.128.55.27, 5080	192.168.10.10, 3357
212.128.55.27, 5081	192.168.10.10, 3358
212.128.55.27, 5082	192.168.10.11, 5439
212.128.55.27, 5050	192.168.10.10, 80

orig. pto orig. pto

A la vista de esas entradas, dar una posible interpretación de lo que está pasando.

4. Consideraremos la red de la figura. Utilizando el algoritmo de Dijkstra, se pide:
 - Calcular las rutas de coste mínimo y sus costes desde el nodo s a todos los nodos de la red.
 - Dibujar el árbol de rutas de coste mínimo desde el nodo s .
 - Escribir la tabla de reenvío resultante para el nodo s .



5. Se necesita transmitir un mensaje de tamaño M bits a lo largo de una ruta que ha de atravesar L equipos (por ejemplo, en la figura se tiene $L = 3$), troceado en paquetes que contienen k bits de datos y h bits de cabecera (un número fijoado) cada uno (supóngase que $M \gg h + k$). El modo de transmitir es *almacenamiento y reenvío*, es decir, cada equipo recibe todo un paquete antes de empezar a transmitirlo al siguiente equipo. Además, cada equipo puede estar emitiendo y recibiendo a la vez. La velocidad de transmisión de todos los enlaces es la misma e igual a R bits por segundo. El tiempo de propagación entre cada equipo es constante y de valor t_p . El tiempo de procesamiento es despreciable.
- ¿Cuál será el número total de bits que se necesita transmitir?
 - ¿Cuál es el retardo total, es decir, el tiempo necesario para conseguir la transmisión de todo el mensaje, en función de los parámetros?
 - ¿Cuál es el valor de k que minimiza ese retardo total? Aplicarlo al caso particular en que $M = 1$ MB, $L = 5$, $h = 64$ bytes, $t_p = 10^{-3}$ s y $R = 1$ Gb/s. ¿Qué velocidad de transmisión efectiva obtenemos?



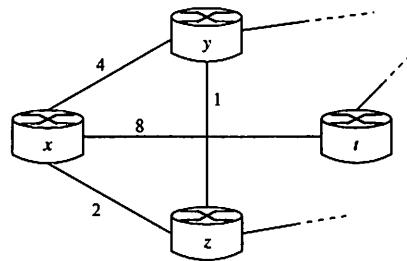
6. A lo largo de todo el problema consideramos que la cabecera IP tiene una longitud fija de 20 bytes. En estas condiciones, supongamos que un datagrama IP de 1684 bytes de longitud llega a un enlace cuya MTU es tan solo de 536 bytes.

Describir los campos *longitud de datagrama*, *identificador*, *indicador* y *desplazamiento* de todos y cada uno de los fragmentos que el router ha de generar para introducirlos en el enlace.

Nota: Se pueden elegir valores aleatorios cuando se vea necesario.

7. Consideremos la red de la figura, con los costes de enlace indicados. Supongamos que se usa el algoritmo de vector distancia para calcular las tablas de reenvío. Los nodos y , z , t han calculado sus vectores de mínima distancia a un cierto nodo u (no visible en la figura) y el resultado es que $d_y(u) = 3$, $d_z(u) = 3$, $d_t(u) = 1$. En ese mismo instante, x tan solo conoce las distancias a sus vecinos (expresadas en la figura) y está inicialmente a distancia infinita de u .
- Cuando los nodos vecinos de x le entreguen sus vectores de distancia mínima, ¿qué distancia mínima calculará x para ir a u ? ¿Qué entrada registrará en su tabla de reenvío?
 - Cuando x calcule su distancia mínima a u y la difunda a sus vecinos, ¿cambiarán estos sus respectivas distancias a u ? Justificar la respuesta.
 - Si de repente z se desconecta de u (suponemos que x , w no enrutan a través de z y por tanto no se ven afectados), de modo que $d_z(u) = \infty$, ¿cuál será la nueva distancia mínima de x a u ? ¿Cambia la tabla de reenvío de x ? ¿Qué le ocurrirá al nodo z ?

- d) ¿Qué valores debieran tener $d_y(u)$, $d_z(u)$ y $d_t(u)$ para que la ruta desde x hasta u tuviera el mismo coste enrutando a través de cualquiera de los vecinos y este coste fuera el mínimo? Suponer que $d_y(u)$, $d_z(u)$ y $d_t(u)$ han de tener como mínimo el valor 1.



8. En una determinada red se ha ejecutado el algoritmo de Dijkstra, obteniéndose la tabla que se ve más abajo. Basándose en ella, se pide

- Reproducir en un dibujo todo lo que se pueda deducir acerca de la estructura de esa red, anotando también los costes de cada enlace que puedan inferirse de la tabla.
- ¿Cuál es el nodo origen de las rutas? Proporcionar la tabla de enrutamiento partiendo de dicho nodo origen.
- Dibujar el árbol de rutas de coste mínimo a los demás nodos, indicando el coste de cada enlace.
- Calcular ahora mediante el algoritmo de Dijkstra las rutas mínimas en esa misma red pero tomando como nodo origen el w . Se pide también para este caso, el árbol de coste mínimo y la tabla de enrutamiento.

Paso	valor N'	$D(x), p(x)$	$D(z), p(z)$	$D(w), p(w)$	$D(v), p(v)$	$D(s), p(s)$	$D(t), p(t)$
0	y	3, y	1, y	—	3, y	—	—
1	yz	2, z		—	3, y	—	5, z
2	yzx			4, x	3, y	—	5, z
3	yzxv			4, x		—	5, z
4	yzxvw					8, w	5, z
5	yzxvw					6, t	
6	yzxvwts						

Capítulo 5

La capa de enlace

1. Sea una codificación tal que la distancia de Hamming entre sus códigos es H . Supongamos que tal codificación se usa en un enlace y que en ese enlace se produce un error en un bit con probabilidad p . ¿Cuál es la probabilidad de que se transmita un código erróneo y pase inadvertido al receptor? Justificar adecuadamente la respuesta.
$$R = p^H$$
2. Un enlace punto a punto emplea el código de redundancia cíclica (CRC) para detectar errores, con un polinomio generador $g(x) = x^6 + x^5 + x^3 + x^2 + x + 1$. El receptor ha recibido una trama que lleva los datos junto con el CRC concatenado al final (es decir, son los bits de menor peso). Dicha trama resulta ser: $(1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0, 1, 1|0, 0, 1, 0, 0, 1, 1)$. Contestar las siguientes cuestiones.
a) ¿Cuáles son los bits que corresponden a los datos y cuáles los que corresponden al CRC?
b) Comprobar si la trama ha llegado bien o si ha habido algún error, justificando la respuesta.
c) Justificar si el CRC así definido proporciona o no el chequeo de paridad par.
3. En una red de área local con medio compartido se utiliza el protocolo ALOHA ranurado. A dicha red se encuentran conectados tres nodos, A , B y C . Las tramas que todos transmiten son de longitud fija, L , y el canal tiene un ancho de banda R . Supongamos que las probabilidades de transmisión respectivas de cada nodo son p_A , p_B y p_C , distintas en general.
 - a) ¿Cuál será la probabilidad de que A consiga transmitir una trama con éxito?
 - b) ¿Cuál será la probabilidad de que cualquier nodo transmita una trama con éxito?
 - c) Si $p_A = 20\%$, $p_B = p_C = 30\%$, $L = 1500$ bytes, $R = 100$ Mb/s, cuál será la tasa media de transmisión de todo el sistema?
 - d) ¿Qué probabilidad de transmisión tendría que tener A para que su tasa media de transmisión fuera el doble que la de B y C , supuestas estas igualdades?

$$\frac{12}{7 \cdot 3} = 3 - 1; \frac{12}{(3-1)} = 2 \cdot 3; \frac{\frac{12}{3}}{(3-1)} = 2; \\ \frac{1}{\eta} = \frac{2 \cdot R_{red}}{v \cdot L} + 1; \frac{2 \cdot R_{red}}{v \cdot L} = \frac{1}{\eta} - 1; \left(\frac{2 \cdot R_{red}}{v \cdot L} \right) = v \cdot L$$

LA CAPA DE ENLACE

$$I + \frac{\frac{2 \cdot d}{v}}{L} = \frac{2 \cdot R_{red}}{v \cdot L} + 1 \quad \text{les? Obtener una expresión general y aplicarlo después al caso en que } p_B = p_C = 30\%.$$

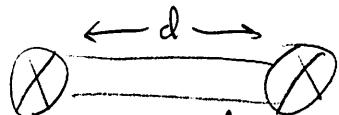
$$L = \frac{(2 \cdot R_{red})}{\left(\frac{1}{\eta} - 1\right)}$$

4. Supongamos una red Ethernet tipo cable coaxial con velocidad de transmisión R . Para ese tipo de red modelamos la eficiencia, η , como

$$I + \frac{2 \cdot e \cdot d/v}{L/R} = \frac{1}{\eta}; \eta = \frac{1}{1 + \frac{2e \cdot t_{prop}}{t_{trans}}} \quad \eta = \frac{1}{1 + 2e^{\frac{t_{prop}}{t_{trans}}}}, \eta = \frac{1}{1 + \frac{2e^{\frac{d}{v}}}{R}} \quad L = \frac{2 \cdot R_{red}}{v \cdot \left(\frac{1}{\eta} - 1\right)}$$

siendo e la base de los logaritmos neperianos, t_{prop} el tiempo de propagación de la señal y t_{trans} el tiempo de transmisión.

