

Práctica 6 – Midiendo throughput en Ethernet

1- Objetivos

El objetivo de esta práctica es familiarizarse con los equipos de red de área local Ethernet y medir y comprender el throughput en una red real.

2- Throughput entre dos PCs [2% tiempo estimado 40 minutos]

En esta práctica usaremos de nuevo los puestos del Laboratorio de Telemática 1 con armarios rack y equipo de red. En esta primera fase conectaremos un PC a otro directamente para comprobar el límite de velocidad de la tarjeta ethernet y posteriormente pondremos redes más complejas entre ellos para ver el efecto.

En primer lugar utilizaremos PC-A para enviar a PC-B. Identifique los puertos de red en el armario que corresponden con eth0 de PC-A y PC-B (recuerde la práctica anterior) y únalos con un cable cruzado (azul).

Configure la dirección IP 10.0.0.1 en PC-A y la 10.0.0.2 en PC-B.

Compruebe que lo ha hecho correctamente probando pings entre PC-A y PC-B

En PC-A

```
$ ping 10.0.0.2
```

En PC-B

```
$ ping 10.0.0.1
```

Si no funciona asegúrese de arreglarlo antes de continuar

Utilizaremos la herramienta iperf para enviar datos continuamente entre los PCs y poder observar el tráfico y medir el throughput. Iperf funciona lanzándose en los ordenadores origen y destino de la información. Los programas se comunican a través de la red y envían la cantidad de información que indiquemos. El propio iperf nos da su propia medida del throughput que consigue enviar pero lo verificaremos observando la red con wireshark

En el ordenador que va a ser el receptor debemos lanzar iperf en modo servidor para que reciba los datos que le envíen desde otros iperfs. Iperf puede recibir datos utilizando un protocolo de transporte fiable llamado TCP o bien utilizando un protocolo de transporte no fiable llamado UDP. El transporte fiable se asegura de que los datos llegan al otro lado y no se pierden, al no fiable le da igual que se pierdan por lo que conseguirán diferentes resultados. Los protocolos de transporte utilizan puertos para identificar los diferentes programas que están usando esos protocolos. Los puertos son simplemente números.

Ponga en PC-B un servidor iperf para recibir datos con transporte fiable en el puerto 10000. Eso se hace con el comando

```
$ iperf -s -p 10000
```

Vera en la pantalla que el programa está esperando a que le lleguen datos. Esperará indefinidamente e imprimirá por pantalla estadísticas cada vez que le lleguen datos. De momento déjelo esperando en un terminal

Ponga en otro terminal de PC-B otro servidor iperf para recibir datos con transporte no fiable en el puerto 20000. Eso se hace con

```
$ iperf -s -p 20000 -u
```

Déjelo esperando en su terminal con eso ya tenemos programas que esperan y reciben los datos que les envíen desde otro ordenador.

Enviemos datos desde PC-A. Lance iperf desde PC-A en modo cliente. En ese modo enviará datos continuamente a un servidor indicado. Continúa pero solo durante el tiempo que le indique. Por ejemplo para enviar datos con transporte fiable al servidor anterior lance en PC-A

```
$ iperf -c 10.0.0.2 -p 10000 -t 10
```

Eso hará que durante 10 segundos el programa envíe usando la red datos continuamente hacia PC-B. Los datos se enviarán con un protocolo de ventana deslizante de los que se ven en teoría y se confirmarán para asegurarse de que llegan. Pero lo que nos importa es ver cuantos datos atraviesan la red. Observe que tanto el iperf cliente como el servidor nos dicen cuanto throughput en Mbps han conseguido enviar pero queremos ser capaces de medirlo de forma independiente

Lance un wireshark en PC-A y otro en PC-B. Ponga cada uno a capturar todo el tráfico de red que se vea en eth0. Lance entonces el cliente durante 10 segundos y espere a que termine. Detenga los wiresharks y observe el throughput en Mbps que han visto ¿cuanto es el throughput que ha observado? ¿Coincide con el que dice iperf? ¿Por que ese throughput? ¿De que velocidad diría que es la tarjeta Ethernet de PC-A y PC-B?

Si usamos iperf con transporte fiable iperf intenta enviar a toda la velocidad que puede y vemos los límites del transporte fiable. Si usamos iperf con transporte no fiable es posible que enviemos datos tan deprisa que se pierdan. Iperf nos deja limitar esta velocidad controlando la carga que introduce un programa en la red. Use ahora el transporte no fiable para enviar datos al servidor. El comando es parecido solo que tiene que decir que es transporte UDP con la opción -u y puede usar la opción -b para especificar la velocidad de envío. Enviemos a varias velocidades para probar

```
$ iperf -c 10.0.0.2 -p 20000 -u -t 10 -b 4Mbps
```

```
$ iperf -c 10.0.0.2 -p 20000 -u -t 10 -b 40Mbps
```

```
$ iperf -c 10.0.0.2 -p 20000 -u -t 10 -b 150Mbps
```

Observe los resultados obtenidos. Realice un experimento para medir esto y presentarlo al profesor. Prepare los wiresharks en el PC-A y PC-B y capture una traza en la que se vean en sucesión rápida las tres pruebas de antes con 4,40 y 150Mbps. Cuando tenga las trazas en el origen y el destino dibuje la gráfica del throughput en la que se vean los tres envíos y se pueda ver los throughputs obtenidos. Cuando tenga esta gráfica abra un terminal y haga

