Redes de Computadoras

Solución Examen – 28 de julio de 2021

Nota

El examen se tomó en modalidad virtual a través de la plataforma EVA.
Consistió de tres partes, cada una de 50 minutos. Cada parte estuvo separada por un descanso de 5 minutos.
La primera parte consistió en 5 preguntas de un set de preguntas que se sortearon de forma aleatoria para cada estudiante.
Las partes 2 y 3, consistieron en ejercicios también con diferentes versiones. Este documento contiene solo una de estas versiones para cada parte.

Parte 1

Pregunta 1 (8 puntos)

V1

Suponga que tiene una red con dos hosts A y B con direcciones IP *IP_A* e *IP_B* pero (debido a un error) ambos tienen la misma dirección MAC *MAC_A*. A y B pertenecen a la misma subred y están conectados a diferentes interfaces de un mismo conmutador de capa de enlace (*switch*).

Describa los problemas que pueden aparecer cuando los hosts A y B se conecten con otros hosts en la misma subred. Analice en particular la incidencia sobre las diferentes capas (2, 3 y 4) y tenga en cuenta en su explicación el proceso de auto-aprendizaje (*self-learning*) de los conmutadores.

Solución:

Cuando otro host quiera comunicarse con A (o B) ensamblará una trama con la dirección MAC_A y con IP_A (o IP_B). Dependiendo del estado de las tablas del switch (actualizadas mediante el auto-aprendizaje), el paquete será entregado a A o B, pero no a ambos (salvo en el caso de borde cuando el switch no conoce a A o B inicialmente).

Por consiguiente solamente uno de los hosts recibirá la trama, independientemente de si es el destinatario de capa superior o no. La capacidad de auto-aprendizaje del switch hará que el puerto destino de la configuración cambie a lo largo del tiempo, en función de su implementación y el tráfico generado desde A o B. Desde el punto de vista de capa 2, podrán haber pérdidas de paquetes en ráfagas (considerando el emisor y receptor finales de la comunicación).

Desde el punto de vista de capa 3, los paquetes entregados correctamente serán procesados, mientras que los entregados al destinatario inválido, serán descartados, dado que en los

cabezales de capa 3 las direcciones IP equivocadas serán descartadas y las esperadas serán aceptadas. El clasificador de esta capa no confundirá tráfico. Desde el punto de vista de capa 4, recibirá ráfagas de tráfico, y dependerá de éstas si es capaz de realizar su procesamiento o no. TCP podrá (o no) recuperarse en función del nivel de pérdidas, y UDP realizará su procesamiento, dejando en manos de la aplicación la recuperación ante fallos.

V2

a) ¿Para qué se utilizan las direcciones MAC? Describa el formato de las direcciones MAC de Ethernet, y escriba la dirección de *broadcast*.

b) Suponga un switch de 8 puertos, con 8 *hosts* conectados directamente a él, formando una LAN. ¿Cuántos dominios de *broadcast* y cuantos de colisión existen en este escenario? Justifique.

c) En el escenario anterior, suponga que un host envía un mensaje *ARP Request*. ¿Que *hosts* recibirán el mismo?. ¿Cuántos responderán el mensaje?

SOLUCION

a) Las direcciones de la capa de enlace se asignan a sus adaptadores (es decir, sus interfaces de red). En la mayoría de las redes LAN (incluyendo las redes Ethernet y las LAN inalámbricas 802.11) la dirección MAC tiene 6 bytes de longitud, lo que nos da 248 posibles direcciones MAC. Suelen expresarse en notación hexadecimal, indicándose cada byte de la dirección mediante una pareja de números hexadecimales. La dirección de broadcast se escribe como FF:FF:FF:FF:FF:FF.
b) Los switches trabajan en modo store and forward, generando un dominio de colisión para cada una de sus interfaces, de manera que tenemos entonces 8 en total para este escenario. En cuanto a dominio de broadcast, solo tenemos uno, ya que una trama enviada a la dirección de destino de broadcast llegará a todos los host conectados a ese switch.
c) El mensaje ARP Request es enviado a la dirección de broadcast MAC, de manera que por definición lo recibirán los 7 restantes host conectados a la red. Todos lo procesarán, pero solo uno de ellos responderá, el que tenga configurada la dirección IP por la que el origen consultó.