Sean ahora dos nodos conectados a esa red y separados por una distancia d . Si la velocidad de propagación de la señal en el cable es v , se pide



$$t_{prop} \Rightarrow \frac{d}{v} \\ t_{trans} \Rightarrow \frac{L}{R}$$

a)

$$d = \frac{L}{R} \cdot t_{trans}$$

$$\eta = \frac{1}{1 + 2e \cdot \left(\frac{d/v}{L/R}\right)};$$

$$\eta = \frac{1}{1 + 2e \left(\frac{d \cdot R}{L \cdot v}\right)};$$

$$\eta = \frac{e}{1 + \frac{2}{e}}$$

Pasar L a t_{trans} .

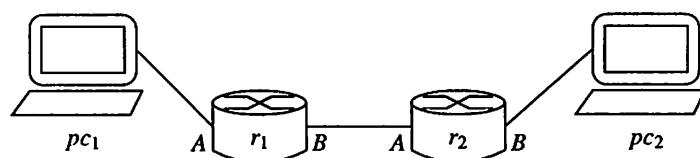
5. Sea un medio compartido en el que están presentes solamente dos nodos, A y B , separados una distancia d . La velocidad de propagación de la señal en ese medio es v . Cada nodo emite a una velocidad de transmisión R . La longitud de las tramas es igual a L .

Se utiliza el protocolo puro CSMA/CD donde cada nodo sondea el canal antes de transmitir y, si está libre, empieza la transmisión de manera inmediata. Si durante la transmisión detecta una colisión, interrumpe instantáneamente la transmisión.

En estas condiciones, se pide:

- Si el nodo A empieza a transmitir en el instante t_0 , ¿durante qué periodo de tiempo debiera B estar en silencio para que la transmisión se termine sin colisiones?
- Si B detecta señal en el canal antes de transmitir, ¿cuánto tiempo debería esperar para tener cierta garantía de encontrar el canal libre?
- Suponiendo un perfecto acuerdo en las transmisiones de A y de B , de forma que no se produzcan colisiones, ¿cuál sería la eficiencia del canal, es decir, qué porcentaje de tiempo está realizando un trabajo de transmisión útil? Expresarlo en función de d , v , L y R .

6. Considere la red Ethernet de la figura en donde los routers están correctamente configurados para enrutar paquetes desde pc_1 a pc_2 y viceversa:



Los datos para cada interfaz de red son:

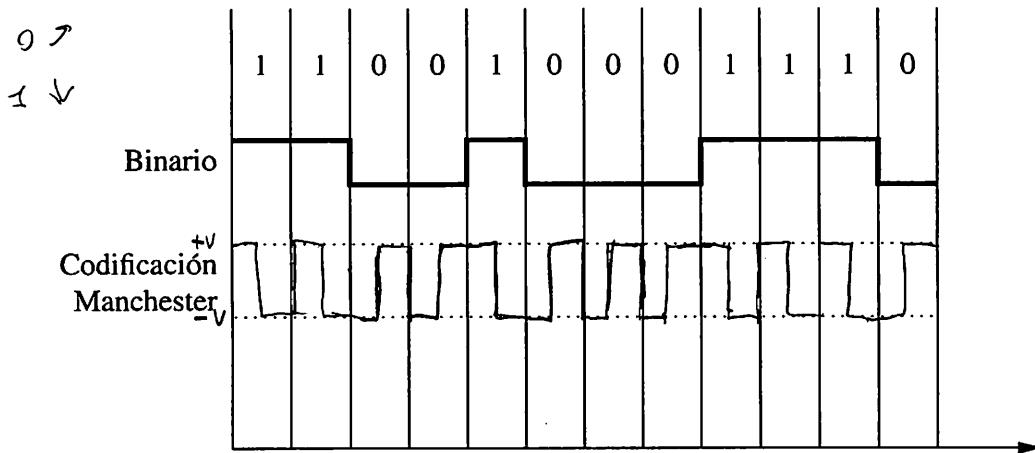
Equipo	Interfaz	Dirección IP	Dirección MAC
pc_1		212.100.1.10	C1:C1:C1:C1:C1:C1
pc_2		159.23.1.10	C2:C2:C2:C2:C2:C2
r_1	A	212.100.1.1	A1:A1:A1:A1:A1:A1
r_1	B	12.10.1.1	B1:B1:B1:B1:B1:B1
r_2	A	12.10.1.2	A2:A2:A2:A2:A2:A2
r_2	B	159.23.1.1	B2:B2:B2:B2:B2:B2

Inicialmente supondremos que las tablas ARP de todos los equipos están vacías. En estas condiciones pc_1 quiere enviar un segmento TCP a pc_2 .

Se pide dibujar cronológicamente todo el tráfico de tramas que van a circular por cada uno de los enlaces presentes en la figura. Para las tramas que encapsulen datagramas IP, se pide dar los valores de los campos *dirección IP origen*, *dirección IP destino* correspondientes a la cabecera IP y los campos *MAC origen*, *MAC destino* correspondientes a la cabecera Ethernet.

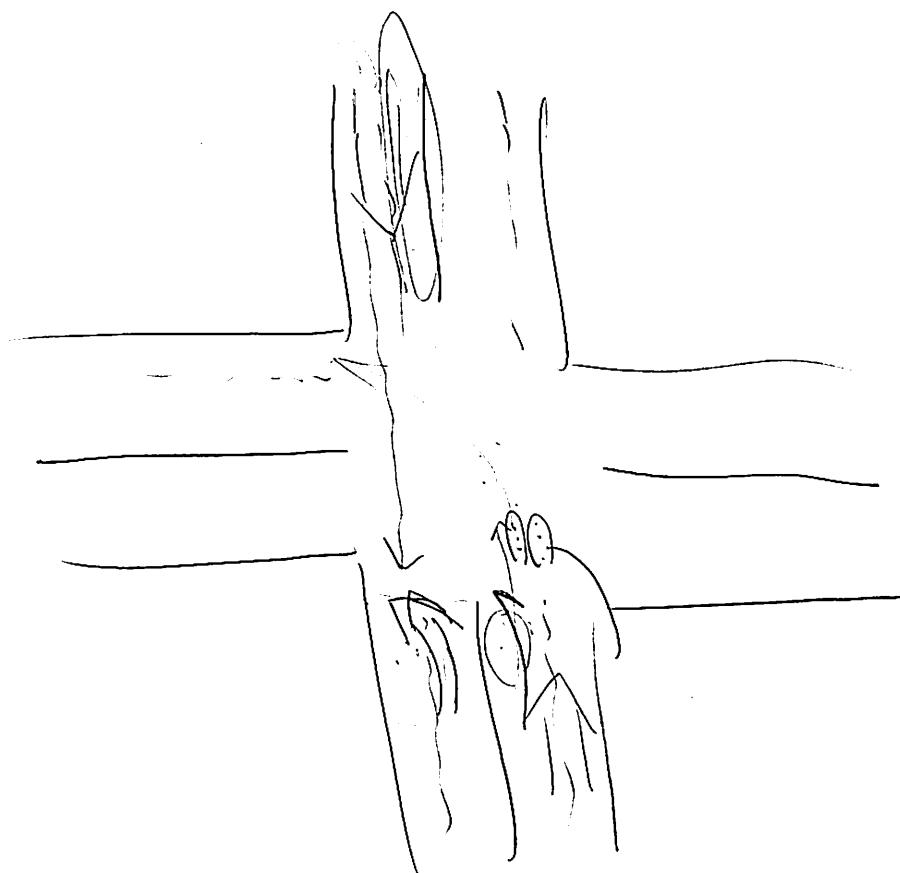
Para las tramas que encapsulen ARP, se piden los campos *MAC origen*, *MAC destino* de la cabecera Ethernet y los valores de *dirección fuente hardware*, *dirección fuente del protocolo IP*, *dirección destino hardware*, *dirección destino del protocolo IP*, encapsulados en esa trama.

