Guia 1: Nivel de Aplicación

Redes de Comunicaciones (TB067) - 2C2024 - FIUBA
Martin Klöckner - mklockner@fi.uba.ar

1. Para una sesión de comunicación entre un par de procesos, ¿qué proceso es el cliente y cuál es el servidor?

El proceso cliente es aquel que inicia la comunicación entre un par de procesos, mientra que el proceso servidor es aquel que espera a que otro proceso inicié la comunicación.

2. Para una aplicación de intercambio de archivos P2P, ¿está de acuerdo con la afirmación: “No existe la noción de los lados cliente y servidor de una sesión de comunicación”? ¿Por qué o por qué no?

No sé está de acuerdo, ya que existe la noción de cliente y servidor en una sesión de transferencia de archivos P2P, solo que cualquier proceso puede ser servidor o cliente, es decir, cualquier proceso puede iniciar la comunicación con otro, o esperar a que otro proceso inicie la comunicación.

3. ¿Qué es un “socket”?

Un “socket” es un conjunto de datos que permite la comunicación entre dos procesos. Cuando se establece una conexión entre dos procesos, cada proceso debe asignar un socket a esa comunicación.

En el modelo TCP/IP, se habla de un socket de internet, el cual permite la comunicación entre dos procesos, por lo general pertenecientes a dos computadoras distintas. Los sockets de internet se identifican por su numero de socket, el cual se crea a partir de el protocolo de transporte utilizado en la comunicación, la dirección IP local y el número de puerto.

4. Mencione una aplicación que requiera que no haya pérdida de datos y que también sea extremadamente sensible al tiempo.

Un ejemplo puede ser el protocolo SMTP utilizado para la comunicación de correos electrónicos, en este caso es extremadamente importante que no ocurra perdida de datos, ya que pueden transportar información sensible y/o importante; en cuanto al tiempo no es tan importante ya los usuarios pueden permitirse que se demore unos segundos.

Existen diversos ejemplos siendo la mayoría correspondiente a servicios interactivos en tiempo real, como la telefonía por internet (VoIP), las teleconferencias, o los juegos multijugador.

5. ¿Cuáles son algunas diferencias entre TCP y UDP?

La principal diferencia entre TCP y UDP es que TCP es más confiable ya que es mas robusto debido a varios mecanismos que lo diferencian de UDP y que lo hacen mas seguro, por ejemplo el proceso de 3 pasos que se utiliza en TCP para iniciar una sesión, o el procedimiento

6. ¿Por qué TCP y UDP no tienen mecanismos de cifrado?

Porque son protocolos relativamente viejos, los cuales en un principio no estaban pensados en términos de seguridad.

7. ¿Por qué HTTP, SMTP e IMAP se ejecutan sobre TCP en lugar de UDP?

Porque son protocolos que requieren que no haya perdida de datos. En el caso de SMTP o IMAP que se utilizan para la transmisión de correo electrónicos, la perdida de datos implicaría perdida de la información, la cual puede ser importante.

8. ¿Cuál es la diferencia entre una conexión HTTP persistente y una conexión no persistente?

La diferencia radica en que la conexión HTTP persistente una vez finalizada la transferencia de datos continua esperando por datos del usuario hasta que se termine el tiempo de conexión, mientras que la conexión HTTP no persistente termina una vez finalizada la transferencia.

9. ¿El almacenamiento en caché web reducirá la demora para todos los objetos solicitados por un usuario o solo para algunos de los objetos? ¿Por qué? ¿En qué casos un almacenamiento en caché web no mejora el tiempo de respuesta?

El almacenamiento en caché web (también llamado servidor proxy) siempre reduce la demora en la carga de archivos, ya que por lo general están mas cerca de los usuarios; en tal caso se evita la transferencia desde el servidor, que por lo general suele estar mas lejos al cliente.

Puede que la copia de datos almacenada en el servidor caché sea obsoleta con respecto a los datos del servidor, en ese caso el servidor caché tendrá que obtener los datos desde el servidor, y en ese caso existe una demora mayor que si no hubiera servidor caché.

