

Redes y Comunicaciones 2017

Práctica 6

Autor: Fermín Minetto (<http://www.github.com/ferminmine>)

2) ¿Qué tipos de estado puede tener una conexión TCP?

Los tipos de estado que puede adoptar una conexión TCP son: *CLOSED*, *LISTEN*, *SYN RECEIVED*, *SYN SENT*, *ESTABLISHED*, *SYN*, *FIN WAIT*, *CLOSE*, *WAIT*, *CLOSED*.

3) Dada la siguiente captura del comando ss

Netid	State	Recv-Q	Send-Q	Local Address:Port	Peer Address:Port	users:((("sshd",pid=468,fd=29))
tcp	LISTEN	0	128	*:22	*:*	users:((("apache2",pid=991,fd=95))
tcp	LISTEN	0	128	*:80	*:*	users:((("named",pid=452,fd=10))
udp	LISTEN	0	128	163.10.5.222:53	*:*	users:((("named",pid=452,fd=10))
tcp	ESTAB	0	0	163.10.5.222:59736	64.233.163.120:443	users:((("x-www-browser",pid=1079,fd=51))
tcp	CLOSE-WAIT	0	0	163.10.5.222:41654	200.115.89.30:443	users:((("x-www-browser",pid=1079,fd=50))
tcp	ESTAB	0	0	163.10.5.222:59737	64.233.163.120:443	users:((("x-www-browser",pid=1079,fd=55))
tcp	ESTAB	0	0	163.10.5.222:33583	200.115.89.15:443	users:((("x-www-browser",pid=1079,fd=53))
tcp	ESTAB	0	0	163.10.5.222:45293	64.233.190.99:443	users:((("x-www-browser",pid=1079,fd=59))
tcp	LISTEN	0	128	*:25	*:*	users:((("postfix",pid=627,fd=3))
tcp	ESTAB	0	0	127.0.0.1:22	127.0.0.1:41220	users:((("sshd",pid=1418,fd=3), ("sshd",pid=1416,fd=3))
tcp	ESTAB	0	0	163.10.5.222:52952	64.233.190.94:443	users:((("x-www-browser",pid=1079,fd=29))
tcp	TIME-WAIT	0	0	163.10.5.222:36676	54.149.207.17:443	users:((("x-www-browser",pid=1079,fd=3))
tcp	ESTAB	0	0	163.10.5.222:52960	64.233.190.94:443	users:((("x-www-browser",pid=1079,fd=67))
tcp	ESTAB	0	0	163.10.5.222:50521	200.115.89.57:443	users:((("x-www-browser",pid=1079,fd=69))
tcp	SYN-SENT	0	0	163.10.5.222:52132	43.232.2.2:9500	users:((("x-www-browser",pid=1079,fd=70))
tcp	ESTAB	0	0	127.0.0.1:41220	127.0.0.1:22	users:((("ssh",pid=1415,fd=3))
udp	LISTEN	0	128	127.0.0.1:53	*:*	users:((("named",pid=452,fd=9))

- ➔ **Conexiones establecidas: 8. Ojo:** *no todas las conexiones established cuentan como establecidas* puesto que 127.0.0.1:22 a 127.0.0.1:41220 y 127.0.0.1:41220 a 127.0.0.1:22 son en realidad la misma conexión pero aparece como doble porque una es la respuesta a la otra, pero solamente cuenta como una.
- ➔ **Puertos abiertos a la espera de conexiones: 5**
- ➔ Los clientes y servidores de las comunicaciones HTTPS(443) no residen en la misma máquina puesto que tienen direcciones IP distintas (*Local Address:Port*, *Peer Address:Port*)
- ➔ El cliente y servidor de las comunicaciones SSH(22) residen en la misma máquina puesto que tienen la misma dirección IP.
- ➔ **Conexiones pendientes por establecerse: 1.** El estado de la conexión por establecerse es SYN SEND.

4) Dada la salida del comando ss

```
servidor# ss -natu | grep 110
```

```
tcp    LISTEN    0      0      *:110      *:*
tcp    SYN-RECV  0      0    157.0.0.1:110  157.0.11.1:52843
```

```
cliente# ss -natu | grep 110
```

```
tcp    SYN-SENT  0      1    157.0.11.1:52843  157.0.0.1:110
```

```
cliente# ss -natu | grep 110
```

El segmento que se está perdiendo en la red es *el segmento SYN ACK del servidor*. El cliente está intentando acceder al protocolo POP3, utilizado para recuperar el correo electrónico de un servidor de correo. Los flags que tendría seteado serían el SYN y el ACK en 1.

