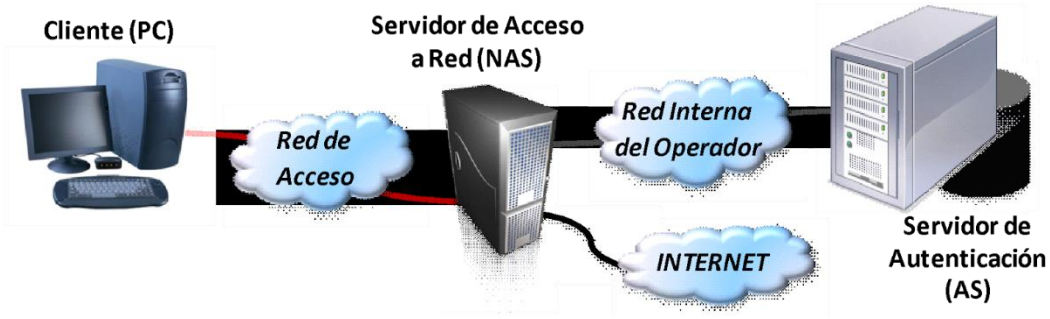
EJERCICIO 1

La figura y mensajes siguientes describen un hipotético protocolo utilizado para permitir el acceso de un cliente a Internet a través de un Servidor de Acceso a Red (NAS). El Servidor de Autenticación (AS) guarda en una base de datos las claves secretas que se solicita a los usuarios para poder acceder a Internet.



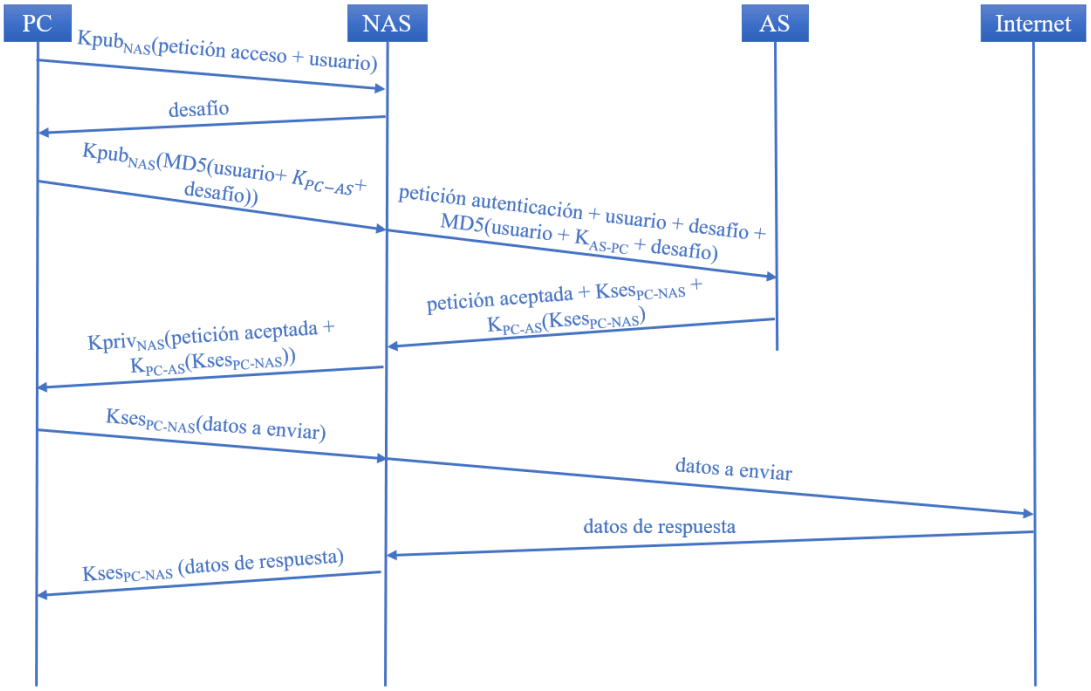
PC ->NAS: $Kpub_{NAS}$ (petición acceso + usuario)
NAS -> PC: desafío
PC -> NAS: $Kpub_{NAS}$ (MD5(usuario + K_{PC-AS} + desafío))
NAS -> AS: petición autenticación + usuario + desafío + MD5(usuario + K_{AS-PC} + desafío)
AS -> NAS: petición aceptada + $Kses_{PC-NAS}$ + $K_{PC-AS}(Kses_{PC-NAS})$ o petición rechazada
NAS -> PC: $Kpriv_{NAS}$ (petición aceptada + $K_{PC-AS}(Kses_{PC-NAS})$) o $Kpriv_{NAS}$ (petición rechazada)
PC -> NAS: $Kses_{PC-NAS}$ (datos a enviar)
NAS -> Internet: datos a enviar
Internet -> NAS: datos de respuesta
NAS -> PC: $Kses_{PC-NAS}$ (datos de respuesta)

