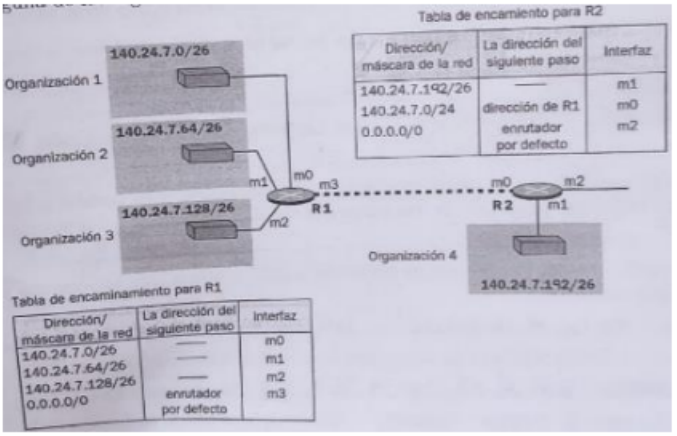
enero 2024

1. (1,5 puntos) Dada la figura, responder a las siguientes preguntas:

a) ¿Puede el router R1 recibir un datagrama con la dirección destino 140.24.7.202? Razona la respuesta indicando, en caso afirmativo, su procedencia. Si esto ocurre, ¿qué pasará con ese datagrama?



Sí. Cualquier datagrama de las organizaciones 1-3 con esta dirección de destino irá a R1.

Si esto ocurre, dado que la dirección no coincide con ninguna de las primeras 3 máscaras de la tabla de R1, se reenviará por la interfaz m3 y llegará a R2. Ahí, dado que coincide con '140.24.7.192/26', se reenviará por la interfaz m1 y llegará a la organización 4.

b) Supongamos que el router R2 recibe un datagrama con la dirección destino 140.24.7.132. ¿Cómo se encamina a su destino final? Describe el procedimiento.

No coincide con la entrada '149.24.7.192/26', pero sí con la entrada '140.24.7.0/24'. Entonces, se reenvía por la interfaz m0 hacia R1, el cual lo reenvía por la interfaz m2 hacia la organización 3.

c) Supongamos que queremos conectar a una nueva interfaz el router R1, m4, la Organización 5, cuya red es 140.24.8.0/26. Modifica o añade las entradas que habría que incluir en cada router. Tener en cuenta que cualquier datagrama con dirección destino no perteneciente a ninguna de las organizaciones debería ser encaminado por la interfaz m2 de R2.

En R1 añadimos una entrada, sin modificar las demás:

140.24.8.0/26	*	m4

En R2 podríamos intentar hacer agregación con la entrada existente que va a m0, pero esto llevaría a redirigir muchos paquetes que no pertenecen a ninguna organización de R0 por la interfaz m0. En su lugar, añadimos una nueva entrada y no modificamos las demás:

140.24.8.0/26	*	m0

.2. (1,5 puntos) Supongamos un sistema de comunicaciones basado en CSMA/CD que interconecta equipos situados a una distancia máxima de 100 metros. La velocidad de transmisión es de 1 Gbps y el retardo de propagación es de 5 ns/m. Las tramas tienen una cabecera de 26 bytes y la carga útil debe estar comprendida entre 64 y 1.474 bytes. Calcular el tiempo máximo que tarda en llegar una trama de un equipo a otro.

Determinar si es posible detectar las colisiones en todos los casos. En caso negativo, indicar soluciones posibles.

t_{prop} = 5ns/m * 100m = $5*10^{-7}$ s

Consideremos el paquete más grande posible, que tardará el tiempo máximo.

t_{trans} = (1500*8)bits / 10^9 Gbps = $1.2*10^{-5}$ s

Tiempo total: $1.2*10^{-5} + 5*10^{-7} = 1.25 * 10^{-5}$ s.

Para asegurar la detección de colisiones, se debe cumplir que $t_{trans} > 2*t_{prop}$. Debemos considerar el t_{trans} del paquete más pequeño posible.

$t_{trans} = (90*8)/10^9 = 7.2 * 10^{-7}$.

No se cumple la desigualdad. Soluciones posibles: aumentar el tamaño de trama mínimo, reducir la velocidad del enlace o reducir el retardo de propagación.

3. (1 punto) Responder brevemente. ¿Por qué no se utiliza el protocolo SMTP como único protocolo de correo electrónico?

SMTP es un protocolo no seguro y se usa para la conexión de usuario a servidor web (envío de correos). Para que el usuario pueda descargar y recibir su correo (servidor→usuario), se usan protocolos seguros como POP3 o IMTP.

4. (1,5 puntos) Responde a las siguientes cuestiones razonando las respuestas:

a) En el mismo instante (ciclo i) en el que se produce la retransmisión de un segmento TCP observamos que las variables del control de congestión correspondientes al umbral (TH) y a la ventana de congestión (CW) valen 1.024 bytes y 1.792 bytes, respectivamente. Sabiendo que el MSS es de 256 bytes, ¿qué podemos deducir que ha ocurrido?

Observamos que CW = TH + 3*MSS. Esto significa la pérdida que ha producido la retransmisión se ha detectado por recibir 3 ACKs duplicados, y se ha aplicado la recuperación rápida. Este mecanismo reduce el TH a la mitad (por lo que antes era de 2048 bytes) y fija después CW al mismo valor, para luego aumentarlo por un MSS por cada ACK duplicado recibido.

b) Durante los ciclos i e i+1 se envían todos los bytes de la ventana de congestión y se reciben los ACKs correspondientes. Determinar el valor de TH y de CW al principio de los ciclos i+1 e i+2.

[i+1]: TH=1024, CW=2048

[i+2]: TH=1024, CW=2304

En cada envío con éxito se aumenta CW por el tamaño de MSS (AIMD)y, mientras que no hay congestión y se recibe los ACKs, no es necesario actualizar TH.

c) ¿Qué ha ocurrido en el ciclo i+2, si al principio del ciclo i+3, TH y CW valen 1.152 bytes y 256 bytes, respectivamente?

Dado que CW = MSS, se ha producido una pérdida que se ha detectado mediante un fin de temporizador y no mediante 3 ACKs repetidos, por lo que la congestión es significativa y se ha vuelto a inicio lento. Esto significa que CW se reduce a 1 MSS y se duplicará hasta llegar al nuevo valor de TH, que es la mitad del previo valor de CW.

5. (1,5 puntos) Suponer que Alicia, cliente, crea una conexión TCP con Benito, el servidor. Intercambian datos y cierran la conexión. Ahora Alicia comienza una nueva conexión con Benito enviando un nuevo segmento SYN. Antes de que Benito reciba este segmento SYN, un duplicado del antiguo segmento SYN de Alicia, que está viajando por la red, llega al ordenador de Benito, provocando el envío de un segmento SYN + ACK desde el ordenador de Benito. ¿Puede este segmento ser confundido por el ordenador de Alicia como respuesta al nuevo segmento SYN? Razonar la respuesta.

Sea x el nº de secuencia del primer SYN enviado por Alicia, y 'y' el del segundo. En respuesta a recibir el SYN con seq=x, Benito enviará un SYN+ACK donde el ACK es x+1. El equipo de Alicia recibirá un mensaje con ACK igual a x+1 cuando se esperaba un ACK de y+1. Es decir, sólo podrá ser confundido si x = y.