Pregunta 2 (8 puntos)

V1

El protocolo TCP calcula un estimado del RTT (*Round Trip Time*) mediante la siguiente fórmula: RTT= (1-α) . RTTestimado + α . RTTmuestra.

a) Explique la razón por la que TCP necesita calcular un estimado del RTT.

b) Explique cada uno de los parámetros de la fórmula.

Solución

a) Cada una de las entidades que establecen una sesión TCP, intenta estimar dinámicamente el tiempo entre ida y vuelta mientras vive la sesión y de esa forma calcular el tiempo de
retransmisión. El tiempo de retransmisión es el tiempo que se espera por la recepción de un mensaje de confirmación antes de considerar un paquete como perdido.

a) RTT: es el tiempo transcurrido desde que se envía un segmento con determinado número de
secuencia hasta que se recibe un acknowledgment que incluye a dicho número.

El RTTmuestra para un segmento es la cantidad de tiempo medida desde que se envía el segmento (o sea, se pasa a la capa de red) hasta que se recibe el correspondiente reconocimiento (ACK).
En la mayoría de las implementaciones TCP ésto no se hace para cada uno de los segmentos
transmitidos, sino que se realiza para al menos uno de los segmentos transmitidos (que no sea
una retransmisión) y todavía no reconocidos; sin ser muy precisos, podemos decir que se
realiza cada RTT segundos.

RTTestimado: tiempo a partir del cual TCP busca calcular un RTT promedio (suavizado sería una mejor calificación) de los RTTmuestra buscando evitar valores atípicos de éstos. Los valores de RTTmuestra fluctuarán de un segmento a otro a causa de la congestión en la red y a la variación de la carga de los sistemas terminales, lo que puede determinar que tengamos valores atípicos mencionados.

V2

a) En los protocolos de transporte confiables, ¿con qué objetivo se utilizan los mecanismos de temporización, número de secuencia y ventana?
b) En un protocolo de tipo *Selective Repeat* de ventana deslizante, ¿cual es la relación de tamaño entre la ventana y el número de secuencia? ¿por qué?

Solución:

a) Temporizador Se emplea para detectar el fin de temporización y retransmitir un paquete, posiblemente porque el paquete (o su mensaje ACK correspondiente) se ha perdido en el canal. Puesto que se puede producir un fin de temporización si un paquete está retardado pero no perdido (fin de temporización prematura), o si el receptor ha recibido un paquete pero se ha perdido el correspondiente ACK del receptor al emisor, puede ocurrir que el receptor reciba copias duplicadas de un paquete. Número de secuencia Se emplea para numerar secuencialmente los paquetes de datos que fluyen del emisor hacia el receptor. Los saltos en los números de secuencia de los paquetes recibidos permiten al receptor detectar que se ha perdido un paquete. Los paquetes con números de secuencia duplicados permiten al receptor detectar copias duplicadas de un paquete. Ventana El emisor puede estar restringido para enviar únicamente paquetes cuyo número de secuencia caiga dentro de un rango determinado. Permitiendo que se transmitan varios paquetes aunque no estén todavía reconocidos, se puede incrementar la tasa de utilización del emisor respecto al modo de operación de los protocolos de parada y espera. Veremos brevemente que el tamaño de la ventana se puede establecer basándose en la capacidad del receptor para recibir y almacenar en buffer los mensajes, o en el nivel de congestión de la red, o en ambos parámetros.
b) el tamaño de la ventana tiene que ser menor o igual que la mitad del tamaño del espacio de números de secuencia en los protocolos SR. Lo que motiva esa decisión es el dilema del receptor de los protocolos SR con ventanas demasiado grandes: ¿un nuevo paquete o una retransmisión?