Siendo:
$Kpub_X$ cifrado con la clave pública de X
$Kpriv_X$ cifrado con la clave privada de X
K_{X-Y} la clave secreta entre X e Y
$Kses_{X-Y}$ la clave secreta de sesión entre X e Y
MD5 una función hash

Aceptadas la disponibilidad y validez de las claves públicas involucradas en base a la existencia de una entidad superior confiable, responda razonadamente:

a. ¿Qué servicios de seguridad se proporcionan en el protocolo descrito?
b. ¿Qué debilidades presenta el esquema propuesto? En su caso, ¿cómo podrían evitarse?

El diagrama de intercambio de mensajes es:



a) Servicios de seguridad:

- **Privacidad:**
  - El PC envía todo cifrado al NAS, ya sea con su clave pública (sólo puede descifrarlo el NAS) o con la clave de sesión que ambos comparten (asignada por el AS).
  - El NAS no cifra el desafío (posible ataque por repetición si se repiten desafíos). Además, el mensaje de "petición aceptada" solo va cifrados con su clave privada, por lo que cualquiera podría descifrarlos con su clave pública (conocida por todos). Aunque los datos sensibles sí van cifrados (e.g. la clave de sesión) adecuadamente, alguien en la red de acceso podría ver que el usuario se ha conectado.
  - Igual que ocurre entre en NAS y AS, nada va cifrado para conseguir la confidencialidad. Aunque pertenece a la red interna del operador, un trabajador podría ver la clave de sesión y pensar algún tipo de ataque, e.g. suplantación del PC.
  - Por último, los datos a enviar/recibir hacia/desde Internet van cifrados con la calve de sesión entre el PC y el NAS, de forma que alguien en la red de acceso no los podría leer. Sin embargo, los datos van sin cifrar por Internet, algo lógico ya que, en general, no se sabe el destino de estos datos, si soporta cifrado, etc. (este último punto es necesario para las aplicaciones típicas, que van sin cifrar).
- **Integridad**:
  - No se utilizan compendios/resúmenes, por lo que no es posible comprobar si los datos han sido modificados.
    Nota: El uso de la función MD5() en el esquema propuesto es para no enviar la clave secreta en texto plano. No tiene nada que ver con la integridad (no es el resumen del mensaje).
- **Autenticación:**
  - El PC solicita su autenticación al NAS a través de un AS remoto. Dado que el procedimiento requiere que conozca su clave secreta compartida con el AS (K_{PC-AS}), queda autenticado al final del procedimiento.

- El NAS inicialmente no se autentica (no usa su $Kpriv_{NAS}$ para que el PC sepa que realmente es el NAS). Ahí podría ser suplantado, y el PC no lo notaría (e.g. atacante para conseguir muchos pares desafíos ↔MD5()). Sin embargo, sí se autentica al enviar la "petición aceptada" cifrada con su clave privada (sólo puedo cifrarlo el NAS). En el envío/recepción de datos tampoco se autentica (aunque debe conocer la clave de sesión).
- El NAS y el AS no se autentican entre ellos. Alguien podría suplantar al NAS y el AS no se daría cuenta, y obtendría la clave de sesión (si bien no se la podría mandar al PC ya que no conocería la $Kpriv_{NAS}$ del auténtico NAS).