El nº de secuencia del primer SYN se decide aleatoriamente al establecer la conexión. Existe una posibilidad, aunque muy pequeña, de que se produzca la confusión ($1/2^{32}$).

6. (1 punto) Indica brevemente qué es y para qué se utiliza la comunicación de etiquetas multiprotocolo (MPLS).

A nivel de internet, MPLS es un mecanismo de encadenamiento que asigna a los paquetes etiquetas multiprotocolo. En los routers MPLS, en lugar de dirigir un paquete a partir de su IP, se redirigirá según su etiqueta, y cada router reemplaza también esta etiqueta con una nueva. Esto presenta ventajas como un redireccionamiento más rápido y seguro. Por esto se usa, por ejemplo, para el establecimiento de VPNs.

7. (2 puntos) Marca con una V las afirmaciones correctas y con una F las incorrectas. En el caso de las incorrectas, corregirlas con una frase. Cada acierto cuenta 0,25 y cada fallo -0,25.

a) El bit PUSH de la cabecera TCP es para indicar al servidor que devuelva inmediatamente la respuesta al cliente.

Falso. Indica al servidor que procese los datos enviados aunque no se haya (halla?) llenado aún el buffer.

b) Los servicios DNS autorizados se suelen comportar como servidores locales.

Verdadero.

c) La primera trama que recibe un ordenador (sin configurar para acceder a Internet) que acaba de llegar a una LAN es una trama que contiene un mensaje del protocolo ARP.

Falso. Será del protocolo DHCP.

d) La multiplexación por división de frecuencia (FDM) se utiliza para acelerar una comunicación concreta.

Falso. Sirve para distribuir recursos entre varias conexiones.

e) La recuperación rápida es un mecanismo de control de congestión que tiene TCP.

(???). Lo tienen algunas versiones de TCP.

f) En las redes de circuitos virtuales el encaminamiento se realiza en función de la dirección del destino.

Falso. En las redes de datagramas sí.

g) A pesar de que la cabecera IPv6 tiene menos campos que la de IPv4, ocupa más bytes que la IPv4 sin opciones.

Verdadero.

h) ARP es un protocolo para obtener la dirección MAC a partir de la IP.

Verdadero.

ENERO 2023

1. 2 routers

R1 tiene las direcciones

140.24.7.0/26 →interfaz m0

140.24.7.64/26 →interfaz m1

140.24.7.128/26 →interfaz m2

0.0.0.0 por defecto → interfaz m3

R2 tiene las direcciones

140.27.7.192 →interfaz

140.27.7.0 →interfaz

0.0.0.0 → el siguiente paso es R1

a) Puede el R1 recibir la dirección 140.24.7.202? En caso afirmativo, indica que pasa.

b) El R2 recibe la dirección 140.27.7.132 describe que pasará y procedimiento.

(aquí faltan unha barbaridad de datos pero)

a) Sí, cualquier paquete que provenga de una de las primeras 3 redes pasará por R1 y será dirigido, a través de la interfaz m3, al router R2, que lo enviará (asumindo que a máscara é /26) a la red 140.27.7.192

b) (dependendo da máscara, pero imagino que) Se reenvía al router R1, que luego lo reenvía por la interfaz m2 hacia la tercera red.

2. Suponer que se tiene un cliente y un servidor web directamente conectado a través de un enlace de velocidad R, que el cliente desea obtener un archivo de tamaño 15S donde S es el MSS y que el RTT es constante. Ignorando las cabeceras del protocolo HTTP, determinar el tiempo necesario para obtener el objeto (incluyendo el tiempo necesario para establecer la conexión TCP), suponiendo que está en la fase de inicio lento, en los siguientes casos:

a) S/R + RTT > 4S/R o bien RTT > 3S/R (RTT alto)

b) S/R > RTT (RTT bajo)

c) 4S/R > S/R + RTT > 2S/R o bien 3S/R > RTT > S/R (RTT intermedio)

Para ello, dibujar los diagramas de tiempo que muestren los segmentos transmitidos.

ejercicio repetidisimo

3. Puede un datagrama de Ipv4 circular por internet de manera indefinida? y de Ipv6? (TTL)

No, en ambos casos. La cabecera IPv4 contiene un campo de 'tiempo de vida' y la de IPv6 un campo de 'nº de saltos'. Ambos impiden que el datagrama circule indefinidamente, y tras cierto nº de pasos sin llegar a su destino se destruirán.

4. Si se aumenta la velocidad de propagación x2, manteniendo todo lo demás constante. Seguiría funcionando la red?

Se debe cumplir que el tiempo de transmisión de el paquete mínimo sea mayor que el doble del tiempo de propagación para poder asegurar que se detectan colisiones. Si duplicamos v_{prop}, se sigue cumpliendo esto. De hecho, permitiría reducir el tamaño mínimo de la trama.

5. Establecer comunicación con el DNS para solicitar una web sin saber nada del DNS (ni su IP ni su MAC). Se conoce la MAC y la IP de todos los elementos de la figura. Explicar cómo van las tramas a lo largo de toda la subred.

Si no se conoce la IP del DNS, será necesario primero obtener una lista de IPs de DNS en la red. Esto se puede conseguir con el protocolo DHCP. Dado que el host ya tiene IP no es necesario un mensaje de descubrimiento DHCP, pero se envía una petición DHCP solicitando la IP de un DNS.

Una vez tiene la IP, se utiliza el protocolo ARP para obtener la MAC del DNS. Se envía una trama con dirección MAC de destino broadcast y IP de destino la obtenida antes. Responderá el servidor DNS indicando su dirección MAC.

Una vez se tienen ambas direcciones, el host puede enviar una petición DNS con ellas. Indicará en el campo de preguntas la el nombre de dominio de la web cuya IP necesita.

6. Conexión TCP de A a B que se quiere enviar un mensaje de 80 bytes. El MSS es de 20 bytes. El inicio de segmento de A es 119 y el de B es de 233. Hace uso de piggybacking, el RTT es constante. La ventana otorgada de B es RW. Indicar el número de secuencia, ACK, los datos en bytes, y las flags SYN y FIN para los siguientes casos:

a. RW = 200 bytes
b. RW = 40 bytes
c. RW = 200 bytes y se pierde el 1er paquete enviado por A.

a)

1. A→B, SYN, SEC = 119
2. B→A, SYN, ACK = 120, SEC = 233
3. A →B. ACK = 234, SEC = 120, datos=20 (piggybacking)
4. A →B. SEC = 140, datos=20
5. A →B. SEC = 160, datos=20
6. A →B, FIN, SEC = 180, datos=20
7. B → A, FIN, ACK=201, SEC=234 (ACK acumulado)
8. A→B, ACK=235

b)

1. A→B, SYN, SEC = 119
2. B→A, SYN, ACK = 120, SEC = 233
3. A →B. ACK = 234, SEC = 120, datos=20 (piggybacking)
4. A →B. SEC = 140, datos=20
5. B →A, ACK=180 (ACK acumulado)
6. A →B. SEC = 160, datos=20
7. A →B, FIN, SEC = 180, datos=20
8. B → A, FIN, ACK=201, SEC=234 (ACK acumulado)
9. A→B, ACK=235

c)

1. A→B, SYN, SEC = 119
2. B→A, SYN, ACK = 120, SEC = 233
3. A →B. ACK = 234, SEC = 120, datos=20 (piggybacking) [no se recibe]
4. A →B. SEC = 140, datos=20
5. B→A, ACK = 120 [recibe paquete incorrecto → indica que aún espera 120]
6. A →B. SEC = 160, datos=20
7. B→A, ACK = 120 [ACK duplicado]
8. A →B, FIN, SEC = 180, datos=20
9. B→A, ACK = 120 [ACK duplicado]
10. A →B, SEC=120, datos=20 [reenvía paquete perdido]
11. B → A, FIN, ACK=201, SEC=234 (ACK acumulado)
12. A→B, ACK=235

7.MPLS. ¿Qué es y para qué sirve?