7. Obtenga la codificación Manchester de la siguiente señal binaria dibujándola sobre el diagrama.



- a) Indique alguna razón que, a su juicio, haga aconsejable este tipo de codificación. En transmisor y receptor tiene que sincronizarse
- b) Sabemos que el preámbulo de una trama Ethernet está constituido por una ráfaga de 7 bytes, cada uno de ellos con esta estructura: (1, 0, 1, 0, 1, 0, 1, 0). ¿Qué aspecto tendrá la señal codificada en codificación Manchester correspondiente a tal preámbulo? ¿Qué frecuencia base tendrá? en relación con la velocidad de transmisión en bits por segundo?
8. Supongamos una red de área local que ejecuta el protocolo MAC propio de Ethernet. Contestar las siguientes cuestiones.

- a) Tras la quinta colisión, ¿cuál es la probabilidad de que un nodo espere un número $k = 7$ de periodos?
- b) Si la velocidad de la red es 100 Mb/s, ¿a cuántos segundos equivale una espera de $k = 7$ periodos?



Bibliografía

Básica

- James F. Kurose, Keith W. Ross. *Redes de computadoras: un enfoque descendente.* 7^a edición. Pearson Educación, Madrid, 2017.

Complementaria

- William Stallings. *Comunicaciones y Redes de Computadores.* 7^a edición traducida. Prentice Hall, 2004.
- Andrew S. Tanenbaum. *Redes de computadoras.* 4^a edición traducida. Prentice Hall, 2003.
- Alberto Leon-Garcia, Indra Widjaja. *Redes de comunicación, conceptos fundamentales y arquitecturas básicas.* McGraw-Hill, 2002.
- Dimitri P. Bertsekas, Robert G. Gallager. *Data Networks.* Second edition. Prentice Hall, 1992.
- F. Halsall. *Redes de computadoras e Internet.* 5^a edición traducida. Pearson Educación, 2006.
- Behrouz A. Forouzan. *Transmisión de datos y redes de comunicaciones.* 4^a edición traducida. McGraw-Hill, Madrid, 2007.
- W. Richard Stevens. *TCP/IP Illustrated, Volume 1: The Protocols.* First edition. Addison-Wesley, 1994.
- Francisco Manuel Márquez García. *UNIX programación avanzada.* 3^a edición. Ra-Ma, 2004.
- Paul Deitel, Harvey M. Deitel. *C: how to program.* Sixth edition. Prentice Hall, 2009.
- Bruce Eckel. *Thinking in Java.* Third edition. Prentice Hall, 2003.

Complementaria (equivalente en inglés)

- James F. Kurose, Keith W. Ross. *Computer networking: a top-down approach.* Seventh edition. Pearson Education, 2017.
- William Stallings. *Data and Computer Communications.* Ninth edition. Prentice Hall, 2010.
- Andrew S. Tanenbaum. *Computer networks.* Fourth edition. Prentice Hall, 2003.

- Alberto Leon-Garcia, Indra Widjaja. *Communication Networks. Fundamental concepts and key architectures*. McGraw-Hill International Editions, Singapore, 2000.

Capítulo 8

Criptografía y seguridad de redes

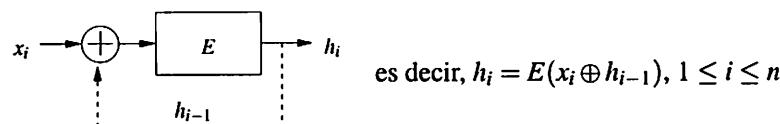
Sergio

1. Tomemos el conjunto de las letras del alfabeto {A, B, C, D, E, F, G, H, I, J, K, L, M, N, Ñ, O, P, Q, R, S, T, U, V, W, X, Y, Z}, y codifiquemos cada letra con su ordinal correspondiente (es decir, la A con 1, la B con 2, hasta las 27 letras y el espacio en blanco es el 0) usando 5 bits.

Supongamos que se dispone de un cifrador de bloque, E , con tamaño de bloque igual a 5 bits con la siguiente tabla de cifrado:

0 \Rightarrow 19	1 \Rightarrow 23	2 \Rightarrow 14	3 \Rightarrow 11	4 \Rightarrow 8	5 \Rightarrow 10	6 \Rightarrow 29	7 \Rightarrow 30
8 \Rightarrow 4	9 \Rightarrow 17	10 \Rightarrow 5	11 \Rightarrow 3	12 \Rightarrow 25	13 \Rightarrow 15	14 \Rightarrow 2	15 \Rightarrow 13
16 \Rightarrow 24	17 \Rightarrow 9	18 \Rightarrow 31	19 \Rightarrow 0	20 \Rightarrow 26	21 \Rightarrow 22	22 \Rightarrow 21	23 \Rightarrow 1
24 \Rightarrow 16	25 \Rightarrow 12	26 \Rightarrow 20	27 \Rightarrow 28	28 \Rightarrow 27	29 \Rightarrow 6	30 \Rightarrow 7	31 \Rightarrow 18

La tabla significa que el 0 se cifra en 19, el 1 en 23, el 2 en 14, etc. Con dicho cifrador en bloque, se construye una función resumen de acuerdo al esquema CBC:



con un valor fijo inicial de $h_0 = 1$. De este modo, si un mensaje está compuesto de n bloques, $m = (x_1, x_2, \dots, x_n)$, su resumen se construye calculando los sucesivos valores h_1, h_2, \dots , de modo que, finalmente, $H(m) = h_n$ (obsérvese que, en realidad, cada bloque es del tamaño de una letra).

En un criptosistema de clave pública, Begoña tiene como clave pública los valores ($n = 253, e = 13$) y Alicia, otra usuaria, tiene como clave pública ($n = 323, e = 11$). Para la realización de la firma, ambas acuerdan públicamente usar como resumen del mensaje el esquema CBC explicado más arriba.

Alicia envía a Begoña un mensaje cifrado (considerando que cada letra es un bloque, y cifrando letra por letra) y firmado. El criptograma, c , y la firma, f , son:

$$c = (144, 136, 26, 126), \quad f = 118.$$

Naturalmente, Eva está espiando y capture el criptograma c y la firma f . Se pide explicar cómo debe atacar Eva el sistema para quebrantar las claves de Alicia y Begoña, de modo que consiga

- averiguar cuál era el mensaje original,
- y si realmente ese mensaje está firmado por Alicia o no.

Aleix

2. La siguiente función iterativa

$$x_{i+1} = x_i^2 \pmod{n}$$

se puede usar como generadora de bits aleatorios de la siguiente manera:

- a) se da un valor inicial x_0 , que es la clave;
- b) se aplica la función anterior iterativamente cuantas veces se quiera para obtener los valores x_1, x_2, \dots
- c) la secuencia de bits aleatorios está constituida por el $\text{LSB}(x_i)$, con $i = 1, 2, \dots$, es decir, por el bit menos significativo de cada valor x_i (excepciónando x_0).

Dos usuarias, Alicia y Begoña, quieren intercambiarse un mensaje secreto mediante un sistema criptográfico con cifrado en flujo, utilizando el anterior generador con un valor del módulo, previamente acordado, $n = 437$.

Para intercambiarse la clave (que es el valor inicial x_0) usan el método de intercambio de clave de Diffie-Hellman, cuyos parámetros son los siguientes: el grupo es $\mathbb{Z}_{257}^* = \{1, 2, \dots, 256\}$, (con la operación de multiplicar módulo 257), y el generador del grupo es $g = 3$ (es decir, todos los elementos del grupo son potencias de 3 módulo 257).

Al comenzar el protocolo, Alicia elige como exponente aleatorio $a = 108$ y recibe de Begoña el elemento $46 \in \mathbb{Z}_{257}^*$.

Con estos datos, se pide

- a) ¿Cuál es el valor de la clave x_0 intercambiada mediante el protocolo Diffie-Hellman?
- b) Begoña ha enviado a Alicia una cadena de bits cifrada con la clave secreta intercambiada, que resulta ser $(0, 1, 1, 1, 1, 1, 0)$ (donde el bit menos significativo es el primero que se ha transmitido). ¿Qué cadena obtendrá Alicia cuando descifre la cadena enviada por Begoña?