10. La siguiente cadena de caracteres ASCII ha sido capturada por Wireshark cuando el navegador enviaba un mensaje GET HTTP. Responda a las siguientes cuestiones, indicando en qué parte del siguiente mensaje GET HTTP se encuentra la respuesta.
    a. ¿Cuál es el URL del documento solicitado por el navegador?
    b. ¿Qué versión de HTTP se está ejecutando en el navegador?
    c. ¿Qué tipo de conexión solicita el navegador, persistente o no persistente?
    d. ¿Cuál es la dirección IP del host en el que se está ejecutando el navegador?
    e. ¿Qué tipo de navegador inicia este mensaje? ¿Por qué es necesario indicar el tipo de navegador en un mensaje de solicitud HTTP?

```
GET /cs453/index.html HTTP/1.1<cr><lf>
Host: gaia.cs.umass.edu<cr><lf>
User-Agent: Mozilla/5.0 (Windows;U; Windows NT 5.1; en-US; rv:1.7.2)
Gecko/20040804 Netscape/7.2 (ax)<cr><lf>
Accept:ext/xml, application/xml, application/xhtml+xml, text/html;q=0.9,
text/plain;q=0.8, > image/png,*/*;q=0.5<cr><lf>
Accept-Language: en-us,en;q=0.5<cr><lf>
Accept-Encoding: zip,deflate<cr><lf>
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7<cr><lf>
Keep-Alive: 300<cr><lf>
Connection:keep-alive<cr><lf><cr><lf>
```

a. La URL del documento solicitado es `www.gaia.cs.umass.edu/cs453/index.html` ya que se indica en la sección del encabezado `Host:` .
b. El navegador está ejecutando la version `1.1` de HTTP, esto se puede ver en la sección `GET /cs453/index.html HTTP/1.1` .
c. El navegador solicita una conexión de tipo persistente, la cual es la acción por defecto del protocolo HTTP. Esto se puede ver en la ultima linea del encabezado la cual indica: `Connection:keep-alive` .
d. La dirección IP del host en que se esta ejecutan el navegador no se puede sabe ya que no se indica en el encabezado HTTP.
e. El tipo de navegador se puede ver en la parte `User-Agent` del encabezado HTTP, en este caso es el navegador Netscape version 7.2 de escritorio, en particular corriendo sobre el sistema operativo Windows NT 5.1; `Mozilla/5.0` indica que es compatible con ese navegador, y se incluye por razones históricas.

11. El siguiente texto muestra la respuesta devuelta por el servidor al mensaje de solicitud GET HTTP del problema anterior. Responda a las siguientes cuestiones, indicando en qué parte del siguiente mensaje se encuentran las respuestas.
    a. ¿Ha podido el servidor encontrar el documento? ¿En qué momento se suministró la respuesta con el documento?
    b. ¿Cuándo fue modificado por última vez el documento?
    c. ¿Cuántos bytes contiene el documento devuelto?
    d. ¿Cuáles son los primeros cinco bytes del documento que se está devolviendo?
    e. ¿Ha acordado el servidor emplear una conexión persistente?

```
HTTP/1.1 200 OK<cr><lf>Date: Tue, 07 Mar 2008 12:39:45GMT<cr><lf>Server:
Apache/2.0.52 (Fedora) <cr><lf>Last-Modified: Sat, 10 Dec2005 18:27:46
GMT<cr><lf>ETag: "526c3-f22-a88a4c80"<cr><lf>Accept- Ranges:
bytes<cr><lf>Content-Length: 3874<cr><lf> Keep-Alive:
timeout=max=100<cr><lf>Connection: Keep-Alive<cr><lf>Content-Type:
text/html; charset= ISO-8859-1<cr><lf><cr><lf><!doctype html public "-
//w3c//dtd html 4.0transitional//en"><lf><html><lf> <head><lf> <meta
http-equiv="Content-Type" content="text/html; charset=iso-8859-1"><lf>
```

```
<meta name="GENERATOR" content="Mozilla/4.79 [en] (Windows NT 5.0; U)
Netscape]"><lf> <title>CMPSCI 453 / 591 / NTU-ST550ASpring 2005
homepage</title><lf></head><lf> <aquí continúa el texto del documento (no
mostrado)>
```

a. Si lo ha podido encontrar ya que el codigo de respuesta es `200 OK` , el moemnto en que se suministro la respuesta fue en la fecha `Tue, 07 Mar 2008 12:39:45 GMT` .
b. El documento recibido por el servidor fue modificado por utilma vez en la fecha `Sat, 10 Dec 2005 18:27:46 GMT` , como se puede ver en la seccion `Last-Modified` del encabezado.
c. El documento devuelto contiene `3874` bytes, esto se puede ver en la etiqueta `Length` del encabezado.
d. Los primeros 5 bytes son `<!doc` , la secuencia `<cr><lf><cr><lf>` indica el termino del encabezado HTTP y luego comienza el documento devuelto (recordemos que cada caracter ocupa 1 byte).
e. Si, se puede ver en la etiqueta `Connection:` del encabezado HTTP, la cual indica `Keep-Alive` .