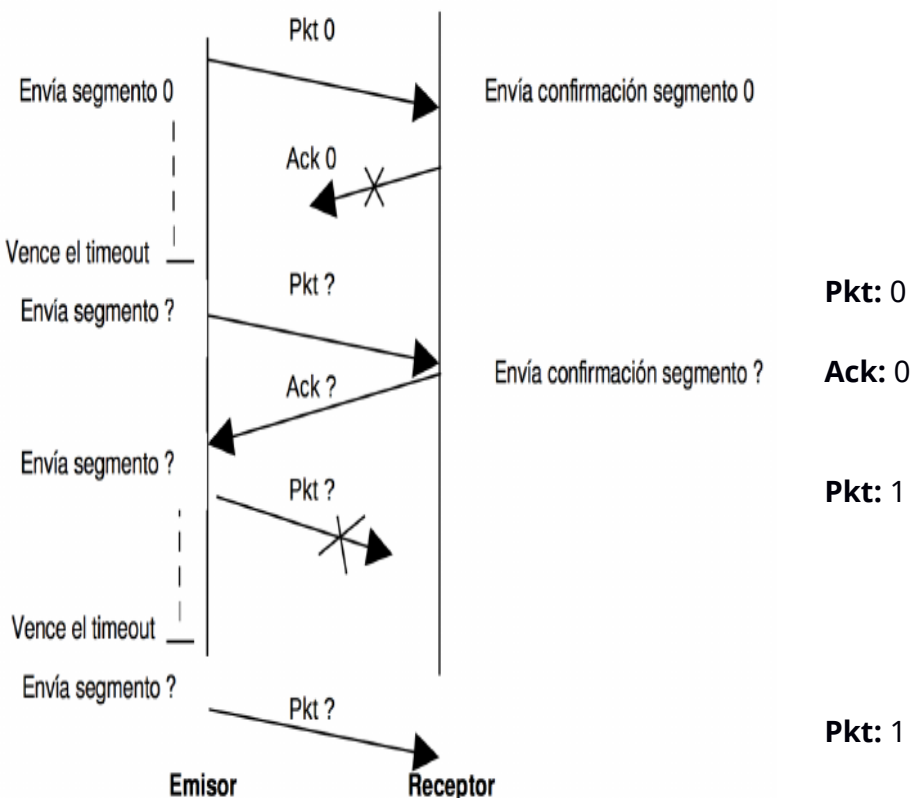
5) ¿Cual es el puerto por defecto que se utiliza en los siguientes servicios? ¿En que lugar de Linux y Windows se almacenan estas correspondencias?

- ➔ **Web:** 80/TCP
- ➔ **SSH:** 22/TCP
- ➔ **DNS:** 53/UDP
- ➔ **Web Seguro:** 443/TCP
- ➔ **POP3:** 110/TCP
- ➔ **IMAP:** 143/TCP
- ➔ **SMTP:** 25/TCP
- ➔ **FTP:** 20/TCP (Data), 21/TCP (Control)
- ➔ **TFTP:** 69/UDP

Estas correspondencias se almacenan en el archivo `/etc/services` de Linux. Su equivalente en Windows es el archivo `C:\Windows\System32\drivers\etc\services`.

6) Complete los ? de la siguiente secuencia

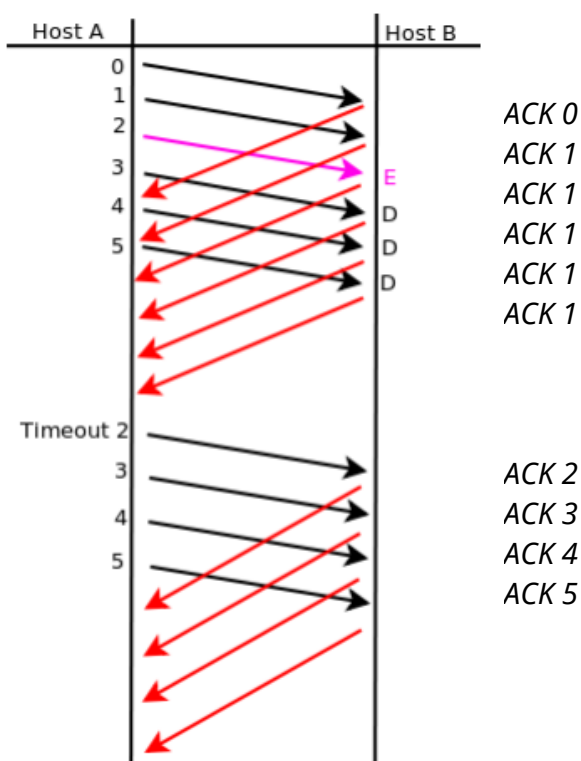
En caso de que se enviara un paquete más, volvería a comenzar desde el 0 puesto que en el stop and wait se van intercalando paquetes 0 y 1.