B En un criptosistema de clave pública, Begoña tiene como clave pública los valores ($n = 323, e = 67$) y Alicia, otra usuaria, tiene como clave pública ($n = 187, e = 103$). Para la realización de la firma, ambas acuerdan usar como resumen del mensaje la suma de comprobación en 8 bits. Acuerdan también usar el conjunto de las letras del alfabeto, $\{A, B, C, D, E, F, G, H, I, J, K, L, M, N, \tilde{N}, O, P, Q, R, S, T, U, V, W, X, Y, Z\}$, codificando cada letra con su ordinal correspondiente (es decir, la A se codifica como 1, la B como 2, etc., hasta las 27 letras).

Se pide simular un ataque que haría Eva para conseguir:

- a) Quebrantar la clave pública de Begoña y averiguar cuál es su clave privada correspondiente.
 - b) Enviar a Alicia el mensaje LEON, cifrando cada letra por separado de acuerdo a la codificación explicada arriba.
 - c) Firmar un resumen (generado como se indica arriba) del anterior mensaje, para hacer creer a Alicia que realmente se lo está mandando Begoña.
 - d) Reproducir los pasos que dará Alicia para verificar la firma que ella cree procedente de Begoña.

4. Supongamos que la clave pública RSA de una autoridad de certificación AC es ($n = 899$, $e = 817$).

4. Supongamos que la clave pública RSA de una autoridad de certificación AC es ($n = 899, e = 817$).

En un momento dado, AC recibe una petición de un cliente C para que le genere un certificado de clave pública. El certificado consta de dos campos de tamaño 1 byte cada uno, que contienen la clave pública de C : el primer campo almacena el valor $n = 221$ y el segundo campo almacena $e = 133$.

El cometido de la autoridad *AC* es generar la firma RSA del resumen de la clave pública de *C* y añadirla al certificado. Dicho resumen consiste en generar la suma de comprobación (hecha en 8 bits) de los dos campos descritos en el párrafo anterior.

Se pide:

checksum < c_2 overflow

- Alex*

 - a) Suplantar a AC y generar el certificado de la clave pública del cliente C , reproduciendo para ello los cómputos que realizaría AC para crear tal certificado.
 - b) Justificar, a la vista de los resultados anteriores, si la clave pública de AC es segura o no.

5. Para calcular el resumen H de un mensaje, M , Alicia y Benito acuerdan utilizar la función iterativa

$$x_i = a \cdot (x_{i-1} + m_i) + b \pmod{m}$$

en donde se supone que el mensaje se divide en octetos (es decir, en grupos de ocho bits), $M = (m_1, m_2, \dots, m_t)$, y se aplica iterativamente la función, de tal manera que el resumen del mensaje M es $H(M) = x_t$.

La clave pública RSA de Benito, bien conocida por Alicia, es $\{n = 667, e = 17\}$. Supongamos que Benito quiere mandar un mensaje firmado a Alicia y para ello, acuerda con ella que los parámetros de la función iterativa que usarán para firmar serán $m = 521$, $a = 17$, $b = 45$, y $x_0 = 27$.

Ahora, el mensaje que Benito manda a Alicia es su cuenta corriente, compuesta de los siguientes números: $M = 5482783192$. Cada número se codifica de como un octeto de acuerdo a la siguiente tabla

'0' \Rightarrow 48	'1' \Rightarrow 49	'2' \Rightarrow 50	'3' \Rightarrow 51	'4' \Rightarrow 52
'5' \Rightarrow 53	'6' \Rightarrow 54	'7' \Rightarrow 55	'8' \Rightarrow 56	'9' \Rightarrow 57

$$17x = 617;$$

$$\begin{array}{ccccccccc}
 & & & 3 & 3 & 3 & 3 \\
 490 & 17 & \rightsquigarrow & 490 & \cdot 490 & \cdot 490 & \cdot 490 \\
 & & & 490 & 3 & 490^2 & \\
 & & & \underbrace{205}_{15} & 205 & 205 & 205 & 647
 \end{array}$$

y empezando por el octeto menos significativo, es decir, $m_1 = '2'$, $m_2 = '9'$, etc.

Se pide:

Cifrar msg? o dejar indicado?

- a) Reproducir los pasos que tiene que dar Benito para enviar a Alicia dicho mensaje junto con la firma generada usando la función resumen y los parámetros citados, y el sistema RSA.
- b) Reproducir los pasos que Alicia debe dar para decidir si el mensaje ha sido firmado verdaderamente por Benito o no.

6. Supongamos que el conjunto de las letras del alfabeto es {A, B, C, D, E, F, G, H, I, J, K, L, M, N, Ñ, O, P, Q, R, S, T, U, V, W, X, Y, Z}, que cada letra se codifica con su ordinal correspondiente (es decir, la A con 1, la B con 2, hasta las 27 letras y el espacio en blanco es el 0) usando 5 bits. Supongamos que se utiliza un cifrador de bloque con tamaño de bloque igual a 5 y trabajando en modo CBC. La tabla del cifrador en bloque es la siguiente:

0 \Rightarrow 19	1 \Rightarrow 23	2 \Rightarrow 14	3 \Rightarrow 11	4 \Rightarrow 8	5 \Rightarrow 10	6 \Rightarrow 29	7 \Rightarrow 30
8 \Rightarrow 4	9 \Rightarrow 17	10 \Rightarrow 5	11 \Rightarrow 3	12 \Rightarrow 25	13 \Rightarrow 15	14 \Rightarrow 2	15 \Rightarrow 13
16 \Rightarrow 24	17 \Rightarrow 9	18 \Rightarrow 31	19 \Rightarrow 0	20 \Rightarrow 26	21 \Rightarrow 22	22 \Rightarrow 21	23 \Rightarrow 1
24 \Rightarrow 16	25 \Rightarrow 12	26 \Rightarrow 20	27 \Rightarrow 28	28 \Rightarrow 27	29 \Rightarrow 6	30 \Rightarrow 7	31 \Rightarrow 18

La tabla significa que el 0 se cifra en 19, el 1 en 23, el 2 en 14, etc. Un receptor recibe, por orden, los siguientes bloques cifrados:

17, 27, 20, 20, 24, 12, 25, 6, 13, 11, 2, 9, 12

Descifrar el mensaje transmitido.

7. Supongamos que la clave pública RSA de una autoridad de certificación AC es ($n = 899$, $e = 11$). Esta clave es perfectamente conocida por un determinado cliente.

En un momento dado, este cliente recibe un certificado de clave pública de un servidor S . El servidor S indica al cliente que tal certificado ha sido firmado por la autoridad AC. El certificado consta de dos campos de tamaño 1 byte cada uno, que contienen la clave pública de S : el primer campo almacena el valor $n = 221$ y el segundo campo almacena $e = 133$. Además, el certificado contiene la firma realizada por la autoridad de certificación AC sobre la suma de comprobación (hecha en 8 bits) de los dos campos anteriores: el valor de dicha firma es $f = 99$. $\rightarrow AC$

Se pide reproducir los cálculos que ha de realizar el cliente para asegurarse de que el certificado del servidor es válido. Según tal cálculo, ¿es válido ese certificado?

8. Utilizando la siguiente tabla ASCII parcial:

“0” \Rightarrow 48	“1” \Rightarrow 49	“2” \Rightarrow 50	“3” \Rightarrow 51	“4” \Rightarrow 52
“5” \Rightarrow 53	“6” \Rightarrow 54	“7” \Rightarrow 55	“8” \Rightarrow 56	“9” \Rightarrow 57

generar la firma RSA del mensaje que consta de los caracteres “1215”, usando como función resumen la suma de comprobación de 8 bits. La clave privada del usuario que va a firmar es $p = 13$, $q = 17$, $d = 7$. $n = p \cdot q$

Bibliografía

Básica

- James F. Kurose, Keith W. Ross. *Redes de computadoras: un enfoque descendente.* 7^a edición. Pearson Educación, Madrid, 2017.