12. Haga Telnet a un servidor web y envíe un mensaje de solicitud multilínea. Incluya en el mensaje de solicitud la línea de encabezado `If-modified-since:` para forzar un mensaje de respuesta con el código de estado 304 No modificado.

Se accede utilizando telnet al servidor web `www.example.com` en el puerto 80, de la siguiente manera:

```
$ telnet www.example.com 80
Connected to example.com.
Escape character is '^]'.
```

Luego se envia un mensaje multilinea de peticion de `index.html` con el codigo `If-Modified-Since: Wed, 10 Oct 2024 10:00:00 GMT` , con lo cual el servidor responde

```
HTTP/1.1 304 Not Modified
Accept-Ranges: bytes
Age: 596825
Cache-Control: max-age=604800
Date: Thu, 17 Oct 2024 16:10:53 GMT
Etag: "3147526947+gzip"
Expires: Thu, 24 Oct 2024 16:10:53 GMT
Last-Modified: Thu, 17 Oct 2019 07:18:26 GMT
Server: ECAcc (mid/871B)
Vary: Accept-Encoding
X-Cache: HIT
```

Notese que se agrega el campo `Host: example.com` ya que este servidor acepta HTTP versión 1.1 y en esa version es obligatorio este campo.

13. La LAN de una universidad tiene una velocidad de transmisión de 100 Mbps. Para acceder a Internet tiene un enlace de acceso cuya velocidad de transmisión es de 10 Mbps. La velocidad media de paquetes es de 20 solicitudes / seg. Si cada paquete es de 1Mbit, se pide:
    a. La velocidad media de los bits en bits/seg.
    b. La intensidad de tráfico en la LAN.
    c. La intensidad de tráfico en el enlace de acceso.
    d. ¿Cómo son las intensidades de tráfico calculadas comparadas con 1? ¿Qué significa esa comparación en términos de retardos?
    e. Proponga dos soluciones posibles para bajar la intensidad de tráfico en el enlace de acceso.
    f. Suponga que la universidad instala una caché y que la tasa de acierto es de 0,45. ¿Qué porcentaje de solicitudes serán satisfechas casi de inmediato?¿Qué porcentaje de solicitudes serán satisfechas por los servidores de origen?

14. Un cliente HTTP desea recuperar un documento web que se encuentra en un URL dado. Inicialmente, la dirección IP del servidor HTTP es desconocida. ¿Qué protocolos de la capa de aplicación y de la capa de transporte además de HTTP son necesarios en este escenario?

En principio para obtener la direccion IP del servidor HTTP se necesita el protocolo del nivel de aplicacion DNS, de manera tal que resuelva la URL y obtenga asi la direccion IP del servidor, luego se necesita un protocolo de la capa de aplicacion que realice una peticion al servidor HTTP por el archivo, por ejemplo FTP; el protocolo FTP utiliza el protocolo de transporte TCP para obtner los archivos del servidor.

15. Mencione 3 motivos por los cuales un servidor de DNS no puede ser centralizado.

En principio porque un servidor DNS es critico para resolver la URL de otros servidores de internet, por lo que ser centralizado implicaria la dependencia de una sola organizacion o servidor central.

Otro motivo seria la enorme cantidad de tráfico que este servidor centralizado tendria que manejar.

Por ultimo que un servidor sea centralizado implicaria una enorme perdida de rendimiento, ya que cualquier region que quiera acceder a internet deberia pasar por este servidor centralizado, que puede que esté a una distancia muy lejana.

16. Dada la jerarquía de servidores DNS (Servidores DNS raíz, Servidores de dominio de nivel superior y servidores autoritativos) se pide
    a. ¿Qué direcciones IP proporcionan cada uno?
    b. Cuando un host realiza una consulta DNS, ¿a qué tipo de servidor de DNS,

que actúa como proxy llega?

c. Llamamos R al Servidor DNS raíz, T al Servidor TLD, A al servidor autoritativo, L al servidor local y H al host que realiza la consulta. Suponíamos que el servidor TLD conoce el servidor DNS autoritativo correspondiente al nombre de host. Indique el trayecto del mensaje de consulta desde que el host lo inicia hasta que obtiene la dirección IP del host consultado mediante la letra correspondiente, un guión, la letra siguiente y así sucesivamente.

d. ¿Cuántos mensajes DNS se necesitan enviar para obtener la dirección correspondiente a un nombre de host si no se encuentra en el proxy del servidor local y servidor TLD conoce el servidor DNS autoritativo correspondiente al nombre del host consultado?