MPLS es un mecanismo que permite conectar dispositivos de capa de red entre sí y dirigir paquetes de forma eficiente mediante el uso de etiquetas multiprotocolo que se insertan entre la cabecera IP y la cabecera de capa de enlace (ej Ethernet). Dado que en internet se usa para conectar dispositivos IP se estudia como una tecnología de capa e enlace.

Las etiquetas de MPLS son utilizadas por los routers MPLS para dirigir los paquetes sin tener que estudiar su IP. Cada router por el que pasa el paquete actualizará su etiqueta. Este método es más rápido que el redireccionamiento IP y se usa comúnmente en VPNs.

xullo 2022

1. Establecer comunicación con el DNS para solicitar una web sin saber nada del DNS (ni su IP ni su MAC). Se conoce la MAC y la IP de todos los elementos de la figura. Explicar cómo van las tramas a lo largo de toda la subred

Si el host no conoce la MAC ni la IP del DNS, deberá solicitarlo poniéndose en contacto con el servidor DHCP de la red. Si el host ya tiene una IP asignada no es necesario un mensaje de descubrimiento DHCP, sino que enviará una petición DHCP, utilizando su propia IP. IP destino: 255.255.255.255 (broadcast), protocolo: DHCP. Mensaje: solicitud de información de la red, en concreto, la dirección del DNS.

El servidor DHCP recibe el mensaje de descubrimiento y responde con un ACK DHCP. La IP de origen será la del servidor DHCP y la de destino es la del host que realizó la pretición. El propósito de este mensaje es enviar información sobre los servicios de la red, como el DNS.

Ahora que el host tiene la IP del DNS, utilizará el protocolo ARP para 'traducirla' y obtener la dirección MAC del DNS. El host envía una trama ARP con direcciones IP/MAC de origen las suyas, IP de destino la del DNS y MAC de destino la de broadcast (FF:FF:FF:FF:FF:FF). El router que tenga la IP correspondiente responderá con una ARP reply, proporcionando su dirección MAC (que es la del DNS).

Ahora, el host conoce la IP y la MAC del DNS, por lo que puede conectarse con él. Habitualmente se usa el protocolo UDP para consultas DNS. Para solicitar una web, el host usará como direcciones de destino en la consulta DNS las obtenidas previamente, y como puerto de destino el 53 (el puerto del cliente no importa). En el cuerpo del mensaje se incluirá la nombre de la web que el host desea traducir.

2. Tenemos una red ethernet a la que se le ha modificado el cableado de forma que se consigue el doble de velocidad de propagación. Manteniendo Dmax, R y Lmin constantes, ¿funcionaría la red correctamente? ¿Variaría el tamaño mínimo de trama? ¿Cuánto sería en este caso?

Para garantizar que las colisiones en una red ethernet se puedan detectar se debe cumplir que $t_{trans} >= 2t_{prop}$, por lo tanto, L >= R*RTT, donde L es el tamaño mínimo de la trama, R es la velocidad de transmisión en bps y RTT es el tiempo de una señal en ir y volver por toda la red (2*D / Vprop).

Si duplicamos la velocidad de propagación, se reduce a la mitad RTT. Entonces, se sigue cumpliendo la relación, y se seguirá garantizando la detección de colisiones. La red funcionaría correctamente.

Ahora que RTT es la mitad, el valor mínimo de L que cumple la previa relación será también la mitad. El tamaño mínimo estándar de una trama ethernet es de 64 bytes. Si la velocidad de propagación es el doble, el tamaño mínimo de una trama que permite garantizar la detección de colisiones será de 32 bytes.

3. Recuperación rápida. Cuando se retransmite el paquete, umbral=2048 y ventana de congestión=2816. Si MSS=256, ¿qué ha sucedido?

La recuperación rápida es un mecanismo de control de congestión usado en TCP. Cuando se detecta la pérdida de un mensaje, se asume que existe congestión en la conexión, por lo que es necesario reducir el tamaño de la ventana de congestión (VS).

VS empieza en 1 y se duplica sucesivamente hasta llegar a un umbral (umbralAL). Una vez se alcanza el umbral de arranque lento se continúa aumentando de forma aditiva, hasta que se detecte una pérdida de paquete, lo cual indica congestión. Si la pérdida se ha detectado por temporizador se vuelve a iniciar con VS=1, pero si es por 3 ACKs duplicados se usa el mecanismo de recuperación rápida.

Este mecanismo consiste en reducir a la mitad VS, pero luego incrementarlo por un tamaño equivalente a MSS por cada ACK duplicado recibido. El umbral se reduce también al valor de VS antes de contar los ACKs. Una vez se haya recuperado la pérdida, VS vuelve a fijarse al umbral y se continúa con el incremento aditivo.

En este caso, al retransmitir el paquete la ventana de congestión es de 2816 bytes. Conocemos, entonces, que no se ha pasado a inicio lento sino a recuperación rápida, por lo que sabemos que se ha producido una pérdida de un paquete y se ha detectado después de que el receptor enviase 3 ACKs duplicados.

Además, se comprueba que VS = umbralAL+ 3*MSS (2816=2048+3*256). Esto nos confirma que se acaba de usar el mecanismo de recuperación rápida para reducir a la mitad la ventana de congestión (a partir del tamaño del umbral, conocemos que antes de la pérdida era de 2048*2=4096 bytes).

4. ¿Cómo funciona el campo offset de la IPv4? Existe algo similar en IPv6

El campo de offset de la cabecera IPv4 se utiliza en la fragmentación de mensajes, junto con los campos de ‘indicadores’ e ‘identificador’.

El campo de offset indica la posición que ocupa el fragmento dentro del datagrama que ha sido fragmentado. Si contiene el inicio del mensaje o es parte de un datagrama no fragmentado, será 0. De lo contrario, incluye la posición que ocupa el primer byte del fragmento dentro del mensaje total, dividido entre 8 y truncado, pues el offset se mide en bloques de 8 bytes.

En IPv6 no existe fragmentación a nivel de capa de red, por lo que no existe un equivalente a este campo en las cabeceras.