Complementaria

- William Stallings. *Comunicaciones y Redes de Computadores.* 7^a edición traducida. Prentice Hall, 2004.
- Andrew S. Tanenbaum. *Redes de computadoras.* 4^a edición traducida. Prentice Hall, 2003.
- Alberto Leon-Garcia, Indra Widjaja. *Redes de comunicación, conceptos fundamentales y arquitecturas básicas.* McGraw-Hill, 2002.
- Dimitri P. Bertsekas, Robert G. Gallager. *Data Networks.* Second edition. Prentice Hall, 1992.
- F. Halsall. *Redes de computadoras e Internet.* 5^a edición traducida. Pearson Educación, 2006.
- Behrouz A. Forouzan. *Transmisión de datos y redes de comunicaciones.* 4^a edición traducida. McGraw-Hill, Madrid, 2007.
- W. Richard Stevens. *TCP/IP Illustrated, Volume 1: The Protocols.* First edition. Addison-Wesley, 1994.
- Francisco Manuel Márquez García. *UNIX programación avanzada.* 3^a edición. Ra-Ma, 2004.
- Paul Deitel, Harvey M. Deitel. *C: how to program.* Sixth edition. Prentice Hall, 2009.
- Bruce Eckel. *Thinking in Java.* Third edition. Prentice Hall, 2003.

Complementaria (equivalente en inglés)

- James F. Kurose, Keith W. Ross. *Computer networking: a top-down approach.* Seventh edition. Pearson Education, 2017.
- William Stallings. *Data and Computer Communications.* Ninth edition. Prentice Hall, 2010.
- Andrew S. Tanenbaum. *Computer networks.* Fourth edition. Prentice Hall, 2003.

- Alberto Leon-Garcia, Indra Widjaja. *Communication Networks. Fundamental concepts and key architectures*. McGraw-Hill International Editions, Singapore, 2000.

• Redes - Criptografía

① Jerga - Jerga

$m \rightarrow$ mensaje plano.

$K_A(m) \rightarrow$ mensaje cifrado con clave A. \Rightarrow Criptograma.

$K_B(K_A(m))$

② Servicios

• Confidencialidad:

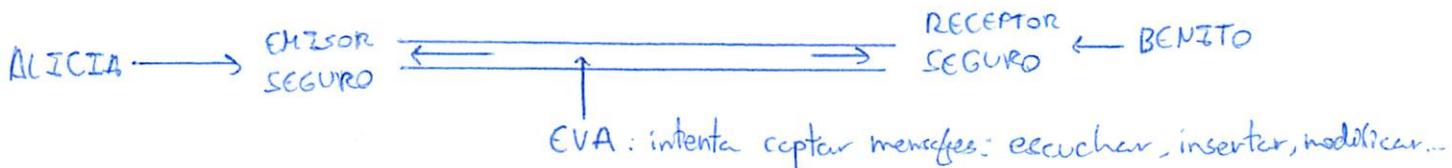
• Autenticación: saber que el emisor es quien dice ser.

• Integridad: que el msg. no sea alterado

• No repudio: un firmante no puede negar que ha firmado.

} Busca una seguridad operacional.

③ Modelo de seguridad



④ Cifrados simples - históricas

4.1. - Sustitución monoalfabética: cada letra equivale a otra.

$\begin{array}{l} \text{ATRACO} \\ \downarrow\downarrow\downarrow\downarrow \\ \text{MLOMBK} \end{array}$	$\begin{array}{l} \text{A} \\ \\ \text{G} \end{array}$	$\begin{array}{l} \text{LAS} \\ \downarrow\downarrow\downarrow \\ \text{GMI} \end{array}$	$\begin{array}{l} \text{TRES} \\ \downarrow\downarrow\downarrow \\ \text{UOC} \end{array}$	Problemas:

{ } ① Análisis de frecuencia
② Facilidad de reconocer preposiciones, conjunciones, artículos, ...

4.2. - Cifrado polialfabético: se toma una red de equivalencias monoalfabéticas. Se usa una para encryptar una letra y se pasa a la siguiente.

Clave: cifradores monoalfabéticas + secuencia

Problemas: los mismos que el anterior. Cuesta más pero se saca.

⑤ Secreto perfecto

• El enemigo sólo conoce (y) un criptograma. (Mensaje cifrado)

• La clave sólo se usa una vez.



Mensaje cifrado = Mensaje + Clave (mod L) donde $L \in \{0, 1, \dots, L-1\}$ símbolos.

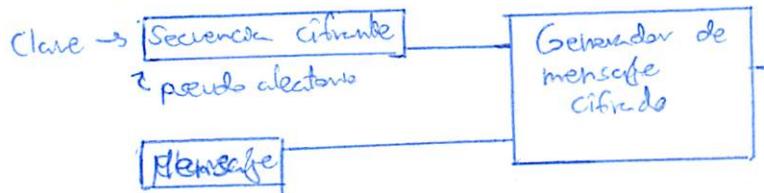
Si la clave es elegida aleatoriamente el criptograma es aleatorio.

⑤ Criptografía clave simétrica

Benito y Ana comparten la misma clave. } Problema: compartir la clave.
 $m \Rightarrow K_s(m) ; K_s(K_s(m)) = m$.

⑥ Cifrado en flujo

Se cifra bit a bit.



Cifrar:
 $\text{bit Mensaje cifrado} = \text{bit sec. cif} \oplus \text{bit. msg.}$

Descifrar:
 $\text{bit msg.} = \text{bit sec. cif.} \oplus \text{bit msg. cifrado}$
 $\oplus \Rightarrow \text{Sumar en mod 2.}$

• Generar sec. por HW-LFSR

Utiliza una rel. linal de recurrencia y un estado inicial (seed) para cifrar un mensaje.

Pasos

- ① Señal valores actuales en mod 2 $\Rightarrow B$
- ② Shift a la derecha de todo.
- ③ La primera pas. (vacía) pasa a ser 0.
- ④ Repetir.

Ejemplo.
 $\text{Rel. Rel.} = X_n = (X_{n-1} + X_{n-2} + X_{n-4} + X_{n-5} + X_{n-8})$

Seed = 1100 1010

Msg. = 1101 1001 0101 0110

n	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
X_n	1	1	0	0	1	0	1	0	1	1	1	1	0	1	0	0
Msg.	1	1	0	1	1	0	0	1	0	1	0	1	0	1	1	0
c(msg)	0	0	0	1	0	0	1	1	0	1	0	0	0	1	0	0

$$X_8 = (X_7 + X_6 + X_4 + X_3 + X_0) \bmod 2$$

$$X_8 = 0 + 1 + 1 + 0 + 1 = 1 \pmod 2$$

y así para el resto de valores de la clave de cifrado que no tengamos.

$$c(\text{msg})(n) = X_n \oplus M_{S(n)}$$

• Generador BBS

- ① Elegir un nsdb. Mod = primo · primo

- ② Tomar un valor inicial $\Rightarrow X_0$

- ③ Formula rel. linal recurrente:

$$X_n = (X_{n-1})^2 \pmod{\text{Mod}}$$

- ④ Se toma el bit menos significativo de X_n .

} Genera una secuencia.

Cifrado en bloque

El msg. en texto plano se cifra en bloques.

Asocia cada bloque a una permutación del mismo tamaño.

Ejemplo tam bloque = 3:

<u>Entrada</u>	<u>Salida</u>
0 0 0	1 1 0
0 0 1	1 1 1
0 1 0	1 0 1
0 1 1	1 0 0
1 0 0	0 1 1
1 0 1	0 1 0
1 1 0	0 0 0
1 1 1	0 0 1

divide en

se cifra en bloques.

una permutación del mismo tamaño.

Es como el cifrado monoalfabetico.

Problema: hacer una tabla de biyecciones de $N=3$ tan grande.

Con $k=64$ es enorme y te ocupa mucho.

Solución: simular la tabla.

CBC

El cifrado del bloque actual depende del anterior.

RSA - Clave pública

Crear clave pública y privada

① Clave ~~simétrica~~ pública: conocida por todos } Emisor y receptor NO

② Clave privada: sólo conocida por el receptor } comparten claves.