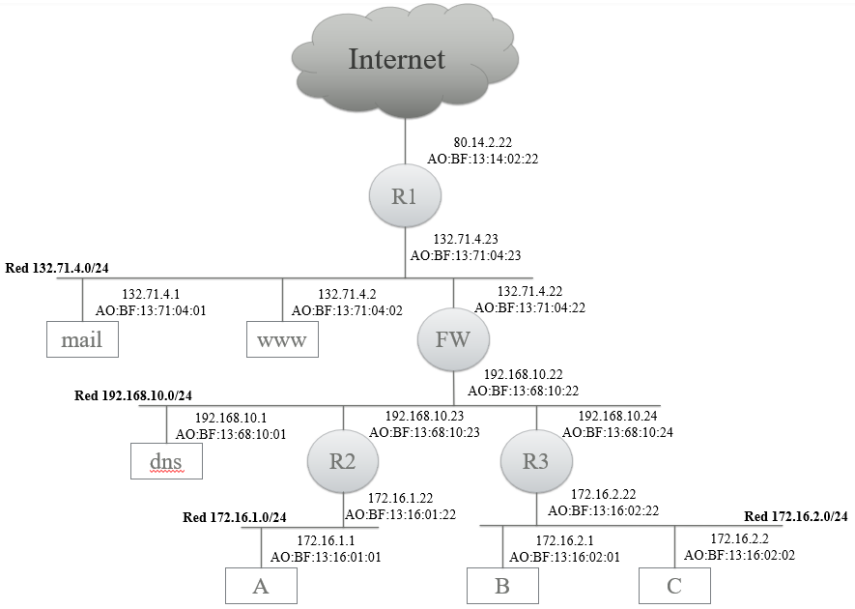
- **No repudio:**
  - Únicamente el mensaje "conexión aceptada" (o rechazada) va cifrado con la clave privada del NAS (sólo la conoce él ⟶ sólo ha podido cifrarlo él ⟶ sirve de prueba de que estuvo en la transacción). El resto de los mensajes no lleva ningún tipo de firma digital ⟶ no es demostrable que entidad los envió/recibió.
  - Así, el usuario sólo tiene "no repudio" de que su conexión ha sido aceptada o rechazada.
- **Disponibilidad:**
  - Con la información que tenemos no se puede garantizar nada (no sabemos si hay líneas de *backup*, ni la configuración de TCP/IP ante ataques, etc.).

b) Las vulnerabilidades se han ido comentado en el apartado anterior. Las soluciones para corregir las debilidades son:

- **Privacidad:** por ejemplo, cifrando todos los mensajes con la clave pública del receptor (en ese caso, AS y PC han de tener sus parejas de clave público-privada).
- **Integridad:** usando compendios del mensaje mediante funciones hash (e.g. MD5, SHA-1). Esos compendios sirven para comprobar si el mensaje ha sido modificado.
- **Autenticación:** por ejemplo, cifrando todos los mensajes con la clave privada del emisor del mensaje o con algún otro tipo de firma digital.
- **No repudio:** idem, cifrando todos los mensajes con la clave privada del emisor. Al menos, NAS y AS deberían hacerlo (el usuario ya se autentica durante el procedimiento).
- **Disponibilidad:** líneas de *backup*, igual que NAS y AS duplicado, comprobar que no se es susceptible a ataques de denegación de servicios, etc.

EJERCICIO 2

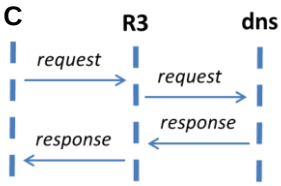
Dada la topología adjunta correspondiente a una red corporativa, en la que se especifican tanto las direcciones IP como las MAC de cada uno de los dispositivos que la forman, analice el tráfico generado al hacer un acceso de correo electrónico desde el *host* "C" al servidor "mail", especificando en una tabla, y para cada trama Ethernet generada:



a) Las direcciones hardware (físicas) origen y destino.
b) Las direcciones IP origen y destino contenidas en el paquete IP encapsulado.
c) En su caso, los puertos origen y destino de la PDU de transporte, así como los *flags* activos y campos de secuencia y ACK.
d) El tipo de mensaje de que se trata.