5. A partir de la ip 193.147.12.0/24, dividir en subredes de forma que:

- 1 con 100 ordenadores como mínimo

- 3 para 25 ordenadores

- 1 para 10 ordenadores

a) Indicar dirección base/sufijo, máscara, broadcast, ejemplo y cuántas direcciones sobran

La primera subred necesita 100 hosts mínimo, por lo que al menos requiere 7 bits de host. Esto significa que utilizará 25 bits de red.

- Base: 193.147.12.0 (sufijo: /25)
- Máscara: 255.255.255.128
- Broadcast: 193.147.12.[0111 1111] → 193.147.0.127
- Rango de direcciones: 193.147.12.1 → 193.147.12.126 (128 dir., 126 útiles)

La segunda subred necesita 25 hosts, por lo que utilizará 5 bits de host y 27 de red.

- Base: 193.147.12.128 (sufijo: /27)
- Máscara: 255.255.255.224
- Broadcast: 193.147.12.159 (base + 31)
- Rango de direcciones: 193.147.12.129 → 193.147.12.58 (32 dir., 30 útil)

La tercera subred tiene el mismo nº de direcciones.

- Base: 193.147.12.160 (sufijo: /27)
- Máscara: 255.255.255.224
- Broadcast: 193.147.12.191 (base + 31)
- Rango de direcciones: 193.147.12.161 → 193.147.12.190 (32 dir., 30 útil)

La cuarta subred tiene el mismo nº de direcciones.

- Base: 193.147.12.192 (sufijo: /27)
- Máscara: 255.255.255.224
- Broadcast: 193.147.12.223 (base + 31)
- Rango de direcciones: 193.147.12.193 → 193.147.12.222 (32 dir., 30 útil)

La quinta subred necesita 10 hosts, por lo que utilizará 4 bits de host y 28 de red.

- Base: 193.147.12.224 (sufijo: /28)
- Máscara: 255.255.255.240
- Broadcast: 193.147.12.239 (base + 15)
- Rango de direcciones: 193.147.12.225 → 193.147.12.238 (16 dir., 14 útil)

Dado que la red original tiene como dirección de broadcast la 193.147.12.255, por lo que quedan sin asignar las direcciones del rango 193.147.12.240 - 193.147.12.255 (16 direcciones).

b) ¿Se podrían crear más subredes? En ese caso indicar dirección base/sufijo y direcciones disponibles

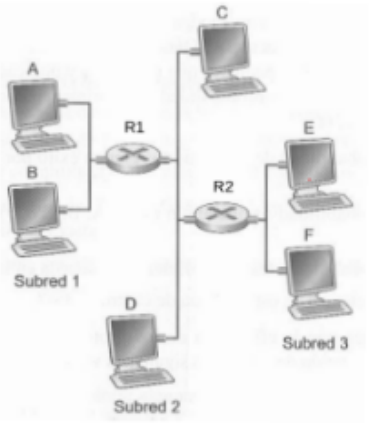
- Base: 193.147.12.240 (sufijo: /28)
- Máscara: 255.255.255.255
- Broadcast: 193.147.12.255 (base + 15)
- Rango de direcciones: 193.147.12.241 → 193.147.12.254 (16 dir., 14 útil)

c) Comparar el número de direcciones sobrantes para los casos anteriores

En el apartado a) sobraban 16 direcciones. Tras crear una subred adicional, no hay ninguna dirección sobrante, ya que sobraban 16 y las 16 son agrupadas en la previa subred.

6. Fragmentación. MTU1=1700 bytes, MTU2=796 bytes, MTU3=1500 bytes. Se envía de F a A 2048 bytes. Indicar tamaños de los fragmentos y nº de fragmentos en subredes 3 y 1. Indicar el offset del último segmento. Asumir el tamaño de las cabeceras sin opciones: TCP de 20 bytes, UDP de 8 bytes, IPv4 de 20 bytes e IPv6 de 4 bytes

(non está claro no enunciado, asumo que o hai que facer para TCP e UDP e para IPv4 e IPv6 porque che da as cabeceras todas)



TCP con IPv4: Cabecera total de 40 bytes.

- Subred 3: MTU=1500 bytes.
  - Como máximo se pueden transmitir 1460 bytes de mensaje. El mensaje de 2048 se debe segmentar en **2**.
  - **Segmento 1:** 1460 bytes de mensaje
    - Tamaño: 1500 bytes
    - Offset: 0
  - **Segmento 2:** 2048-1460=588 bytes de mensaje
    - Tamaño: 628 bytes
    - Offset: 0 (1460/8 (truncado) = 182)[1]
- Subred 2: MTU=796 bytes.
  - El segmento 1 supera el MTU y se segmenta de nuevo. Como máximo se pueden transmitir 796-40=756 bytes de mensaje.
  - **Segmento 1a**: 756 bytes de mensaje.
    - Tamaño: 796 bytes
    - Offset: 0
  - **Segmento 1b:** 1460-796=664 bytes de mensaje
    - Tamaño: 704 bytes
    - Offset: 0 (756/8 (truncado) = 94)
  - **Segmento 2:** no supera MTU, sigue igual
- Subred 1: MTU = 1700 bytes.
  - Ningún segmento supera MTU → se mantienen los mismos (total: 3 segmentos)

[1] o valor do campo 'desplazamiento de fragmentación' na cabecera IP sería 0 porque non se produce fragmentación a nivel de red, pero imagino que ainda así hai que calcular o offset do segmento dentro do mensaje

TCP con IPv6: Igual, pero con cabecera de 24 bytes. Los tamaños y nº de paquetes son los mismos que con IPv4. El offset del último paquete sería (1500-24 /8) = 184.

UDP con IPv4: UDP no segmenta los mensajes. Se crea un único segmento de 2048+8 = 2056 bytes. Luego, será IPv4 el protocolo encargado de fragmentarlo.

- Subred 3: MTU=1500 bytes.
  - **Segmento 1:** 1480 bytes de mensaje (la cabecera UDP se considera parte del mensaje)
    - Tamaño: 1500 bytes
    - Offset: 0
  - **Segmento 2:** 2056-1480 = 576 bytes de mensaje
    - Tamaño: 576
    - Offset: 1480/8 = 185
- Subred 2: MTU=796 bytes.
  - **Segmento 1a**: 776 bytes de mensaje.
    - Tamaño: 796 bytes
    - Offset: 0
  - **Segmento 1b:** 1480-776=704 bytes de mensaje
    - Tamaño: 724 bytes
    - Offset: 776/8 (truncado) = 97
  - **Segmento 2:** no supera MTU, sigue igual
    - Tamaño: 576
    - Offset: 1480/8 = 185
- Subred 1: MTU = 1700 bytes.
  - Ningún segmento supera MTU → se mantienen los mismos (total: 3 segmentos)

UDP con IPv6: UDP no segmenta los mensajes. IPv6 no permite la segmentación en nodos intermedios, por lo que cuando detecte el segmento de 2060 bytes cuando la MTU es de 1500, enviará un ICMP de 'paquete demasiado grande' al emisor.

7. ¿Cómo se implementan VPNs con túneles IP?

Para implementar una VPN se puede usar un túnel IP, donde los extremos serán los gateways VPN. Cuando se quiere pasar un mensaje por esta red, deberá pasar primero por el servidor VPN que encapsula el mensaje añadiendo una cabecera adicional (para que la IP de destino sea la del otro extremo del túnel) y encriptará el contenido utilizando un protocolo determinado.