Pregunta 3 (8 puntos)

V1

a) Describa las principales funcionalidades del plano de control y el plano de datos de la capa de red.

b) Considerando el protocolo OSPF, ¿Pertenece a alguno de estos planos? Justifique

SOLUCIÓN

a)La capa de red puede descomponerse en dos partes que interaccionan mutuamente: el plano de datos y el plano de control. El plano de datos implementa la funcionalidad de reenvío (forwarding). El reenvío hace referencia a la acción local que realiza un router al transferir un paquete desde una interfaz de un enlace de entrada a una interfaz del enlace de salida

apropiado. Esto es, cuando un paquete llega al enlace de entrada de un router, este tiene que pasar el paquete al enlace de salida apropiado.

El plano de datos implementa la funcionalidad de enrutamiento (routing). El enrutamiento hace referencia al proceso que realiza la red en conjunto para determinar las rutas extremo a extremo que los paquetes siguen desde el origen al destino. Esto permite a la capa de red determinar la ruta o camino que deben seguir los paquetes a medida que fluyen de un emisor a un receptor. Los algoritmos que calculan estas rutas se conocen como algoritmos de enrutamiento.

b) Se puede decir que el protoloco OSPF pertenece al plano de control, dado que es un algoritmo que permite calcular las rutas que los paquetes a medida que fluyen de un emisor a un receptor.

V2

a) Explique brevemente en que se diferencian los algoritmos de tipo estado de enlaces (l*ink-state*) y vector de distancias (*distance vector*).

b) Suponga que tiene una red de gran porte donde un router funciona mal y difunde información incorrecta. Exponga como se comportaría la red (respecto a la robustez) si utilizáramos un protocolo de tipo estado de enlaces versus si utilizáramos uno de vector de distancias.

SOLUCIÓN

a) En un algoritmo de estado de enlaces, la topología de la red y el coste de todos los enlaces son conocidos; es decir, están disponibles como entradas para el algoritmo LS mientras que en uno de vector de distancias solo necesita saber quienes son sus vecinos, los costos hacia ellos y sus vectores de distancia. Respecto a la información que se intercambia, los LS inundan con su información local (vecinos, costos) mientras que los DV intercambian vectores de distancias. Dados estos intercambios, los LS ejecutan algoritmos como el algoritmo de Dikjstra (que tiene como entrada la info de toda la red), mientras que uno DV se vale de las ecuaciones de Bellman-Ford para calcular las rutas de costo mínimo. Finalmente, se puede comparar la velocidad de convergencia de estos algoritmos. Un algoritmo de LS de O(n^2), requiere O(nE) mensajes (n nodos, E links) y puede tener oscilaciones, mientras que en un DV el tiempo de convergencia varía y pueden contar con problemas como el de conteo a infinito.

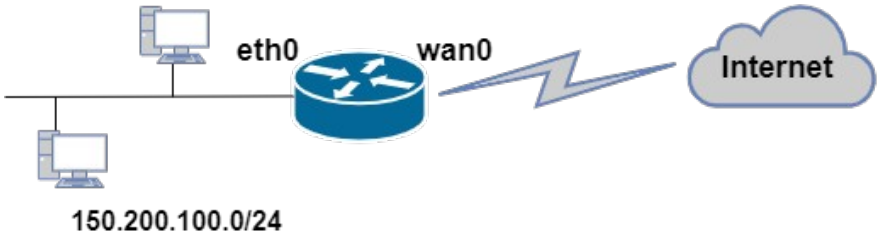
b) Con el algoritmo de estado de enlaces, un router podría difundir un coste incorrecto para uno de sus enlaces conectados (pero no para los otros). Un nodo también podría corromper o eliminar cualquier paquete recibido como parte de un mensaje de difusión LS. Pero, con el algoritmo LS, un nodo solo calcula su propia tabla de reenvío, mientras que otros nodos realizan cálculos similares por sí mismos. Esto significa que los cálculos de rutas son hasta cierto punto independientes en LS, proporcionando un mayor grado de robustez. Con el

algoritmo de vector de distancias, un nodo puede anunciar rutas de coste mínimo incorrectas a uno o a todos los destinos. En un sentido más general, observamos que, en cada iteración, los cálculos de un nodo con el algoritmo de vector de distancias se pasan a sus vecinos y luego, indirectamente, al vecino del vecino en la siguiente iteración. En este sentido, con el algoritmo de vector de distancias, un cálculo de nodo incorrecto puede difundirse a través de toda la red.