NOTA: *suponga todas las tablas ARP son conocidas y, por simplicidad utilice sólo los dos últimos de los 6 octetos de las direcciones físicas de las NIC (interfaces o tarjetas de red)*

1) PASO 1: Petición DNS y Respuesta

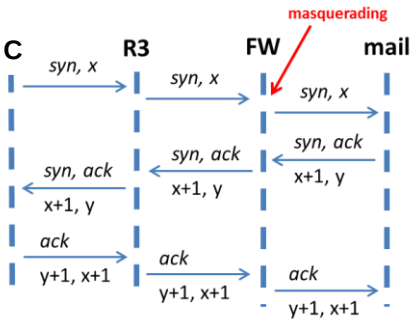


Es una **petición sobre UDP** no hay establecimiento de conexión previo.

ETH ORI.	ETH DES.	IP ORI.	IP DEST.	PORT ORI.	PORT. DES.	FLAGS	MENSAJE	COMENTARIOS
02:02 (C)	02:22 (R3)	172.16.2.2 (C)	192.168.10.1 (DNS)	(1*)	53	---	Solicitud DNS. Dominio mail	A través de R3
10:24 (R3)	10:01 (DNS)	172.16.2.2 (C)	192.168.10.1 (DNS)	(1*)	53	---	Solicitud DNS. Dominio mail	Retransmisión a DNS
10:01 (DNS)	10:24 (R3)	192.168.10.1 (DNS)	172.16.2.2 (C)	53	(2*)	---	Respuesta DNS IP de mail	A través de R3
02:22 (R3)	02:02 (C)	192.168.10.1 (DNS)	172.16.2.2 (C)	53	(2*)	---	Respuesta DNS IP de mail	Retransmisión a B

(1) Asignado por el S.O.* *(2*) Puerto elegido en (1*)*

2) PASO 2: Establecimiento conexión TCP



SMTP: protocolo de la capa de aplicación que va sobre TCP en el puerto 25.

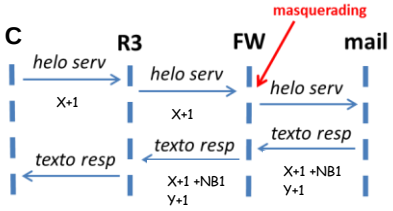
Masquerading es una traducción de IPs entre subredes. Hay que hacerlo para poder salir a la zona de direcciones públicas de la red

ETH ORI.	ETH DES.	IP ORI.	IP DEST.	PORT ORI.	PORT. DES.	FLAGS	MENSAJE	COMENTARIOS
02:02 (C)	02:22 (R3)	172.16.2.2 (C)	132.71.4.1 (mail)	(1*)	25	SYN Nº sec: x(3*)	Solicitud estab. TCP a mail	A través de R3
10:24 (R3)	10:22 (FW)	172.16.2.2 (C)	132.71.4.1 (mail)	(1*)	25	SYN Nº sec: x(3*)	Solicitud estab. TCP a mail	Retransmisión a FW
04:22 (FW)	04:01 (mail)	132.71.4.22 (FW)	132.71.4.1 (mail)	(5*)	25	SYN Nº sec: x(3*)	Solicitud estab. TCP a mail	*Masquerading* (4*) FW entrega a mail
04:01 (mail)	04:22 (FW)	132.71.4.1 (mail)	132.71.4.22 (FW)	25	(5*)	SYN, ACK Nº acuse: x+1, Nº sec: y	Aceptación y estab. en el otro sentido	mail hacia FW
10:22 (FW)	10:24 (R3)	132.71.4.1 (mail)	172.16.2.2 (C)	25	(2*)	SYN, ACK Nº acuse: x+1, Nº sec: y	Aceptación y estab. en el otro sentido	Deshace Masquerading (4*) FW retransm. a R3
02:22 (R3)	02:02 (C)	132.71.4.1 (mail)	172.16.2.2 (C)	25	(2*)	SYN, ACK Nº acuse: x+1, Nº sec: y	Aceptación y estab. en el otro sentido	R3 retransmisión a C
02:02 (C)	02:22 (R3)	172.16.2.2 (C)	132.71.4.1 (mail)	(2*)	25	ACK Nº acuse: y+1, Nº sec: x+1	Aceptación en el otro sentido	A través de R3
10:24 (R3)	10:22 (FW)	172.16.2.2 (C)	132.71.4.1 (mail)	(2*)	25	ACK Nº acuse: y+1, Nº sec: x+1	Aceptación en el otro sentido	Retransmisión a FW
04:22 (FW)	04:01 (mail)	132.71.4.22 (FW)	132.71.4.1 (mail)	(5*)	25	ACK Nº acuse: y+1, Nº sec: x+1	Aceptación en el otro sentido	Masquerading (4*) FW entrega a mail