El mensaje viaja encriptado por el túnel IP y al llegar al destino, el cliente VPN elimina la cabecera añadida y desencripta el mensaje devolviéndolo al estado previo. Finalmente, lo reenvía al destino del mensaje.

De esta forma, el mensaje viaja por internet, pero al pasar por un túnel IP mientras viaja por la red el contenido está cifrado, mejorando la privacidad y la seguridad.

enero 2020

1. **(2,5 puntos)Esquema de red, 3 routers (RA, RB, RC) con distintas IP 192.53.10.X (con X distinto en cada subred) conectadas a cada uno. FALTAN DATOS???**
    a. Máscara de las subredes A y D en formato sufijo y dirección.
    b. La dirección base de la subred B.
    c. Si es posible agregación de rutas en el router RB con las subredes E y D y si la entrada para un
    d. caso puede ser la de por defecto.
    e. Tablas de los Routers y encaminamiento de paquetes
    f. Dirección MAC de destino de un datagrama enviado desde la subred E hasta la subred A.
    g. Decir si es posible cambiar la dirección de un host a otra y seguir estando en la subred

    (ej 23 do boletin)

2. **(2 puntos) Sobre el esquema anterior supongamos que las redes A,B y D tienen un MTU de 1500 Bytes y las redes F, E y C tienen un MTU de 536 Bytes. Supongamos que se desea enviar un paquete de 3112 Bytes mediante el protocolo UDP desde un host de la red D hasta uno de la red B. La capa de aplicación añade 64 bytes. La cabecera UDP 8 bytes. Las cabecera IPv4 20 bytes y la cabecera IPv6 40 bytes.**
    a. Numero y tamaño de datagramas IPv4 que viajan por la red D
    b. Numero y tamaño de datagramas IPv4 que viajan por la red C
    c. Desplazamiento indicado en la cabecera IPv4 del ultimo datagrama del apartado anterior
    d. Numero y tamaño de datagramas IPv4 que viajan por la red B
    e. Numero y tamaño de datagramas IPv6 en la red D

    (falta o esquema pero pola forma na que está formulada creo que vai D→C→B)

    Total de datos: 8+64+3112 = 3176bytes (la cabecera UDP se añade solo una vez)

a) Datos por fragmento: 1500-20 = 1480 bytes.
Son necesarios 2 fragmentos de 1500 bytes y uno de (3176-1480-1480+20)=236 bytes.

b) (asumindo que xa pasou pola red D)
Cada fragmento de 1500 bytes se debe volver a fragmentar.
Datos por fragmento: 536-20=516.
Cada fragmento de 1500 bytes se fragmenta en dos de 536 bytes y uno de (1480-516-516+20)=468 bytes. El de 236 bytes no supera MTU y no hay que fragmentarlo.
Total: 4 fragmentos de 536 bytes, 2 fragmentos de 468 bytes y uno de 236 bytes.

c) Dado que los 6 primeros datagramas han transmitido 3096 bytes, el último fragmento transmite los datos a partir del 3096. El campo se expresa en bloques de 8 bytes: (3096/8)=387.

d) (asumindo que este é como o do 2022 e que xa pasou pola red C) Los mismos que en la previa red, pues ya está fragmentado y ningún fragmento supera la MTU de B.

e) No existe fragmentación a nivel de capa de red en IPv6. Se enviará un mensaje de error al emisor, que debe dividir el mensaje correctamente.

3. (1 punto) Protocolos de las redes inalámbricas, funcionamiento y cómo manejan las colisiones.

Para el acceso al medio las redes WLAN utilizan el protocolo MACA o CSMA/CA. Cuando un host quiere transmitir algo, primero sondea el medio, esperando un intervalo denominado DIFS. De estar ocupado, vuelve a esperar, y esto se repite con un algoritmo de espera exponencial binario hasta que esté libre.

Las colisiones se manejan mediante el uso de tramas de control. El emisor solicita el canal con una trama de petición de envío RTS que incluye el tiempo de ocupoación, y el destino responde con una trama de reserva del canal CTS. Tras reservarse el canal (si no se recibe CTS, el host sabe que hubo colisión y que el canal no se pudo reservar), los demás hosts deberán esperar a que esté libre para poder usarlo. Los hosts esperarán como mínimo el tiempo que solicitó el emisor antes de comprobar si el canal está libre. Se usan también ACK para confirmar el fin de una transmisión.

4. (2 puntos) Dado un tamaño L de archivo en bytes, una velocidad de enlace 622Mb/s y una velocidad de empaquetado de 128Kb/s. Cabeceras de protocolos de 40 bytes en total.
asumindo datos en kilobits/s e megabits/s

a. Cálculo de retardo de creación de paquetes (retardo de empaquetado) respecto de L.
L*8 / ($128*10^3$)

b. Cálculo del retardo de creación de paquetes para los tamaños 1400 y 60. Sabiendo que a partir de los 20ms de retardo se produce un eco molesto, qué tamaño de paquete es mejor usar.
$1400*8/(128*10^3) = 0.875s=875ms$
$60*8/(128*10^3) = 0.00375ms = 3.75ms$
Mejor el segundo tamaño para evitar el eco.

c. Calcula el tiempo de transmisión para los dos tamaños de paquetes del apartado anterior.
$1400*8 / (622*10^6)=18$ µs
$60*8 / (622*10^6)=0.771$ µs

d. Ventajas e inconvenientes de enviar paquetes pequeños.
npi

5. **(1 punto) Notificación explícita de congestión. Qué es y funcionamiento.**

   Es un conjunto de bits que los routers activan en la cabecera IP de un fragmento para avisar de que la congestión es significativa. Cuando lo ve el receptor, avisa al emisor mediante un flag que se incluye en el encabezado del próximo paquete TCP. Esto avisa al emisor de que reduzca el tamaño de la VC (de la misma forma que en retransmisión rápida).

6. **(1,5 puntos) Sea una página web de tamaño MSS que contiene N objetos de tamaño MSS cada uno. A partir de que N es mejor el esquema de HTTP no persistente con N conexiones paralelas que el esquema de HTTP persistente sin entubamiento.**

En no persistente, se tardan dos RTT en solicitar la conexión, y luego comienza la transferencia de datos. Hay que recibir la página y los N objetos, y disponemos de N conexiones paralelas. En total cada solicitud y envío de objeto tardará dos RTT y el tiempo de transmisión. La carga de todos los objetos se realiza siempre simultáneamente, en paralelo. Entonces, el tiempo total es de 4 RTT y 2 t_{trans}.

En HTTP persistente sin entubamiento, se realiza primero la conexión y descarga de la página (2RTT + t_{trans}) y luego cada objeto implica un coste adicional de RTT+t_{trans}. Para N=1 el total será 3RTT+2t_{trans} (mejor que el previo), para N=2 será 4RTT+3t_{trans} (peor). Entonces, a partir de N=2 ya es mejor el esquema no persistente con 2 conexiones paralelas.

enero 2017

1. Suponer que hay cuatro nodos conectados a un concentrador mediante enlaces Ethernet a 10 Mbps. Las distancia entre el concentrador y estos cuatro nodos son 300, 400, 500 y 700 metros respectivamente. La velocidad de propagación de la señal es de 2 x 10^8 m/s. ¿Cuál es el tamaño mínimo de trama requerido? ¿Cuál es el tamaño máximo de trama requerido?