$K_B^+ \Rightarrow$ Clave pública } $m \xrightarrow{\text{alg. cifrado}} K_B^+(m) \xrightarrow{\text{Alg. descifrado}} K_B^-(K_B^+(m)) = m$
 $K_B^- \Rightarrow$ Clave privada }

RSA

• Aritmética mod. $\Rightarrow x \pmod n = \text{resto de } x \text{ al ddr. por } n$ ($x \in \mathbb{N}$).

$$(a \pmod n)^d \pmod n = a^d \pmod n;$$

$$\text{ej. } (4 \pmod{10})^2 \pmod{10} = 4^2 \pmod{10}$$

Crear claves pública y privada

① Elegir 2 primos grandes: p y q .

$$② \left\{ \begin{array}{l} n = p \cdot q \\ z = (p-1) \cdot (q-1) \end{array} \right.$$

③ Elegir e tal que $\left\{ \begin{array}{l} e < n \\ \text{mcd}(e, z) = 1 \end{array} \right.$

④ Elegir d tal que $ed \pmod z = 1$.

$$K^+ = (n, e)$$

$$K^- = (p, q, d)$$

Cifrar y descifrar

Cifrar $m \Rightarrow c = m^e \pmod n$ } $m = (m^e \pmod n)^d \pmod n$
 Descifrar $c \Rightarrow m = c^d \pmod n$

Ejemplo

$$① p = 5; q = 7;$$

$$② n = 35 = 5 \cdot 7 = 35$$

$$z = 4 \cdot 6 = 24;$$

$$d = \frac{z + 1}{e}$$

③ $\left\{ \begin{array}{l} e < n \Rightarrow 5 \\ \text{mcd}(e, z) = 1 \Rightarrow 5 \end{array} \right\} 5 = e$

$$④ ed \pmod z = 1; 5d \pmod{24} = 1; 5 \cdot 5 = 25; 25 \frac{124}{24}; d = 5;$$

Conclusión

$$\left\{ \begin{array}{l} p = 5 \\ n = 35 \\ z = 24 \\ e = 5 \\ d = 5 \\ q = 7 \end{array} \right.$$

Cifrar $m = 0000 \ 1100 \Leftrightarrow 12; c = m^e \pmod n \Rightarrow 12^5 \pmod{35}$
 $c = 17$

Descifrar $c = 17; m = c^d \pmod n; 17^5 \pmod{35} \Rightarrow$
 $m = 12$

• RSA

¿Por qué funciona?

$$\text{T. Euler: } x^y \bmod n = x^{(y \bmod \varphi(n)) \bmod n}; \quad \begin{cases} n = pq \\ \varphi(n) = (p-1)(q-1) \end{cases}$$

① Demostrar que $x^d \bmod n = m$:

$$x^d \bmod n = m \equiv (m^e \bmod n)^d \bmod n \\ (m^e \bmod n)^d \bmod n \equiv m^e \bmod n \equiv \cancel{m^e} m^{(ed \bmod \varphi(n))} \bmod n;$$

$$ed \bmod \varphi(n) = 1; \quad m^e \bmod n = m$$

② Demostrar $K_B^-(K_B^+(m)) = m = K_B^+(K_B^-(m))$

$$(m^e \bmod n)^d \bmod n = m^e \bmod n = (m^d \bmod n)^e \bmod n$$

• DLP

Intercambio de claves Diffie - Hellman

① Elegir un primo p .

② Elegir entero $\alpha \in \{1, 2, 3, \dots, p-2\}$

③ $A = \alpha^a \bmod p$ } $a \in \text{elegida por Alicia}$
 $B = \alpha^b \bmod p$ } $b \in \text{elegida por Benito}$

④ Alicia $\xrightarrow{A} \text{Benito}$
 Alicia $\xleftarrow{B} \text{Benito}$

$$⑤ K_{AB} = B^a \equiv (\alpha^b)^a \equiv \alpha^{ab} \bmod p \\ K_{AB} = A^b \equiv (\alpha^a)^b \equiv \alpha^{ab} \bmod p$$

Ejemplo

$$⑥ p = 29$$

$$⑦ \alpha = 2$$

$$⑧ \begin{cases} a = 5; A = 2^5 \bmod 29; A = 3 \\ b = 12; B = 2^{12} \bmod 29; B = 7 \end{cases}$$

⑨ Alicia $\xrightarrow{3} \text{Benito}$
 Benito $\xrightarrow{7} \text{Alicia}$

$$⑩ \text{Alicia } \Rightarrow K_{AB} = B^a = (\alpha^b)^a = \alpha^{ab} \bmod p = 2^{5 \cdot 12} \bmod 29 = 16 \\ \text{Benito } \Rightarrow K_{AB} = A^b = (\alpha^a)^b = \alpha^{ab} \bmod p = 2^{5 \cdot 12} \bmod 29 = 16$$

④ Integridad de mensajes

- Comprobar si el mensaje no ha sufrido alteración
- Comprobar sucesión de mensajes

Función resumen

- ① Toma un mensaje largo de long. variable
- ② En base a él, genera una firma de mensaje de long. fija

Debe de ser	<ul style="list-style-type: none">- Fácil de calcular- Unidireccional- Resistente a colisiones- Que parezca aleatoria	<u>Algoritmos</u> <ul style="list-style-type: none">• MD5 \Rightarrow Obsoleta• SHA-x \Rightarrow Actual.
-------------	--	--

⑤ Firma digital

Benito quiere enviar mensaje m :

- ① Calcula Hash (f. resumen) de m : $H(m)$
- ② Cifra con su clave privada m : $K_B^-(m)$ (opc.)
- ③ Cifra con su clave pública $H(m)$: $K_B^-(H(m))$
- ④ Envía $K_B^-(H(m)) \oplus K_B^-(m)$

Alicia quiere verificar el msg.:

- ① Saca $K_B^-(m)$ y lo descifra con cl. pública $\Rightarrow K_B^-(K_B^-(m)) = m$
- ② Saca $K_B^-(H(m))$ y lo descifra con cl. pública $\Rightarrow K_B^-(H(m)) = H(m)$
- ③ Recalcula $H'(m)$ partiendo de m , y verifica que $H'(m) = H(m)$.

⑥ Authorities de certificación

Como Eva puede tener a Alicia diciendo que es Benito, existen autoridades que registran las claves públicas asociadas a una entidad.

De esta forma, las entidades cifran con su clave privada y tomando la pública de ese registro, podemos saber que Benito es quien dice ser.

No repudio: una vez firmas no puedes decir que no has sido tú.

• SSL

Protocolo de seguridad soportado por navegadores que se apoya en TCP. (Usa un puerto seguro).
(443)

Ofrece

- Confidencialidad
- Integridad
- Autenticación

• Fases

- Negociación: Alicia y Benito se autentican e intercambian clave base: MS
- Derivación de la clave: a partir de MS, Alicia y Benito derivan

○ Repaso examen

① CRC

Datos: 1000 1110

$$g(x) : x^3 + 1 \approx 1001$$

Montarlos: como es de grado 3, añado tres ceros:

$$\begin{array}{r} 1000 \quad 1110 \quad 000 \\ \oplus 1001 \quad (g(x)) \\ \hline 000 \oplus 1110 \quad 000 \\ \hline 1001 \\ \hline -1100 \quad 000 \\ \hline 1001 \\ \hline -101000 \\ \hline 1001 \\ \hline -001100 \\ \hline 1001 \\ \hline -0101 \quad \boxed{\text{CRC}} \end{array}$$

Message a enviar:
1000 1110 101
DATOS CRC

Desmontarlo

$$\begin{array}{r} 1000 1110 101 \\ \oplus 1001 \quad (g(x)) \\ \hline -000 \oplus 1110 101 \\ \hline 1001 \\ \hline -1100 101 \\ \hline 1001 \\ \hline -010101 \\ \hline 1001 \\ \hline -001001 \\ \hline 1001 \\ \hline -000 \quad \boxed{0000} \end{array}$$

Resto = 0 \rightarrow Datos integros

Resto $\neq 0$ \rightarrow Datos no tan integros.

② Reducción de potencias tocadas en mod n

$$\text{Propiedad: } \alpha^{a \cdot b} \bmod n = [\alpha^a \bmod n] \cdot [\alpha^b \bmod n]$$

Técnica: dividir la potencia y aplicar mult. sucesivas.