Pregunta 4 (8 puntos)

V1

Considere la red de la figura.



El router permite configurarle reglas en su interfaz *eth0* de manera de establecer qué paquetes pueden entrar o salir del router a través de ella. Dichas reglas se pueden establecer especificando condiciones a nivel de capa de red (direcciones IP origen y destino y, tipos y subtipos de mensajes ICMP) y de capa de transporte (protocolos TCP y UDP y, puertos de origen y destino).
Las reglas son de la siguiente forma (no es necesario configurar todos los campos):
Regla entrante(saliente)/eth0
IP origen:
IP destino:
Tipo ICMP:
Subtipo ICMP:
Protocolo:
Puerto/s origen:
Puerto/s destino:

Especifique las reglas a configurar en la interfaz *eth0* tanto para el tráfico entrante como para el tráfico saliente al router, de manera que se permita la ejecución del comando traceroute basado en UDP desde todo *host* de la red LAN dirigido a cualquier *host* en Internet. Cuando quiera especificar la condición de “cualquiera” (IP o puerto) indíquelo con la palabra *any*.

Solución

Regla in/eth0

IP origen: 150.200.100.0/24
IP destino: any
Tipo ICMP: N/A
Subtipo ICMP: N/A
Protocolo: UDP
Puerto/s origen: any “muy alto”
Puerto/s destino: any “muy alto”

“muy alto”: número de puerto que ofrece una muy alta probabilidad de que no responda. Para fijar el concepto, mayor a 34000.

Reglas out/eth0

IP origen: any
IP destino: 150.200.100.0/24
Tipo ICMP: Time Exceeded (11)
Subtipo ICMP: N/A
Protocolo: N/A
Puerto origen: N/A
Puerto destino: N/A

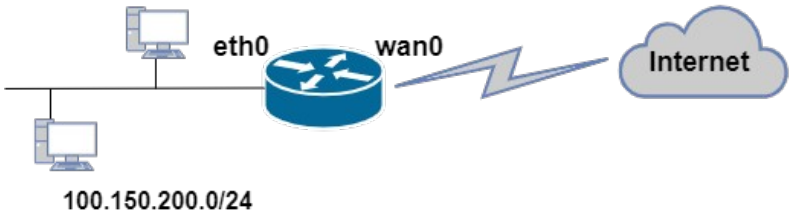
Regla out/eth0

IP origen: any
IP destino: 150.200.100.0/24
Tipo ICMP: Destination Unreachable (3)
Subtipo ICMP: Port Unreachable (3)
Protocolo: N/A
Puerto origen: N/A
Puerto destino: N/A

V2

Considere la red de la figura.

Redes de Computadoras



El router permite configurarle reglas en su interfaz *eth0* de manera de establecer qué paquetes pueden entrar o salir del router a través de ella. Dichas reglas se pueden establecer especificando condiciones a nivel de capa de red (direcciones IP origen y destino y, tipos y subtipos de mensajes ICMP) y de capa de transporte (protocolos TCP y UDP y, puertos de origen y destino).
Las reglas son de la siguiente forma (no es necesario configurar todos los campos):
Regla entrante(saliente)/eth0
IP origen:
IP destino:
Tipo ICMP:
Subtipo ICMP:
Protocolo:
Puerto/s origen:
Puerto/s destino:

Especifique las reglas a configurar en la interfaz *eth0* tanto para el tráfico entrante como para el tráfico saliente al router, de manera que se permita la ejecución del comando traceroute basado en ICMP desde todo *host* de la red LAN dirigido a cualquier *host* en Internet. Cuando quiera especificar la condición de “cualquiera” (IP o puerto) indíquelo con la palabra *any*.