Al ser un concentrador, cualquier transmisión a un nodo se transmite por todos los enlaces, por lo que el tamaño de trama mínimo debe ser suficiente para todos los enlaces. El tamaño mínimo de trama requerido dependerá de la distancia máxima que debe recorrer una trama entre todos los potenciales casos. En este caso, consideramos la distancia más grande de las 4 (700 metros).

Para asegurar la detección de colisiones se debe cumplir que el tiempo que tarda en transmitirse una trama competa debe ser mayor que el que tarda una señal en viajar de un extremo a otro del enlace. Entonces, L_{min} >= r * RTT, siendo:

L_{min}: tamaño mínimo de trama

r: 10^7bps

RTT: 2 * 700 / ($2*10^8$) = $7*10^{-6}$ s

Obtenemos que el tamaño mínimo de trama requerido es de 70 bits, o de 9 bytes. El tamaño máximo de trama en Ethernet es de 1518 bytes y no depende del enlace.

2. Suponer que la MTU de los enlaces entre el host A y el host B está limitado a 1500 bytes. Indicar cuántos datagramas IPv4 se necesitarían para enviar un archivo de 4000 bytes en los siguientes casos:

a) **La aplicación utiliza TCP con un MSS de 1460 bytes.**
b) **La aplicación utiliza UDP.**

Especificar para cada caso el tamaño, el valor del campo identificación suponiendo que comienza en 356, en valor de los indicadores MF (más fragmentos) y NF (no fragmentar) y el valor del campo desplazamiento de cada uno de los datagramas. Asumir el tamaño de las cabeceras sin opciones: TCP de 20 bytes, UDP de 8 bytes, IP de 20 bytes. ¿Qué ocurriría en ambos casos con IPv6?

TCP: Al ser TCP, es necesario enviar 2 datagramas primero, uno con SYN y otro con ACK tras recibir el SYN del host B. Estos datagramas no pueden ser fragmentados (NF=1) y no llevan datos (tamaño=40).

Dado que se desean enviar 4000 bytes y el MSS es 1460, serán necesarios 3 segmentos. Los dos primeros llevan 1460 bytes de datos y ocupan 1500 bytes en total. El último lleva los 1080 bytes restantes y ocupa 1120 bytes. El valor del campo NF puede ser 1 o 0 dependiendo de la implementación.

El valor del campo MF será 0 para todos pues cada datagrama es su propio fragmento (no es necesaria fragmentación a nivel IP), y por el mismo motivo todos tienen 0 en el campo desplazamiento.

Tamaño	Identificación	MF	NF	Desplazamiento
40	356	0	1	0
40	357	0	1	0
1500	358	0	X	0
1500	359	0	X	0
1120	360	0	X	0

UDP: No ofrece segmentación a nivel de capa de transporte, por lo que ocurrirá fragmentación IP.

A la capa de red llega un único segmento de 4008 bytes. Dado que el tamaño del segmento total supera la MTU, es necesario que el campo de 'NF' sea 0 para poder fragmentarlo a nivel de capa de red.

Teniendo en cuenta la cabecera IP de 20 bytes y el MTU de 1500, los datos de este segmento se deberán dividir en fragmentos que lleven, como máximo, 1480 bytes. Serán necesarios 3 fragmentos. Todos forman parte del mismo segmento, por lo que el nº de identificación será el mismo.

El primer datagrama lleva 1480 bytes de datos y ocupa 1500. Su desplazamiento es 0 por ser el primero.

El segundo lleva 1480 bytes de datos y ocupa 1500. Dado que lleva los bytes desde el 1480 hasta el 2959, su desplazamiento es de 1480/8 = 185.

El tercer segmento lleva los 1048 bytes restantes, por lo que con la cabecera ocupa 1068. Su desplazamiento será 370.

El valor de MF (más fragmentos) será 1 para los dos primeros y 0 para el último, pues tras este no quedan más fragmentos.

Tamaño	Identificación	MF	NF	Desplazamiento
1500	356	1	0	0
1500	356	1	0	185
1068	356	0	0	370

TCP con IPv6: Dado que la segmentación se produce en capa de transporte, usar IPv6 sólo afecta a que cambiará el tamaño de la cabecera (en lugar de 40 bytes serían 20+6=26). Esto permite que se envíen 14 bytes más por mensaje. El tamaño del último mensaje sería menor (1078 bytes).

UDP con IPv6: IPv6 no permite fragmentación en routers intermedios. Se enviará al host un mensaje ICMP de 'segmento demasiado grande', y es el emisor el encargado de solucionarlo.

3. ¿Cuál es el retardo total de una trama de 5 millones de bits que se envía por un enlace con 10 routers, cada uno de los cuales tiene un tiempo de espera en la cola de 2 µs y un tiempo de procesamiento de 1 µs? La longitud total de los enlaces es de 2000 km y la velocidad de la señal a través de los enlaces es de $2*10^8$m/s. Los once enlaces tienen un ancho de banda de 5mbps. ¿Qué componente del retardo total es dominante? ¿Cuál es despreciable?

Calculamos las 4 componentes del retardo.

Retardo de transmisión: $5*10^6$ bits / $5*10^6$bps = 1 s

Retardo de propagación: $2*10^6$ m / $2*10^8$ m/s = 0.01 s

Retardo de cola: 10 * 2 µs = 20 µs

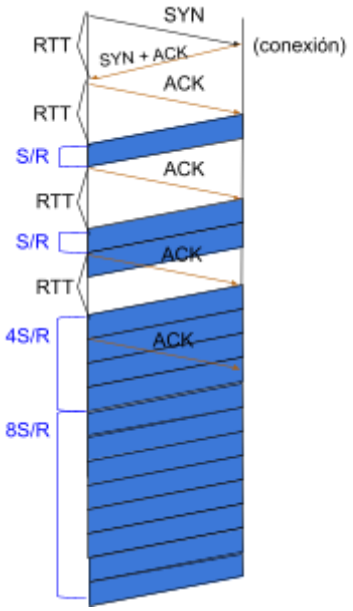
Retardo de procesamiento: 10 * 1 µs = 10 µs.

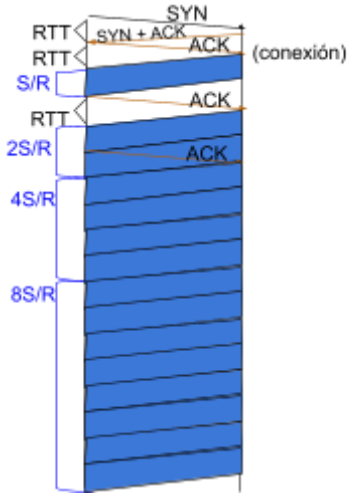
Retardo total: 1.01 s + 30 µs. El retardo de transmisión es dominante, y los retardos de cola y procesamiento son comparativamente despreciables.