(1) Asignado por el S.O.* *(2*) Puerto elegido en (1*)* *(3*) Num. Aleatorio elegido por el emisor*

(4) FW al hacer masquerading mapea*
[IP intranet, puerto host intranet] -> [IP pública FW, puerto libre en FW]

(5) Puerto elegido por FW en (4*)*



3) PASO 3: Acceso a correo electrónico

SMTP -> sobre TCP en el puerto 25.

Masquerading es una traducción de IPs entre subredes.

… Se podrían enviar más mensajes

ETH ORI.	ETH DES.	IP ORI.	IP DEST.	PORT ORI.	PORT. DES.	FLAGS	MENSAJE	COMENTARIOS
02:02 (C)	02:22 (R3)	172.16.2.2 (C)	132.71.4.1 (mail)	(2*)	25	x+1	helo servidor	Conexión inicial a servidor SMTP. A través de R3
10:24 (R3)	10:22 (FW)	172.16.2.2 (C)	132.71.4.1 (mail)	(2*)	25	x+1	helo servidor	Retransmisión a FW
04:22 (FW)	04:01 (mail)	132.71.4.22 (FW)	132.71.4.1 (mail)	(5*)	25	x+1	helo servidor	Masquerading (4*) FW entrega a mail
04:01 (mail)	04:22 (FW)	132.71.4.1 (mail)	132.71.4.22 (FW)	25	(5*)	ACK x+1+NB(helo) y+1	texto respuesta serv.	mail hacia FW
10:22 (FW)	10:24 (R3)	132.71.4.1 (mail)	172.16.2.2 (C)	25	(2*)	ACK x+1+NB(helo) y+1	texto respuesta serv.	Deshace Masquerading (4*) FW retransm. a R3

02:22 (R3)	02:02 (C)	132.71.4.1 (mail)	172.16.2.2 (C)	25	(2*)	ACK x+1+NB(helo) y+1	texto respuesta servidor	R3 retransmisión a C

(2) Puerto elegido por el S.O. en el paso anterior*

(4) FW al hacer masquerading mapea*
[IP intranet, puerto host intranet] -> [IP pública FW, puerto libre en FW]

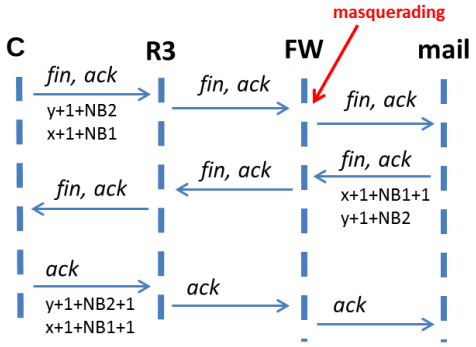
(5) Puerto elegido por FW en (4*)*

NB: Número de bytes del mensaje

4) PASO 4: Cierre de la conexión TCP

Se envía confirmación del último mensaje del servidor, junto con la solicitud de cierre de conexión

NB1 -> longitud en bytes del mensaje "helo"
NB2 -> longitud en bytes de la respuesta



- *La tabla y los campos son iguales que los del establecimiento de la conexión, salvo los flags, números de acuse y acks que se muestran en la figura.*