Solución

Regla in/eth0

IP origen: 100.150.200.0/24
IP destino: any
Tipo ICMP: Echo (8)
Subtipo ICMP: N/A
Protocolo: N/A
Puerto/s origen: N/A
Puerto/s destino: N/A

Reglas out/eth0

IP origen: any
IP destino: 100.150.200.0/24
Tipo ICMP: Time Exceeded (11)
Subtipo ICMP: N/A
Protocolo: N/A
Puerto origen: N/A
Puerto destino: N/A

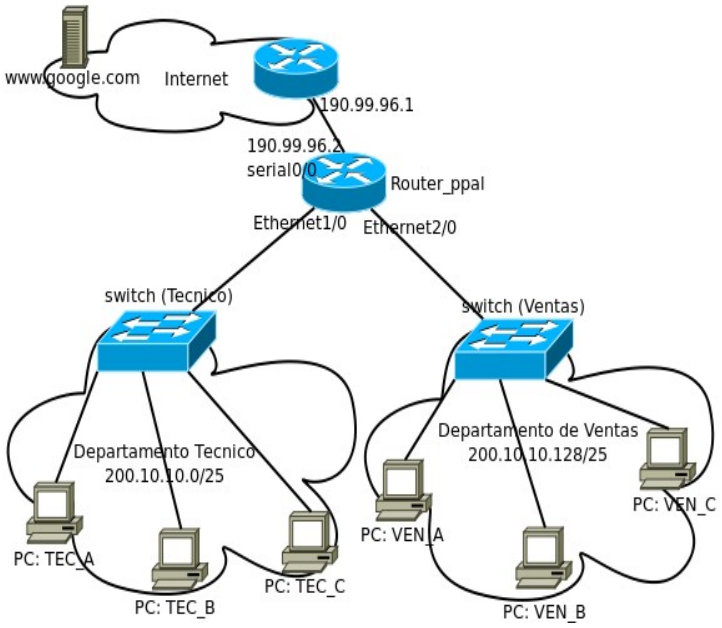
Regla out/eth0

IP origen: any
IP destino: 100.150.200.0/24
Tipo ICMP: Echo Reply (0)
Subtipo ICMP: N/A
Protocolo: N/A
Puerto origen: N/A
Puerto destino: N/A

Pregunta 5 (8 puntos)

V1

Sea el siguiente esquema de red:



a) Proponga direcciones IP para las interfaces *Ethernet1/0* y *Ethernet2/0* justificando su elección.

b) Indique la tabla de *forwarding* del router_ppal de forma que todos los equipos tengan acceso a Internet. Explique cada entrada de la tabla.

Solución

a) Ethernet1/0 pertenece a la subred del departamento técnico por lo que hay que elegir una dirección dentro del rango dado. Por ejemplo: 200.10.10.1
Ethernet3/0 pertenece a la subred del departamento de ventas por lo que hay que elegir una dirección dentro del rango dado. Por ejemplo: 200.10.10.129

b) La tabla del router sería:

Red destino	gateway	interfaz
200.10.10.0/25	DC	Ethernet1/0
200.10.10.12/25	DC	Ethernet2/0
190.99.96.0/30	DC	serial0/0
default	190.99.96.1	serial0/0

Las primeras tres entradas corresponden a las redes directamente conectadas, donde se indica la interfaz de salida.
La ultima entrada es la ruta default, lo que indica que todo destino que no matchee con las anteriores seguirá esa ruta. En ese caso se indica que el siguiente salto en la ruta es 190.99.96.1 y se sale por la interfaz serial0/0

Parte 2 (30 puntos)

Considere dos equipos E1 y E2 conectados a la misma red donde el equipo E1 establece una conexión TCP con el equipo E2.
Para sus respuestas de las partes b, c y d, considere el formato de los cabezales IP y TCP vistos en el curso, pero especificando únicamente los siguientes campos: *Source Address, Destination Address, Source Port, Destination Port, Sequence Number, Acknowledgement Number y flags (ACK, SYN, FIN)*.