Ejemplo

$$490^{17} \bmod 667;$$

$$\textcircled{1} \text{ Pasar exp. a binario: } 17_d = 10001$$

$$\textcircled{2} \quad 490^{17} = 490^{16} \cdot 490^1$$

Ejemplo 2

$$21^{26} \bmod 14;$$

$$\textcircled{3} \quad 26_d = 11010;$$

$$\textcircled{2} \quad 21^{26} = 21^2 \oplus 21^8 \cdot 21^{16}$$

$$21^2 \times 14 \leq 7; \quad (\bmod 14)$$

$$21^8 \approx 7^4 \approx 7 \quad (\bmod 14)$$

$$21^{16} \approx 7^2 \approx 7 \quad (\bmod 14)$$

$$(7 \cdot 7 \cdot 7) \times 14 = 7_{//}$$

Ejemplo 3

$$23^{31} \bmod 12;$$

$$\textcircled{3} \quad 31 = 1111_2; \quad 23^{31} = 23^{1+2+4+8+16}$$

$$\left. \begin{array}{l} (1) 23^1 \times 22 = 11; \\ (2) 11^2 \times 12 = 1; \\ (4) 1^2 \times 12 = 1; \\ (8) 1^2 \times 12 = 1; \\ (16) 1^2 \times 12 = 1; \end{array} \right\} 11 \cdot 1 \cdot 1 \cdot 1 \cdot 1 = 11_{//}$$

Ejemplo 4

$$16^{14} \bmod 25; \quad 14 = 1110_b$$

$$\textcircled{1} \quad 16^1 \times 25 = 16_{//}$$

$$\textcircled{2} \quad 16^2 \times 25 = 64_{//} \quad \left. \begin{array}{l} 6 \cdot 11 \cdot 21 = 1386; \\ 1386 \times 25 = 34_{//} \end{array} \right\}$$

$$\textcircled{4} \quad 6^2 \times 25 = 11_{//}$$

$$\textcircled{8} \quad 11^2 \times 25 = 21_{//}$$

• Repaso labo

RIP: Subnetting dinámico.

Router > enable

Configure terminal

auto-summary

Router#

no auto-summary

Version 2

Redes sin routers → passive-interface fa 0/X

Redes con routers → network xxx.xxx.xx.xx

DSPF

Router > enable

configure terminal

wildcard

router ospf 1

network 8.8.8.8 area 0

end

redes con routers

Dom difusión y colisión

hub } Video

Colisión: Switch, router y hub

VLAN.

Difusión: Router y VLAN.

ACL

Restringir tráfico

ACL estandar: origen

wildcard

* access-list 1 permit/deny source netmask

Crear la: { * access-list 1 deny 10.5.3.0 0.0.0.255
access 100 { * access-list 1 permit host 10.5.3.39
100 { * access-list 1 permit any

(en aplicaciones) { * interface Fa 0/1
a un destino { * IP access-group 1 out (out- permit
in- deny

ACL extendida: origen y destino

* access-list 100 permit/deny protocol source netmask destino netmask eg puerto
* access-list 100 permit ip 10.5.4.0 0.0.0.255 host 10.5.64.30 0.0.0.255 eg 80

$$h_i = x_i \oplus h_{i-1}$$

$$12 \oplus 1 = 13$$

$$5 \oplus 10 = 10$$

$$5 \oplus 8 = 13$$

$$15 \oplus 24 = 23$$

$$15 \oplus 1 =$$

(2)

a) Diffie - Hellman

$$g = \alpha = 3; \quad A = \alpha^a \bmod p = 3^{257} \bmod 257 = 3^{108} \bmod 257 = 117;$$

$$p = 257 \quad | \quad B = \alpha^b \bmod p = 46;$$

$$K_S = \alpha^{a \cdot b} \bmod 257; \quad 46^{108} \bmod 257 = 249; \quad a) \quad x_0 = 249$$

$$b) \quad C = (0, 1, 1, 1, 1, 1, 1, 0)$$

$$x_i = (x_{i-1})^2 \bmod n;$$

$$n = 437; \quad x_0 = 249;$$

	LSB	
$x_1 = (249)^2 \bmod 437 = 384;$	0	$K_S = 0111 \quad 0001$
$x_2 = (384)^2 \bmod 437 = 187;$	1	
$x_3 = (187)^2 \bmod 437 = 9;$	1	<u>Mensaje original</u>
$x_4 = (9)^2 \bmod 437 = 81;$	1	$\oplus \quad C \approx 0111 \quad 1110$
$x_5 = (81)^2 \bmod 437 = 6;$	0	$K_S \approx 0111 \quad 0001$
$x_6 = (6)^2 \bmod 437 = 36;$	0	
$x_7 = (36)^2 \bmod 437 = 422;$	0	
$x_8 = (422)^2 \bmod 437 = 225;$	1	$0000 \quad 1111 = m$

$$\textcircled{3} \quad K_B^+ = (n=323, e=67)$$

$$K_A^+ = (n=187, e=103)$$

a) Sacar privada de Begoña:

$$n = p \cdot q = 323; \quad e \cdot d = 1 \pmod{288}$$

$$p = 17 \quad | \quad 67 \cdot d \pmod{288} = 1;$$

$$q = 19 \quad | \quad \left(\begin{array}{cc} 67 & 288 \\ 1 & 0 \end{array} \right) \rightarrow \left(\begin{array}{cc} 67 & 20 \\ 1 & -4 \end{array} \right) \rightarrow \left(\begin{array}{cc} 7 & 20 \\ -3 & -4 \end{array} \right) \rightarrow \left(\begin{array}{cc} 7 & 6 \\ 13 & -30 \end{array} \right) \rightarrow \left(\begin{array}{cc} 1 & 6 \\ 43 & -30 \end{array} \right)$$

$$d = 43$$

b) Cifrar: $m^d \pmod{323};$

$$c_1 = 12^{43} \pmod{323} = 278$$

$$c_2 = 5^{43} \pmod{323} = 130$$

$$c_3 = 16^{43} \pmod{323} = 169$$

$$c_4 = 14^{43} \pmod{323} = 160$$

c) Firma

8 bits $\approx \bmod 255;$
 ~~$(278 + 130 + 169 + 160) \bmod 255 = 227;$~~

~~Checksum = $255 - 227 = 28.$~~

$(12 + 5 + 16 + 14) \bmod 255 = 47;$

Checksum: $255 - 47 = 208$

d) Descifrar: $c^e \bmod n$

$$m_1 = 278^{67} \pmod{323} = 12$$

$$m_2 = 130^{67} \pmod{323} = 5$$

$$m_3 = 169^{67} \pmod{323} = 16$$

$$m_4 = 160^{67} \pmod{323} = 14$$

Check:

$$\textcircled{1} \quad 12 + 5 + 16 + 14 \bmod 255 = 47$$

$$\textcircled{2} \quad 255 - 47 = 208 \quad \checkmark$$

Alicia tec que no se la han pegado pero la han tirado de lo lindo.

Gobernadas Redes Cap. 8

$$\begin{array}{l|l} \textcircled{1} \quad P_B^+ (n=253, e=13) & C = (144, 136, 26, 126) \quad (A \rightarrow B) \\ P_A^+ (n=323, e=11) & f = 118; \quad 118 \text{ mod } 323 = 16 \\ \text{Firma: } h_i = E(x_i \oplus h_{i-1}) & \end{array}$$

a) Descifrar msg.

El msg. estarán cifrados con la clave pública de Begoña. Para descifrarlo tendré que buscar su privada. Como n es muy pequeño, puedo factorizar:

$$253 \left| \begin{array}{c} 11 \\ 23 \\ 1 \end{array} \right. \begin{array}{l} p=11 \\ q=23 \\ z=220 \end{array} \quad \begin{array}{l} ed \text{ mod } 3 = 2; \\ B \text{ d mod } 220 = 2; \end{array} \quad \left(\begin{array}{cc} 13 & 220 \\ 1 & 0 \\ 0 & 1 \end{array} \right) \xrightarrow{\text{ }} \left(\begin{array}{cc} 13 & 12 \\ 1 & -16 \\ 0 & 1 \end{array} \right) \xrightarrow{\text{ }} \left(\begin{array}{cc} 19 & 12 \\ -1 & -16 \\ 0 & 1 \end{array} \right) \quad \left. \begin{array}{l} d=17 \\ \end{array} \right\}$$

Descifrar:

$$m = C^d \text{ mod } n; \quad \begin{cases} m_1 = 144^{17} \text{ (mod 253)} = 12 = L \\ m_2 = 136^{17} \text{ (mod 253)} = 5 = E \\ m_3 = 26^{17} \text{ (mod 253)} = 16 = O \\ m_4 = 126^{17} \text{ (mod 253)} = 14 = N \end{cases}$$

b) Comprobar firma

$$h_0 = 8;$$

$$L = 12 \rightarrow 25$$

$$E = 5 \rightarrow 10$$

$$O = 16 \rightarrow 24$$

$$N = 14 \rightarrow 2$$

XOR BINARIO

$$h_1 = 12 \oplus 8 = 24$$

$$h_2 = 5 \oplus 24 = 18$$

$$h_3 = 24 \oplus 18 = 10$$

$$h_4 = 10 \oplus 8 = 8$$

$$\begin{array}{r} 11001 \\ 00001 \\ \hline 11000 \end{array} \quad \begin{array}{r} 01010 \\ 11000 \\ \hline 10010 \end{array} \quad \begin{array}{r} 11000 \\ 10010 \\ \hline 01010 \end{array} \quad \begin{array}{r} 1010 \\ 00010 \\ \hline 01000 \end{array}$$

$$H(m) = 8 \neq 118;$$

El mensaje no ha sido firmado por Alicia.