*** Primero se indican los acuses y luego los identificadores (números de secuencia) de cada segmento ***

EJERCICIO 3

Una sucursal con 50 empleados en Granada tiene una red interna basada en *FastEthernet* (100 Mbps) que se conecta a Internet con una red de acceso ADSL de 0,5 Mbps de subida y 1,5 Mbps de bajada. Cada empleado, en el desempeño de su trabajo, realiza un promedio de 2000 solicitudes de información a la hora a un servidor de Base de Datos ubicado en la central del banco, en Madrid, donde cada solicitud supone el envío por parte del servidor de un promedio de 10 registros de 1 KB cada uno. Adicionalmente, la modificación de datos tras algunas de estas solicitudes supone el envío promedio de 100 actualizaciones, de 10 registros de media, a la hora desde la sucursal al servidor. El resto de los servicios telemáticos se restringe. Calcule el promedio de la velocidad de transmisión requerida. ¿Es la velocidad del enlace de acceso suficiente?

Despreciando los paquetes de solicitud y confirmación, que contarán simplemente con cabeceras, podemos ver la velocidad requerida en promedio:

$$v_{download} = 2000\ \frac{sol}{emp\ x\ h} x\ 1\ \frac{registro}{sol} x\ 10 \frac{KB}{registro} x\ (8x1024) \frac{b}{KB}\ x\ 50\ emp\ x\ \frac{1}{3600}\ \frac{h}{s}$$

$$v_{download} = 2.28 \cdot 10^6\ bps = 2.28\ Mbps$$

$$v_{upload} = 100\ \frac{act}{emp\ x\ h} x\ 1\ \frac{registro}{act} x\ 10 \frac{KB}{registro} x\ (8x1024) \frac{b}{KB}\ x\ 50\ emp\ x\ \frac{1}{3600}\ \frac{h}{s}$$

$$v_{upload} = 0.11 \cdot 10^6\ bps = 0.11\ Mbps$$

La velocidad del enlace es insuficiente, ya que el *download* requerido en promedio es menor que el de la red.

EJERCICIO 4

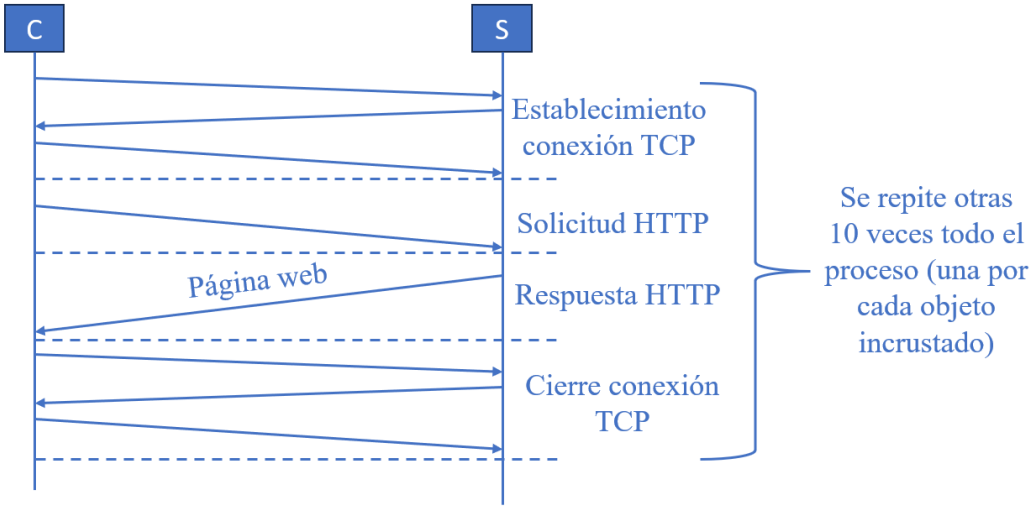
Compare el rendimiento en términos temporales de HTTP persistente y no persistente considerando los siguientes parámetros:

a. Descarga de una página web con 10 objetos incrustados.
b. Tiempo de Establecimiento de conexión TCP: 5 ms.
c. Tiempo de Cierre de conexión TCP: 5 ms.
d. Tiempo de solicitud HTTP: 2 ms.
e. Tiempo de respuesta HTTP (página web u objeto): 10 ms.

En una conexión persistente solo se hará una conexión TCP, mientras que en una conexión no persistente se utilizarán múltiples conexiones TCP, una por cada objeto solicitado.