17. ¿Cuál es la diferencia entre consultas de DNS iterativas y consultas recursivas?

Una consulta DNS recursiva ocurre cuando un servidor DNS se comunica con otros servidores DNS para intentar resolver una direccion URL y devolverla al cliente, en cambio, una consulta DNS iterativa ocurre cuando el cliente se comunica directamente con cada servidor DNS involucrado en la resolucion de la direccion URL [1] .

18. Dado que la correspondencia entre el nombre de un host y su dirección IP puede cambiar, ¿cuál es el comportamiento de un servidor de DNS respecto de la información almacenada en su caché DNS para prevenir esto?

19. Un cliente se conecta a una aplicación de home banking basada en la web mediante protocolo HTTP, el cual no tiene memoria del estado de la conexión. Una vez que inició sesión, ¿cómo identifica el servidor al cliente?

20. Supongamos que estás realizando un análisis de tráfico de red y te encuentras con la siguiente captura de dos paquetes DNS relacionados con la resolución de nombres de dominio para “wireshark.org”.

    a. ¿Qué tipo de consulta DNS se envía en el primer paquete y quién la realiza?

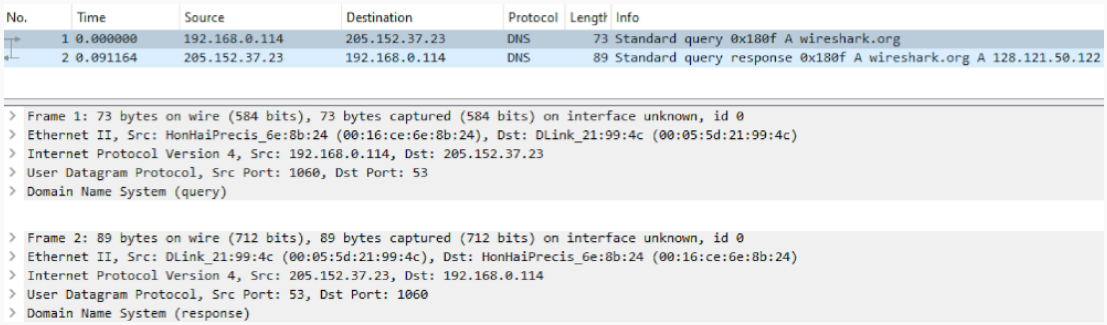
    b. ¿Cuál es la dirección IP del servidor DNS al que se envía la consulta DNS en el primer paquete?

    c. ¿Qué tipo de respuesta se recibe en el segundo paquete y cuál es la dirección IP asociada al nombre de dominio “wireshark.org”?

    d. ¿Cuáles son las direcciones MAC de origen y destino en ambos paquetes Ethernet?

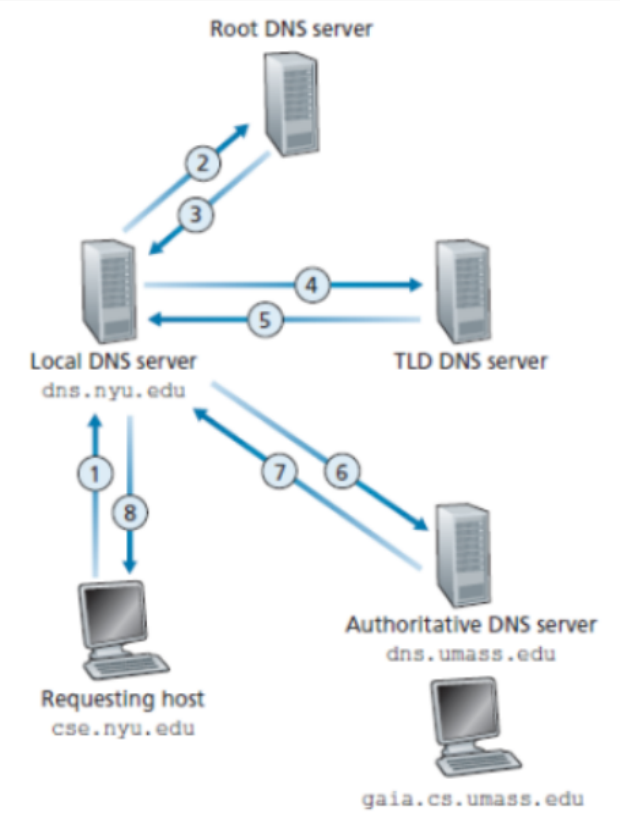
    e. ¿Puedes explicar por qué el puerto de origen y destino cambia entre la consulta DNS en el primer paquete y la respuesta DNS en el segundo paquete?