Se pide:
a) Indique que datos de la conexión (a nivel de capa de red y transporte) son necesarios para identificar de forma unívoca a esta conexión. Asigne valores para estos datos.

b) Suponenido que E1 y E2 eligen como número inicial de secuencia 200 y 1500 respectivamente, indique los cabezales IP y TCP intercambiados para el establecimiento de la conexión (*three-way handshake*).

c) Suponga que en un cierto momento (con la conexión ya establecida), E2 envía a E1 un segmento con la siguiente información en el cabezal TCP, *Sequence Number* = 6300 *Acknowledgement Number* = 3100 conteniendo 1200 Bytes de datos. Suponga además que E1 tiene 500 Bytes para enviar a E2 que puede enviarlos en un único segmento TCP. Responda las siguientes preguntas:

1. ¿Que información contiene el cabezal del segmento de *acknowledgment* enviado por E1 a E2, para notificar su recepción?
2. Si E2 tiene a continuación 1100 Bytes más para enviar a E1, ¿qué información contienen los campos del encabezado del siguiente segmento para su envío?
3. Suponiendo que se pierde el segmento de *acknowledgment* propuesto en la parte c.1, y el enviado en c.2 llega correctamente, i) ¿cambia en algo su respuesta de la parte c.2? ii) ¿qué número de secuencia esperará E1 después de la recepción del segmento enviado en c.2?

d) Después de enviados y aceptados todos los datos indicados en la parte c), E1 desea cerrar la conexión establecida en la parte b), indique el intercambio de segmentos entre los dos equipos.

Solución:

a) Para la identificación de una conexión TCP en forma unívoca, se debe tener los siguientes parámetros definidos:

- Source Address, asigno E1 IP 200.40.40.40
- Destination Address, asigno E2 IP 100.50.50.50
- Source Port, asingo E1 port 36444
- Destination Port, asigno E2 port 443

Con éstos parámetros queda identificada la conexión TCP. Según el sentido (cliente a servidor o servidor a cliente) las parejas {src-address,src-port} y {dst-address,dst-port} pueden ser intercambiadas según el sentido.

b) El proceso de three-way handshake se realiza con el intercambio de los siguientes tres datagramas:

	Src IP	Dst IP	Src port	Dst port	Seq. number	Ack. Number	flags
1	200.40.40.40	100.50.50.50	36444	443	200	0	SYN
2	100.50.50.50	200.40.40.40	443	36444	1500	201	SYN, ACK
3	200.40.40.40	100.50.50.50	36444	443	201	1501	ACK

c)

NOTA: A modo de completitud, en las respuestas se muestra el segmento descrito en la letra y

se marca con una flecha (-->) la respuesta esperada.

c.1:

	Src IP	Dst IP	Src port	Dst port	Seq. number	Ack. Number	Flags	Cantidad datos
	100.50.50.50	200.40.40.40	443	36444	6300	3100		1200
-->	200.40.40.40	100.50.50.50	36444	443	3100	7500	ACK	500

c.2:

	Src IP	Dst IP	Src port	Dst port	Seq. number	Ack. Number	Flags	Cantidad datos
	100.50.50.50	200.40.40.40	443	36444	6300	3100		1200
	200.40.40.40	100.50.50.50	36444	443	3100	7500	ACK	500
-->	100.50.50.50	200.40.40.40	443	36444	7500	3600	ACK	1100

c.3:

	Src IP	Dst IP	Src port	Dst port	Seq. number	Ack. Number	Flags	Cantidad datos
	100.50.50.50	200.40.40.40	443	36444	6300	3100		1200
X	~~200.40.40.40~~	~~100.50.50.50~~	~~36444~~	~~443~~	~~3100~~	~~7500~~	~~ACK~~	~~500~~
-->	100.50.50.50	200.40.40.40	443	36444	7500	3100		1100

Como el segmento de la parte c1, con el ACK y los datos de E1 no llegaron, el número de secuencia no cambia pero cambia el número de ACK, dado que el último byte recibido correctamente es el 3100.
Por lo tanto, la respuesta dada en c2 cambia, el numero de ACK debe ser 3100 y la flag de ACK debe estar apagada.