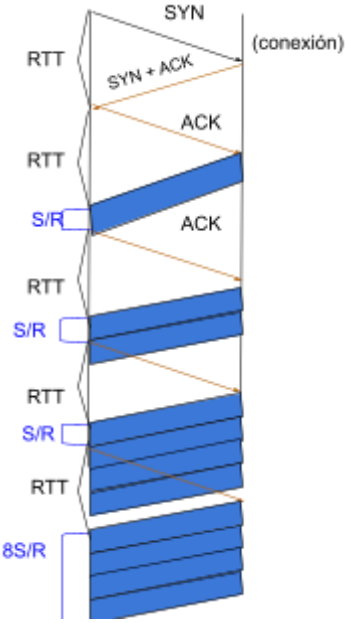
4. Suponer que se tiene un cliente y un servidor web directamente conectado a través de un enlace de velocidad R, que el cliente desea obtener un archivo de tamaño 15S donde S es el MSS y que el RTT es constante. Ignorando las cabeceras del protocolo HTTP, determinar el tiempo necesario para obtener el objeto (incluyendo el tiempo necesario para establecer la conexión TCP), suponiendo que está en la fase de inicio lento, en los siguientes casos:

a) **4S/R > S/R + RTT > 2S/R**
b) **S/R + RTT > 4S/R (RRT alto)**
c) **S/R > RTT (RTT bajo)**

Para ello, dibujar los diagramas de tiempo que muestren los segmentos transmitidos.







a) RTT + RTT + S/R + RTT + S/R +RTT + 4S/R + 8S/R (4RTT + 14 S/R)
b) RTT + RTT + S/R + RTT + 2S/R + 4 S/R + 8 S/R (3RTT + 15 S/R)
c) RTT + RTT + S/R + RTT + S/R + RTT + S/R + RTT + 8S/R (5RTT + 11S/R)

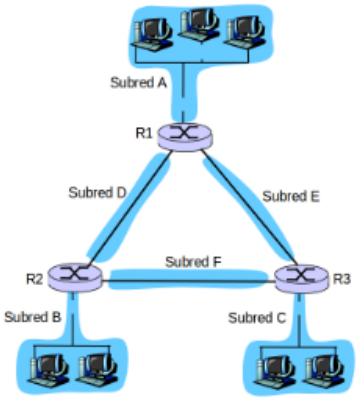
5. Representa en un diagrama todos los pasos involucrados en la resolución de nombres recursiva donde el equipo jefe.empresa.com consulta un servidor DNS (dns.empresa.com) por la dirección IP resuelta al host www.serrico.gr. Supón que la caché DNS del servidor TLD dispone de la entrada correspondiente con la IP del host www.serrico.gr. Completar la siguiente tabla con los datos de los sucesivos mensajes DNS que se producen. Representar las direcciones IP que necesites durante todo el proceso.

El TLD ya tiene la entrada en su caché, por lo que no necesita llamar a un servidor DNS autorizado, sino que devuelve la entrada directamente

Paso	Origen	Destino	Tipo	Información
1	jefe.empresa.com	dns.empresa.com	consulta	Ip de www.serrico.gr?
2	dns.empresa.com	DNS raíz	consulta	“
3	DNS raíz	Servidor TLD (IP)	consulta	“
4	Servidor TLD (IP)	DNS raíz	respuesta	Ip: A.B.C.D
5	DNS raíz	dns.empresa.com	respuesta	“
6	dns.empresa.com	jefe.empresa.com	respuesta	“

6. A partir de la red 198.144.130.0/23, asignar direcciones IP a cada una de las seis subredes de la figura, teniendo en cuenta las siguientes consideraciones: la subred A dispondrá de direcciones suficientes como para dar soporte a 250 interfaces, la subred B a 120 interfaces y la subred C a 60 interfaces. Las subredes D, E y F, al no tener hosts conectados, es suficiente con dos interfaces cada una. Para cada una de las subredes, especificar la dirección de red (en formato a.b.c.d/x) y el rango de direcciones. En base a la asignación realizada, indicar las entradas que habría que incluir en el router R1. Si es posible, aplicar agregación de rutas.

A: 250 interfaces (al menos 8 bits de host) → 24 bits de red.



- Dirección: 198.144.130.0/24
- Rango: 198.144.130.0 - 198.144.130.255

B: 120 interfaces (al menos 7 bits de host) → 25 bits de red.

- Dirección: 198.144.131.0/25
- Rango: 198.144.131.0 - 198.144.130.127

C: 60 interfaces (al menos 6 bits de host) → 26 bits de red.

- Dirección: 198.144.131.128/24
- Rango: 198.144.131.128 - 198.144.130.191

D,E,F: 2 interfaces (al menos 2 bits de host) → 30 bits de red.

- Dirección: 198.144.131.192/30
- Rango: 198.144.131.192 - 198.144.130.195
- Dirección: 198.144.131.196/30
- Rango: 198.144.131.196 - 198.144.130.199
- Dirección: 198.144.131.200/30

- Rango: 198.144.131.200 - 198.144.130.203

Asumimos que R1 tiene 3 interfaces, la primera conecta con la subred A, la segunda con la D y la tercera con la E.

red destino	interfaz de enlace
A (198.144.130.0/24)	1
B (198.144.131.0/25)	2
C (198.144.131.128/26	3
D (198.144.131.192/30)	2
E (198.144.131.196/30)	3
F (198.144.131.200/30)	2
default (0.0.0.0)	X

Se pueden agregar, por ejemplo, B,D y F en 198.144.131.0/24, con la interfaz de enlace siendo la segunda.

red destino	interfaz de enlace
A (198.144.130.0/24)	1
B,D,F (198.144.131.0/24)	2
C (198.144.131.128/26)	3
E (198.144.131.196/30)	3
default (0.0.0.0)	X

Aunque la red ‘198.144.131.0/24’ es muy amplia e incluye direcciones cuyos paquetes deberían ser dirigidos a C y E, como estas entradas de la tabla son más específicas, los paquetes que deban ir a C y a E serán dirigidos por la interfaz 3 y no por la 2. De esta forma se aprovecha al máximo la agregación de rutas. (otra alternativa sería agregar C, E y F con interfaz 3).

xuño 2017

1. Suponer un conmutador Ethernet con autoaprendizaje que tiene seis nodos, A, B, C, D, E, F, conectados en estrella. Supongamos que ocurren los siguientes sucesos en orden:

I. **B envía una trama a E**
II. **E responde enviando una trama a B**
III. **A envía una trama a B**
IV. **B responde enviando una trama a A**

Inicialmente la tabla del conmutador está vacía. Mostrar el estado de la tabla del conmutador antes y después de cada uno de estos sucesos. Para cada suceso identificar el enlace a través de los cuales se reenviará la trama transmitida y justificar brevemente las respuestas.

Estado inicial:

Dirección MAC	**Enlace**

B envía una trama a E: no se conoce el enlace de 'E', así que enviará la trama por todos los enlaces excepto el de origen. Además, aprende la dirección de B y su enlace a partir de la interfaz de entrada. Sea '2' el enlace de B.

Dirección MAC	**Enlace**
B	2

E responde enviando una trama B: Se consulta la tabla y se reenvía por el enlace 2. Además, se aprende la dirección de origen E y la asocia con el enlace de origen de E.