$$\textcircled{4} \quad K_{AC}^+ = (n=899, e=817)$$

$$K_C^+ = (n=221, e=133)$$

c) $\textcircled{2}$ Halla K_{AC}^- :

$$n = 899 = 29 \cdot 31$$

$$p = 29$$

$$q = 31$$

$$z = 840$$

$$\boxed{K_{AC}^- = (d=73, n=899)}$$

$$\begin{array}{l} ed \text{ mod } 3 = 1; \\ 817 \text{ d mod } 840 = 2; \\ (817 \xrightarrow{-1} 840) \xrightarrow{\text{ }} (817 \xrightarrow{-35} 23) \xrightarrow{\text{ }} (12 \xrightarrow{-1} 23) \\ (1 \xrightarrow{-1} 0, \xrightarrow{-1} 1) \xrightarrow{\text{ }} (36 \xrightarrow{-1} -35) \xrightarrow{\text{ }} (73 \xrightarrow{8} 37) \end{array} \quad \left. \begin{array}{l} \xrightarrow{-35} \\ \xrightarrow{-1} \\ \xrightarrow{-1} \\ \xrightarrow{-1} \end{array} \right\} d = 73$$

c₂) Firma:

Checksum 8 bits:

$$221 + 133 \text{ (mod 255)} = 99;$$

$$\text{Checksum} = 255 - 99 = 156$$

$$f = \text{firma} = 156$$

b) Bos puntos:

- El checksum como firma es un cagarrro.
↳ tiene muchas colisiones: distintas combinaciones que dan mismo resultado.

- El n de K_{AC}^+ es muy bajo, fácilmente factorizable y por lo tanto débil.

• Gj. 5. Redes 8 - Criptografia

⑥ CBC de 5 bits; $\text{mod } 2^5 = \text{mod } 32$

$$C(0) = 27;$$

$$m(i) = k_s(c_i) \oplus c_{i-1};$$

Letra Coseq. \oplus anterior XOR binaria. en módulo 32

$$m_1 = k_s(27) \oplus 17 = 28 \oplus 17 = 45 \quad (\text{mod } 32)$$

$$m_2 = k_s(20) \oplus 27 = 26 \oplus 27 = 53 \quad (\text{mod } 32)$$

$$m_3 = k_s(20) \oplus 20 = 26 \oplus 20 = 46 \quad (\text{mod } 32)$$

$$m_4 = k_s(24) \oplus 20 = 16 \oplus 20 = 36 \quad (\text{mod } 32)$$

$$m_5 = k_s(12) \oplus 24 = 25 \oplus 24 = 49 \quad (\text{mod } 32)$$

$$m_6 = k_s(25) \oplus 12 = 12 \oplus 12 = 24 \quad (\text{mod } 32)$$

$$m_7 = k_s(6) \oplus 25 = 29 \oplus 25 = 54 \quad (\text{mod } 32)$$

$$m_8 = k_s(13) \oplus 6 = 15 \oplus 6 = 21 \quad (\text{mod } 32)$$

$$m_9 = k_s(11) \oplus 13 = 3 \oplus 13 = 16 \quad (\text{mod } 32)$$

$$m_{10} = k_s(2) \oplus 11 = 14 \oplus 11 = 25 \quad (\text{mod } 32)$$

$$m_{11} = k_s(9) \oplus 2 = 17 \oplus 2 = 15 \quad (\text{mod } 32)$$

$$m_{12} = k_s(12) \oplus 9 = 25 \oplus 9 = 34 \quad (\text{mod } 32)$$

se corresponde a	H
= 13	I
= 14	N
= 4	D
= 1	A
= 0	U
= 4	D
= 5	I
= 14	N
= 5	E
= 15	R
= 16	O
= 2	O

m = "MANADA DINERO"

$$\textcircled{5} \quad M = 5482783292; \quad m = S3 - S2 - S6 - S0 - S5 - S6 - S1 - 49 - 57 - 50$$

$$\text{b) Cifrar } m_i : C(m_i) = m_i^e \pmod{667}$$

② Hash:

$$m = 621$$

$$a = 17$$

$$b = 45$$

$$x_0 = 27$$

$$\begin{aligned} m_1 &= 53^{17} \pmod{667} = 571 \\ m_2 &= 52^{17} \pmod{667} = 219 \\ m_3 &= 56^{17} \pmod{667} = 385 \\ m_4 &= 50^{17} \pmod{667} = 48 \end{aligned}$$

Igual para $m_5 \dots m_{10}$

$$\begin{aligned} x_1 &= 17(27+50) + 45 \pmod{521} = 322 \\ x_2 &= 17(312+54) + 45 \pmod{521} = 28266 \\ x_3 &= 17(66+48) + 45 \pmod{521} = 437 \\ x_4 &= 17(437+51) + 45 \pmod{521} = 5 \\ x_5 &= 17(5+56) + 45 \pmod{521} = 40 \\ x_6 &= 17(40+55) + 45 \pmod{521} = 97 \\ x_7 &= 17(97+50) + 45 \pmod{521} = 1260 \\ x_8 &= 17(1260+56) + 45 \pmod{521} = 482 \\ x_9 &= 17(482+52) + 45 \pmod{521} = 249 \\ x_{10} &= 17(249+53) + 45 \pmod{521} = 490 \end{aligned}$$

$$H(M) = 490$$

③ Aplicar RSA:

$$n = pq = 667 = 23 \cdot 29$$

$$e = 17$$

$$z = 22 \cdot 28 = 616$$

$$\begin{cases} ed \pmod{2} = 1; \\ 17d \pmod{616} = 1; \\ d = 145; \end{cases}$$

$$\begin{aligned} \text{a) Cifrar} \\ c(H(m)) &= 490^{17} \pmod{667} \\ &= (-177)^{17} \pmod{667} \\ \text{b) Decifrar} \\ m &= 433^{145} \pmod{667} = 490 \end{aligned}$$

(3)

① Hash.

$$m = 1215; m^1 = 49 - 50 - 49 - 53;$$

$$\begin{array}{r}
 49 \approx 0011 \ 0001 \\
 50 \approx 0011 \ 0010 \\
 49 \approx 0011 \ 0001 \\
 53 \approx 0011 \ 0101 \\
 \hline
 1100 \ 1001 = 201
 \end{array}$$

$$H(m) = 201.$$

② RSA:

$$\textcircled{1} p = 13; q = 17; n = 221$$

$$\textcircled{2} z = 12 \cdot 16 = 192$$

$$\textcircled{3} \left. \begin{array}{l} e < n \\ e \mid \text{mod}(e, z) = 1 \end{array} \right\} e = 5;$$

$$\textcircled{4} \left. \begin{array}{l} ed \equiv 1 \pmod{z} \\ ed \equiv 1 \pmod{n} \end{array} \right\} d = 11; e = 5$$

Cifrar:

$$c = m^d \pmod{n}; 201^5 \pmod{221} = 176$$

Descifrar:

$$\dots b$$

⑦ Clave pública $\{k_p\}$ $\{n = 899, e = 133\}$ S recibe $\underline{\{n = 221, e = 133\}}$, $i f = 99$

$$\text{Checksum} \left\{ \begin{array}{l} 71101111 \\ 221 \approx 1101 \ 1101 \\ 133 \approx 1000 \ 0101 \end{array} \right.$$

$$\left. \begin{array}{l} 10011101 = 156 \\ 10011101 = 157; \\ 1001100010; \text{ CAI} \Rightarrow 10011101 = 157; \end{array} \right.$$

Descifrada: $m = c^e \pmod{n} \Rightarrow 99^{133} \pmod{899} = 336$
 ~~$157 \pmod{99^{133}} \pmod{221}$~~ No es válido: $99^{133} \pmod{899} = 336 \neq 157$

Checksum =

- ③ Suma modular
② Inversa de la suma

Q. $7+13$ con 4 bits:

$$\text{mod } 2^4 = 1 =$$

$$20 \pmod{16} = 4;$$

$$15 - 4 = 11$$

$$\begin{array}{r}
 111 \\
 0111 \\
 1401 \\
 \hline
 110100 \\
 \text{Checksum: } 1011
 \end{array}$$