Como los segmentos de datos de E2 no se perdieron, E1 espera por el num de sec 8600 (7500 + 1100)

d)
La desconexión es iniciada por E1.

	Src IP	Dst IP	Src port	Dst port	Seq. number	Ack. Number	Flags
1	200.40.40.40	100.50.50.50	36444	443	3600	8600	FIN
2	100.50.50.50	200.40.40.40	443	36444	8600	3601	ACK

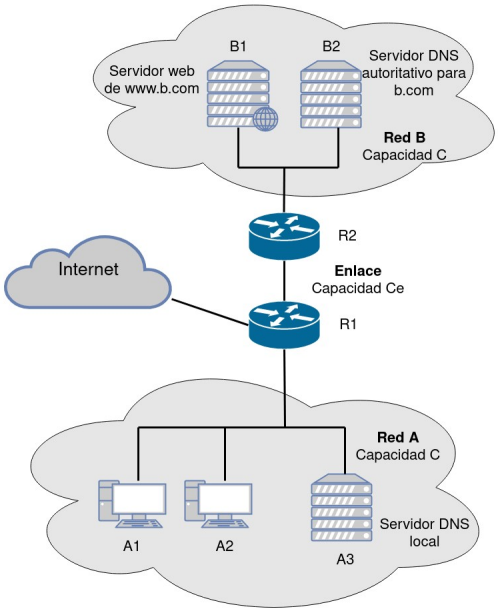
3	100.50.50.50	200.40.40.40	443	36444	8600	3601	FIN
4	200.40.40.40	100.50.50.50	36444	443	3601	8601	ACK

Se intercambian 4 paquetes para el cierre:

1. E1 envía FIN con el número de secuencia y ACK de los últimos bytes enviados (seq 3600 y ACK 8600)
2. E2 envía FIN y ACK del del FIN recibido. Se incrementa en uno el ACK
3. E2 envía FIN con el número de secuencia y ACK de los últimos bytes enviados (seq 8600 y ACK 3601)
4. E1 confirma la recepción del FIN y manda ACK, incrementando ACK number en 1

Parte 3 (30 puntos)

Considere la siguiente figura.



La red A posee dos hosts, A1 y A2 y un servidor DNS local configurado para realizar consultas iterativas.
La red B posee un servidor web donde se aloja www.b.com y un servidor DNS autoritativo para el dominio b.com.

Se pide:

a) Suponga que un usuario en A1 ingresa la URL http://www.b.com/video.mp4 en un

navegador con el objetivo de obtener un archivo de video desde www.b.com y asuma que es el primer pedido a www.b.com realizado desde la red A.
Describa toda la secuencia de mensajes **de capa de aplicación** necesaria para llevar a cabo la transferencia de `video.mp4` desde que se ingresa la URL en el navegador hasta que el archivo es recibido completamente.
Considere solo los mensajes observados en la Red A y explicite origen, destino y contenido de los mensajes. Por ejemplo: "*El equipo A2 envía un mensaje HTTP al equipo A1 con el método POST requiriendo el recurso pepe.html*".

b) Si ahora asume que antes del pedido realizado por A1 la máquina A2 ya había realizado el mismo pedido, ¿que diferencias se observarían con respecto a la respuesta dada en la parte a)?

c) Calcule el tiempo necesario para completar todos los pasos de la pregunta anterior (b) **solo** considerando la transmisión de los mensajes **de capa de aplicación**. Muestre el resultado en función de **N, P, C y Ce** (ver figura y nota).