[1] https://www.cloudflare.com/learning/dns/what-is-recursive-dns/



21. Descargar, analizar mediante Wireshark y contestar las preguntas sobre la transmisión de datos HTTP relacionada con la descarga de un archivo de imagen desde un servidor web a partir de la captura de paquetes de HTTP.cap del siguiente enlace: https://packetlife.net/captures/category/web/
    a. ¿Qué recurso se está solicitando en el primer paquete HTTP y quién realiza la solicitud?
    b. ¿Cuál es el código de estado de la respuesta del servidor en el segundo paquete y qué significa este código?
    c. ¿Qué tipo de archivo se está transfiriendo según la información proporcionada en la captura?
    d. ¿Cuál es el tamaño del contenido (en bytes) de la imagen que se está transfiriendo según la respuesta del servidor?
    e. ¿Cuál es la longitud de la respuesta HTTP (en bytes) en el segundo paquete?
    f. ¿Cuál es la fecha y hora en que se envió la respuesta del servidor al cliente según los datos proporcionados en la captura?

22. Enviar un mensaje de consulta DNS directamente desde el host en el que está trabajando al servidor google.com.ar mediante nslookup.
    a. ¿Qué información devuelve este comando?
    b. ¿Cuál es la dirección IP del servidor web de google.com.ar?
    c. ¿Cuál es la dirección IP del servidor DNS que proporcionó la respuesta al comando nslookup?
    d. La respuesta de este comando proporciona dos datos, ¿Que representa cada uno?
    e. Existen tres clases de servidores DNS: los servidores DNS raíz, los servidores DNS de dominio de nivel superior (TLD, Top-Level Domain) y los servidores DNS autoritativos. Los servidores de nombres raíz proporcionan las direcciones IP de los servidores TLD. Los servidores TLD proporcionan las direcciones IP para los servidores DNS autoritativos. Los servidores autoritativos contienen las direcciones IP y sus nombres correspondientes de cada página web. ¿Qué servidor, según el esquema de la figura siguiente, devuelve esta información?



23. Para el comando `nslookup -type=NS fi.uba.ar` se pide:
    a. ¿Qué información devuelve el comando?
    b. ¿La respuesta al comando nslookup provino de un servidor autorizado o no autorizado?
    c. ¿Qué significa “Respuesta no autoritativa” en la respuesta?
    d. ¿De qué tipo es el archivo de recursos que contiene la información devuelta?
    e. ¿Por qué hay dos tipos diferentes de direcciones IP?

24. ¿Qué respuesta aparece en la pantalla con el comando nslookup 157.92.1.1?

25. DNS usa UDP en vez de TCP. Si se pierde un paquete DNS, no hay recuperación automática. ¿Provoca esto un problema y, de ser así, cómo se resuelve?

26. Suponga que en UDPCliente.py, después de crear el socket, añadimos esta línea: clientSocket.bind((’’, 5432))
    a. ¿Será necesario modificar el programa UDPServidor.py?
    b. ¿Cuáles son los números de puerto para los sockets en UDPCliente y UDPServidor luego del cambio?
    c. ¿Cuáles eran antes de realizar este cambio?

27. Análisis de captura de paquetes del protocolo DNS.
    a. Mediante el comando `ipconfig /all` obtener
        - Dirección IPv4 e IPv6 la placa de red o de WiFi según corresponda.
        - Dirección MAC
        - Dirección IP de la puerta de enlace predeterminada
        - Direcciónes IP del servidor DNS
    b. En Wireshark, en Interface List (captura-opciones-entrada) elija la asociada a la IP y MAC registrada en a. Guardar la captura en un archivo llamado ejercicio30.pcap (captura-opciones-salida)
    c. Comience a capturar paquetes.
    d. Ir a www.google.com en un navegador.
    e. Al ver la página de Google detenga la captura de paquetes.
    f. Filtre paquetes DNS Si no se ve ninguno, cerrar el navegador web y enviar el comando ipconfig /flushdns. Repetir los pasos desde b. Si no se ven paquetes DNS, enviar el comando nslookup www.google.com en vez de usar el navegador.
    g. Busque un paquete standard query (A) google.com y complete la tabla:
    Número de trama:
    Cantidad de bytes capturados:
    Dirección MAC de origen:
    Dirección MAC registrada en a):
    Dirección MAC de destino:
    Dirección IP de origen:
    Dirección IP registrada en a):
    Dirección IP de destino:
    Dirección IP de la puerta de enlace predeterminada registrada en a)