\$ ifconfig eth0

coloque el terminal de forma que se vea el resultado y la gráfica correctamente y haga una captura de pantalla. Grabe la captura de pantalla de PC-A y PC-B con los nombres

cp1-origen.png

cp1-destino.png

Estos son los ficheros que deberá subir para probar que ha realizado el checkpoint 1. Si quiere puede mostrar primero el checkpoint y generar los archivos cuando le haya dado el visto bueno

CHECKPOINT1: (puntuación 2%) muestre sus resultados al profesor y suba los ficheros

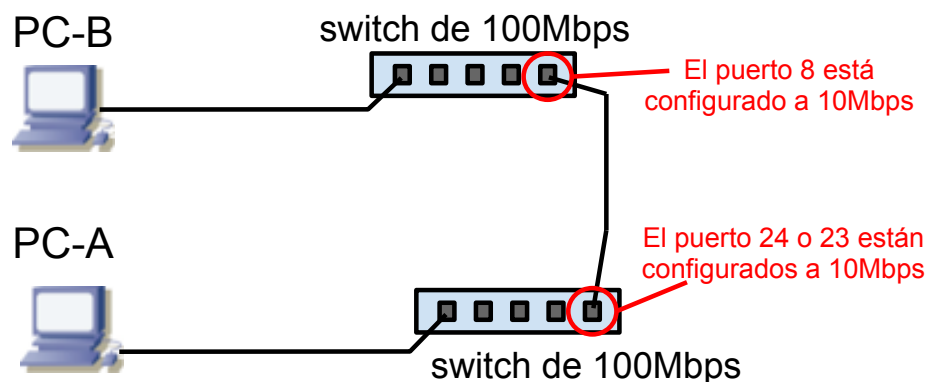
Si ha llegado hasta aquí llame al profesor de practica y muéstrole los throughputs que ha obtenido. Explíquele si tienen sentido y conteste a las preguntas.

Suba los ficheros con las fotos de pantalla en la tarea correspondiente

3- Throuhput atravesando Ethernet [2% tiempo estimado 40minutos]

Prepare un escenario en el que haya elementos de red entre PC-A y PC-B como se ve en la figura. Queremos que PC-A y PC-B estén en redes Ethernet de 100Mbps pero que estas dos redes estén unidas entre si por un enlace a 10Mbps. Para eso se han configurado unos puntos a 10Mbps en los conmutadores del laboratorio. Debe usar exactamente los conmutadores y puertos indicados. En primer lugar probemos los dos PCs en el mismo conmutador, conecte el puerto eth0 de PC-A al conmutador Ethernet (marca D-Link) inferior del armario. Conecte eth0 de PC-B a otro puerto del mismo switch (que no sean los puertos últimos 23 y 24). Vuelva a realizar los experimentos anteriores con iperf y compruebe que dan el mismo resultado. Si ha elegido un switch de 100Mbps no debería haber diferencia.

Configure ahora el escenario de la figura. PC-A ya está en el switch correcto. Cambie PC-B al switch de la práctica anterior (el verde de marca OVISLINK). Use uno de los puertos del 1 al 7. Ahora no debería poder hacer ping de PC-A a PC-B porque falta unir los conmutadores. Una el puerto 8 del conmutador superior con el puerto 23 o 24 del conmutador inferior. Al hacer esto debería poder hacer ping. Asegúrese de que esto funciona correctamente antes de seguir.



Una vez configurado realice el experimento anterior nuevamente con transporte no fiable y velocidades de 3Mbps, 8Mbps y 15Mbps. Obtenga las gráficas en PC-A y PC-B. ¿Por qué ahora no son iguales?

Genere pruebas de haber realizado este experimento como en el caso anterior. Guárdelas con el nombre cp2-origen.png y cp2-destino.png

CHECKPOINT2: suba los ficheros del segundo experimento (puntuación 2%)

Para conseguir este checkpoint tiene que subir los ficheros.

No es necesario mostrárselos al profesor de prácticas.

4- Escenario avanzado con mas hosts [1% tiempo estimado 40minutos]

Configure el PC-C en el mismo switch que PC-A, use la dirección 10.0.0.3 por ejemplo. Coloque en PC-C tres envíos con iperf hacia el servidor de PC-B cada uno de los cuales debe enviar 2Mbps y mantenerse indefinidamente. Por ejemplo poniéndoles 1h de tiempo.

Prepare capturas con wireshark en los tres ordenadores para lanzarlas cuando tenga listo lo siguiente.

En PC-A prepare un envío con transporte fiable hacia PC-B que dure unos 20segundos. Lance las capturas espere 10 segundos lance el envío y después de que acabe espere otros 10segundos y pare todas las capturas.

Haga las gráficas del throughput y compárelas. Interprete si tienen sentido los throughputs obtenidos. Guarde las capturas de A y C como cp3-origen.png y cp3-destino.png

CHECKPOINT3: suba los ficheros obtenidos en PC-A y PC-B (el C da igual) (puntuación 1%)

Muestre al profesor los resultados del checkpoint 2 y 3 y suba los ficheros