Nota: Asuma que:

- Todos los mensajes de capa de aplicación excepto aquellos que transportan el video tienen tamaño **P Bytes** (y son transportados en un único datagrama IP).
- El mensaje que contiene el archivo de video tiene tamaño **N Bytes** (y es transportado en varios datagramas IP de tamaño P Bytes)**.**
- Los retardos de propagación, procesamiento y cola son pequeños y pueden ser despreciados.
- Las capacidades están expresadas en **bits por segundo** y se cumple que **C=10*Ce** .

Solución:

a) Los pasos seguidos para la obtener la URL solicitada tiene los siguientes pasos:

- Obtención de la IP de www.b.com
- Obtención del archivo video.mp4

La primera parte se resuelve de la siguiente forma:

id	Protocolo App.	Host origen	Host Destino	observaciones
1	DNS	A1	Dns_local	Envia query de registro A de www.b.com
2	DNS	Dns_local	Dns_root	Envia query de registro A de www.b.com
3	DNS	Dns_root	Dns_local	Envia response de registro NS del TLD com (Dns_TLD) con el correspondiente registro A
4	DNS	Dns_local	Dns_TLD	Envia query de registro A de www.b.com
5	DNS	Dns_TLD	Dns_local	Envia response de registro NS de autoritativo b.com

				(Dns_b) con el correspondiente registro A
6	DNS	Dns_local	Dns_b	Envia query de registro A www.b.com
7	DNS	Dns_b	Dns_local	Envía response de registro A www.b.com (IP_wwwb)
8	DNS	Dns_local	A1	Envía response de registro A www.b.com (IP_wwwb)

Ahora que contamos con la dirección IP de www.b.com, realizamos la solicitud del archivo video.mp4

La primera parte se resuelve de la siguiente forma:

id	Protocolo App.	Host origen	Host Destino	observaciones
1	HTTP	A1	IP_wwwb	Envia solicitud de la página (previo existe conexión) GET /video.mp4 HTTP/1.1 Además incluye parámetros como User-Agent, Host, etc
2	HTTP	IP_wwwb	A1	Envia respuesta, HTTP/1.1 200 OK Content-Length: (tamaño archivo video.mp4) se envía el contenido del archivo

b) Si la IP se encuentra en cache del servidorlocal, los pasos de la consulta DNS de parte a) numerados del 2 al 7 no se realizan, dado que no debe realizar consulta para obtener la respuesta, porque ya tiene la respuesta.

c)
Datos proporcionados:
N – largo en bytes del mensaje con el video,
P – tamaño en bytes de mensajes (exceptuando los de datos)
T – retardo a nodos en internet en segundos (no se utilizará)
C y Ce – capacidad de enlaces en bits/segundo

Tiempo para DNS:

tiempo	Host origen	Host Destino	observaciones
P*8/C	A1	Dns_local	Envia query de registro A www.b.com
P*8/C	Dns_local	A1	Envía response de registro A www.b.com (IP_wwwb)

Total tiempoDNS = 2*P*8/C

El tiempo dependerá fundamentalmente del enlace mas lento.
En los casos que se envía un único datagrama IP, como los routers realizan *store and forward* se debe sumar el tiempo que tarda en transmitir por cada uno de los 3 enlaces.

En el caso del mensaje con el video, el cual está formado por varios datagramas, el tiempo estará dado por el enlace mas lento, en el cual se encolarán los paquetes que llegan del enlace rápido (debido a que C=10*Ce). A esto se le debe sumar el tiempo que tarda el primer datagrama en llegar a ese enlace mas lo que tarda el último datagrama en ser transmitido por el enlace rápido final.

Tiempo HTTP:

id	Host origen	Host Destino	observaciones
P*8/C + P*8/Ce + P*8/C	A1	IP_wwwb	Envia solicitud de la página (previo existe conexión) GET /video.mp4 HTTP/1.1 Además incluye parámetros como User-Agent, Host, etc
P*8/C + N*8/Ce + P*8/C	IP_wwwb	A1	Envia respuesta, HTTP/1.1 200 OK Content-Length: (tamaño archivo video.mp4) se envía el contenido del archivo

Total tiempoHTTP = 4*P*8/C + (P+N)*8/Ce