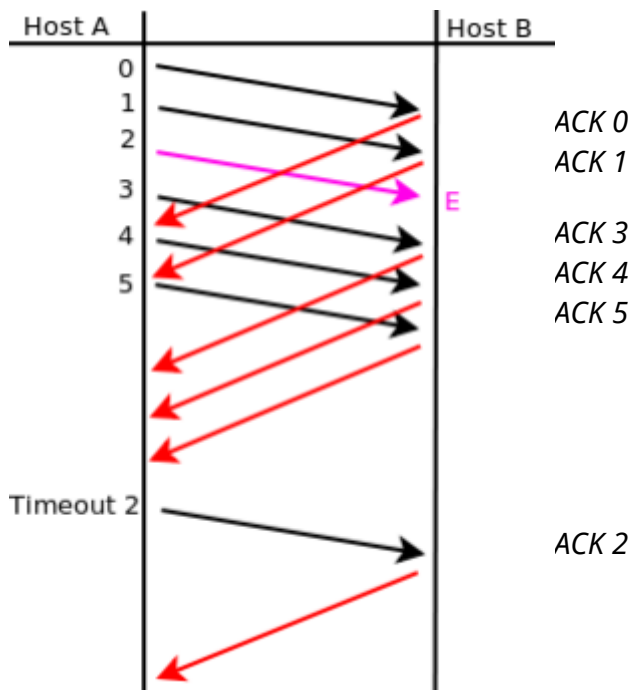
7) Explicar la lógica del Go Back N

En un protocolo GBN, el emisor puede transmitir varios paquetes (si están disponibles) sin tener que esperar a que sean reconocidos, pero está restringido a no tener más de un número máximo permitido, N (para así no congestionar los enlaces), de paquetes no reconocidos en el canal. N suele denominarse tamaño de ventana y cuando el protocolo opera esta ventana se va desplazando hacia adelante, aumentando el número de segmentos que pueden ser enviados, por eso se lo denomina un protocolo de ventana deslizante. En el protocolo GBN, el reconocimiento de un paquete es acumulativo, es decir que si se manda el reconocimiento de un paquete n , todos los paquetes con un número mayor o igual que n han sido correctamente recibidos por el receptor. Si se produce un fin de temporización, el emisor reenvía todos los paquetes que no hayan sido reconocidos por el receptor.

8) Suponiendo Go Back N; tamaño de ventana 4 y sabiendo que E indica que el mensaje llegó con errores y que D significa que el mensaje será descartado por llegar fuera de secuencia. Indique en el siguiente gráfico, la numeración de los ACK que el host B envía al Host A.



9) Suponiendo Selective Repeat; tamaño de ventana 4 y sabiendo que E indica que el mensaje llegó con errores. Indique en el siguiente gráfico, la numeración de los ACK que el host B envía al Host A.



10) ¿Qué restricción existe sobre el tamaño de ventanas en el protocolo Selective Repeat?

**11) Conéctese al servidor ftp utilizando el comando ftp
ftp.redes.unlp.edu.ar**

FTP es un protocolo de transferencia de archivos a través de Internet. FTP mantiene una conexión para enviar información de control y otra para la transferencia de archivos. FTP utiliza TCP, el puerto 20/TCP para los datos y el puerto 21/TCP para la información de control.

FTP tiene dos modos: pasivo y activo.

En el modo activo el servidor se conecta al cliente. El cliente le debe indicar al servidor con el comando PORT a que número de puerto debe conectarse el servidor, el cual será un puerto no privilegiado (arriba de 1024). El servidor desde su puerto 20 se conectará al puerto indicado por el cliente para cada operación que requiera transferencia de datos. Era el modo default pero ya no se recomienda su uso.

En el modo pasivo el cliente es el que se conecta al servidor. El cliente le indica con el comando PASV que quiere conectarse al servidor, y este último como respuesta le indica el número de puerto al que debe conectarse.

Diferencias con TFTP: Usa el puerto 69 y UDP como protocolo de transporte. TFTP usa comandos simples mientras que FTP usa comandos robustos. TFTP es un protocolo *no orientado a la conexión*, FTP si y utiliza dos conexiones, una para mantener el control de la conexión y otro para los datos. TFTP requiere menos memoria y esfuerzo de programación, FTP requiere más memoria y más esfuerzo en la programación.

Comandos FTP: mkdir, rmdir, ls, cd, get (recuperar archivos en posición local donde

está la terminal) y put (colocar archivo desde posición actual en máquina local desde terminal).