HTTP ofrece dos tipos de servicio:
- **No persistente** → Se envía únicamente un objeto en cada conexión TCP.
- **Persistente** → Pueden enviarse múltiples objetos sobre una única conexión TCP entre cliente y servidor

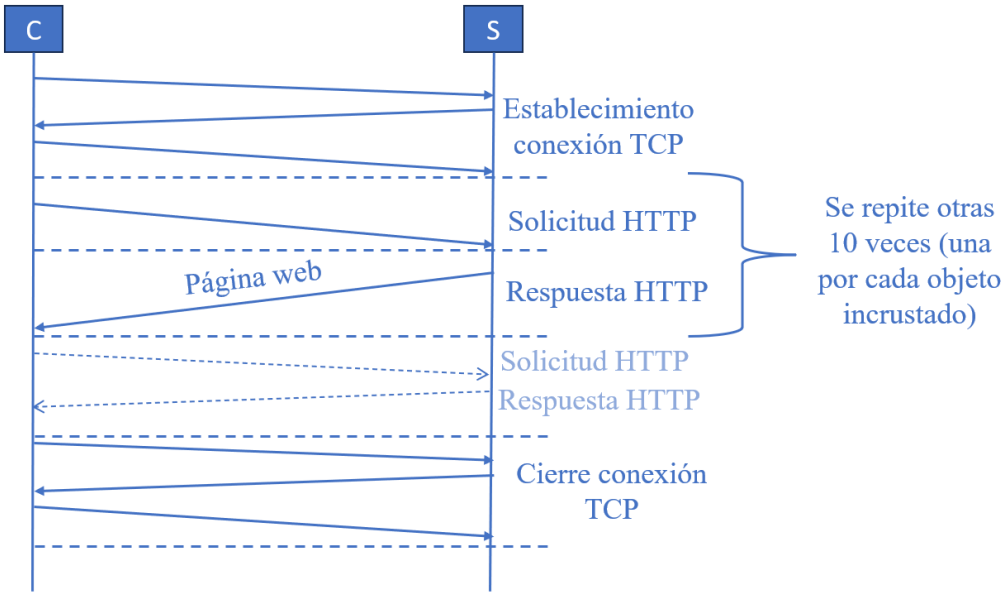
Para **HTTP no persistente** tenemos que el proceso de descarga sería el siguiente:



El tiempo de descarga para el modo HTTP no persistente es por lo tanto:

$$r_{descarga}^{np}=11\times\left(t_{estab}+t_{solic}+t_{resp}+t_{cierre}\right)=242ms$$

Para **HTTP persistente** tenemos que el proceso de descarga sería el siguiente:



El tiempo de descarga para el modo HTTP persistente es por lo tanto:

$$r_{descarga}^{p}=t_{estab}+11\times\left(t_{solic}+t_{resp}\right)+t_{cierre}=142ms$$

EJERCICIO 5

Discuta las características de las siguientes aplicaciones en términos de su tolerancia a la pérdida de datos, los requisitos temporales, la necesidad de rendimiento mínimo y la seguridad.

a) La telefonía móvil.
b) WhatsApp.
c) YouTube.
d) Spotify.
e) Comercio electrónico.

Nótese que el diseño de una aplicación siempre debe tener en cuenta todas las características anteriores. No obstante, varias de estas características son difíciles de conseguir de forma simultánea, cuando no son antagónicas. Por este motivo, el diseño de una aplicación debe considerar qué características primar en detrimento de otras.

Se propone la siguiente tabla de asignación de prioridades, utilizando la siguiente notación:

Requisito fundamental: ↑ Requisito relevante: ↔ Requisito secundario: ↓

	Tolerancia a pérdidas de datos	**Requisitos temporales (*Delay*)**	**Rendimiento mínimo (*Throughput*)**	**Seguridad**
Telefonía móvil	↓	↑	↑	↔
WhatsApps	↑	↓	↓	↔
YouTube	↓	↔	↑	↓
Spotify	↓	↔	↑	↓
Comercio electrónico	↑	↔	↔	↑