Dirección MAC	**Enlace**
B	2
E	5

A envía una trama a B: Mismo caso, se aprende la dirección/enlace de A y se reenvía la trama al enlace 2.

Dirección MAC	**Enlace**
B	2
E	5

A	1

B envía una trama a A: Se conoce la dirección de B y de A, no es necesario modificar la tabla. La trama se reenvía al enlace 1.

Dirección MAC	**Enlace**
B	2
E	5
A	1

2. Suponer que la MTU de los enlaces entre el host A y el host B está limitado a 1500 bytes. Indicar cuántos datagramas IPv4 se necesitarían para enviar un archivo de 4000 bytes en los siguientes casos:

a) **La aplicación utiliza TCP con un MSS de 1460 bytes.**
b) **La aplicación utiliza UDP.**

Especificar para cada caso el tamaño, el valor del campo identificación suponiendo que comienza en 356, en valor de los indicadores MF (más fragmentos) y NF (no fragmentar) y el valor del campo desplazamiento de cada uno de los datagramas. Asumir el tamaño de las cabeceras sin opciones: TCP de 20 bytes, UDP de 8 bytes, IP de 20 bytes. ¿Qué ocurriría en ambos casos con IPv6?

(aaaaaaaaaaa)

3. Suponer un enlace de microondas a 10 Mbps entre un satélite geoestacionario y su estación base en la Tierra, a una distancia de 36.000 Km. El satélite toma (una) fotografía digital por minuto y la envía a la estación base. La velocidad de propagación es de $2.4*10^8$m/s

a) **¿Cuál es el retardo de propagación del enlace?**
b) **Calcular el producto retardo por ancho de banda**
c) **Sea x el tamaño de la fotografía en bytes. Calcular el valor mínimo de x para que el enlace esté transmitiendo continuamente.**

a) El retardo de propagación es el tiempo que tarda la señal en viajar por el enlace.

$3.6*10^6$m / $2.4*10^8$m/s = 0.15 s

b) Se asume que el retardo es sólo por propagación.

10 Mbps * 0.15s = $1.5*10^6$bit

c) para que esté transmitiendo continuamente, debe cumplirse que tarde 1 minuto en transferirse.

$10*10^6*60 / 8 = 7.5*10^7$ bytes = 75 MB

4. Suponer que se tiene un cliente y un servidor web directamente conectado a través de un enlace de velocidad R, que el cliente desea obtener un archivo de tamaño 15S donde S es el MSS y que el RTT es constante. Ignorando las cabeceras del protocolo HTTP, determinar el tiempo necesario para obtener el objeto (incluyendo el tiempo necesario para establecer la conexión TCP), suponiendo que está en la fase de inicio lento, en los siguientes casos:

a) **4S/R > S/R + RTT > 2S/R**
b) **S/R + RTT > 4S/R (RRT alto)**
c) **S/R > RTT (RTT bajo)**

Para ello, dibujar los diagramas de tiempo que muestren los segmentos transmitidos.

(outra vez este)

5. Supongamos que el ISP A conecta a 4 organizaciones, asignando las direcciones IP de la siguiente manera:

- **200.23.16.0/23 a la organización 0**
- **200.23.18.0/23 a la organización 1**
- **200.23.20.0/22 a la organización 2**
- **200.23.24.0/21 a la organización 3**

Además, el ISP B dispone del bloque de direcciones IP 199.31.0.0/16. Supongamos un router C de internet, con una interfaz hacia el ISP A y otra al ISP B, además de otras interfaces hacia otros ISPs. Contesta razonando las respuestas.

A. **Indica las entradas en formato dirección base/máscara que tendrá el router C para encaminar paquetes con destinos pertenecientes a los ISP A y B. Indica también la máscara en formato máscara.**
B. **Supongamos ahora que la organización 1 cambia al ISP B, pero sin cambiar sus direcciones IP asignadas, ¿qué entradas tendrá ahora el router C?**
C. **Indica cómo determina el router C la entrada apropiada para un datagrama con destino a 200.23.19.160 en ambos casos**

Organización 0: 200.23.[0001 000X].X

Organización 1: 200.23.[0001 001X].X

Organización 2: 200.23.[0001 01XX].X

Organización 3: 200.23.[0001 1XXX].X

Se pueden agregar todas en 200.23.[0001 XXXX].X → 200.23.16.0/20.

En formato máscara: 255.255.[1111 0000].0 → 255.255.240.0

Dirección	Máscara	Interfaz
200.23.16.0/20	255.255.240.0	A
199.31.0.0/16	255.255.0.0	B
0.0.0.0	255.255.255.255	X

B: Se añade otra entrada, más restrictiva, para asegurar que las direcciones de la organización 1 vaya a B.

En formato máscara: 255.255.[1111 1110].0 → 255.255.254.0

Dirección	Máscara	Interfaz
200.23.16.0/20	255.255.120.0	A
199.31.0.0/16	255.255.0.0	B
200.23.18.0/23	255.255.254.0	B
0.0.0.0	255.255.255.255	X

C: 200.23.19.160 → 200.23.19.[0001 0011].160

Comprueba las entradas con máscara más restrictiva (prefijo mayor) primero.

En el primer caso comprueba con 200.23.16.0/20. Deben coincidir los 20 primeros bits: 200.23.[0001 XXXX].X. Dado que coinciden, se envía por la interfaz A.

En el segundo caso la entrada añadida es más restrictiva, se comprueba con esta primero. Deben coincidir los 23 primeros bits: 200.23.[0001 001X].X. Dado que coinciden, se envía por la interfaz B.

6. Explica brevemente qué es el control de flujo y cómo funciona en TCP.

El control de flujo es el mecanismo que permite al receptor indicar al emisor el ritmo al que puede recibir datos evitando la congestión que lleva a pérdidas de datos.

Para esto, el receptor actualiza con cada envío de datos el tamaño de la ventana de congestión, que es la cantidad máxima de datos que el emisor podrá enviar de una vez antes de requerir una confirmación (ACK).

Se distinguen distintas etapas en TCP en las que la variación del tamaño de la ventana de congestión es distinto. Primero se fija a un valor inicial de 1 MSS (maximum segment size) y luego se va duplicando con cada envío de datos (inicio lento). Esto continúa hasta alcanzar un umbral determinado.

Cuando se alcanza este umbral se pasa a 'incremento aditivo' (AIMD). Consiste en aumentar por 1 MSS el tamaño de la ventana en cada ciclo. Esto continúa hasta que se produzca alguna pérdida de datos.

En caso de una pérdida se actuará de forma distinta según la causa. Si fue producida por temporizador se vuelve a empezar con inicio lento y el tamaño de la ventana vuelve a 1MSS.

Si la pérdida se ha detectado por 3 ACKs duplicado, se pasa a recuperación rápida: se reduce a la mitad el tamaño de VS y el umbral previamente menciionado. Además, se le suma al tamaño de VSS 1 MSS por cada ACK duplicado recibido. Una vez solucionada la pérdida, el tamaño de la ventana vuelve a reducirse al de la ventana (se quita lo sumado por ACKs) y se continúa con incremento aditivo.