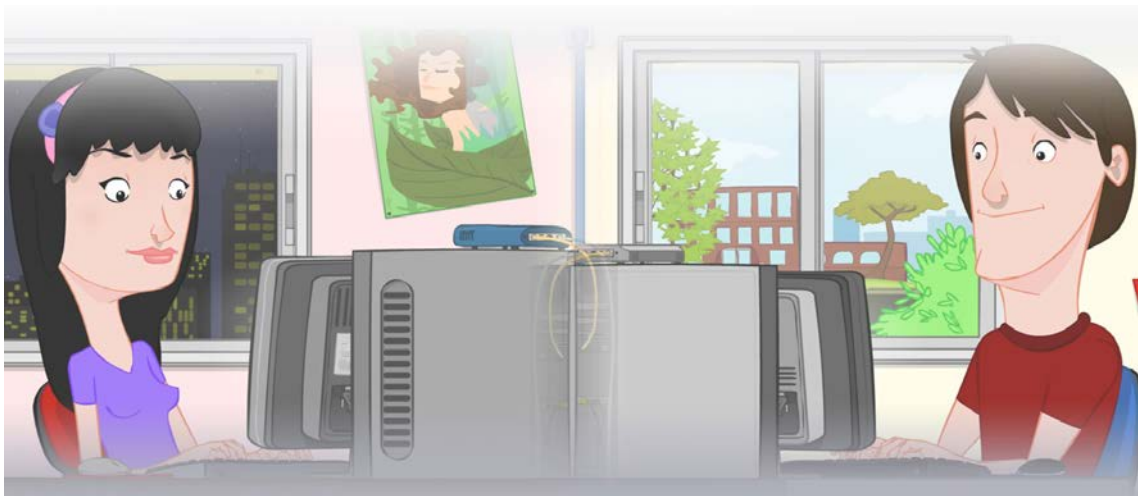


Práctica 1

Introducción a la Red de Acceso y Análisis de la QoS



Curso 2021-2022

**Grado en Ingeniería en Tecnología de
Telecomunicación**

(3^{er} Curso- Especialidad Telemática)

Área de Ingeniería Telemática

Índice

1	Descripción del laboratorio y normas de uso	3
2	Objetivo de la práctica	3
3	Conociendo la red y medidas de QoS en Linux	1
3.1	Introducción	1
3.2	Conociendo la red	1
3.3	Configurando la red interna	3
3.4	Midiendo retardos (delay)	3
3.4.1	Midiendo retardos en la red interna	3
3.4.2	Midiendo retardos en Internet	4
3.5	Midiendo throughput	5
4	Conociendo la red y medidas de QoS en Windows	7
4.1	Introducción	7
4.2	Conociendo la red	7
4.3	Configurando la red	8
4.4	Midiendo retardos	8
4.5	Midiendo throughput	8
5	Análisis de test de velocidad para estimación de la calidad en la red de acceso	9
5.1	Conceptos	9
5.2	Desarrollo de la práctica de test de velocidad	10
5.2.1	Test de Ookla: http://www.speedtest.net/	10
5.2.2	Test de RTR (Regulador Austríaco): https://www.netztest.at/en/Test	11
5.2.3	Conclusiones	11

ANEXOS

1	Estándares de QoS	1
2	IPERF/JPERF	3
2.1	Introducción a Iperf	3
2.2	Instalación IPERF	4
2.2.1	Instalación Iperf en LINUX	4
2.2.2	Instalación Iperf en WINDOWS	5
2.2.3	Instalación Jperf en WINDOWS	5
2.3	Ejemplos de uso de pruebas con iperf	5
2.4	Ejemplos de uso de pruebas con jperf	5
2.5	Algunas pruebas propuestas	6



Práctica 1: Análisis de la QoS en la Red de Acceso

1 Descripción del laboratorio y normas de uso

Todas las prácticas se desarrollarán en el laboratorio II del área de Ingeniería Telemática:

Nombre	Referencia Escuela	Localización
Labtel II	S1B5L	Planta sótano edificio B (bajo CCs)

El laboratorio cuenta con 24 puestos. Cada equipo disponga de 2 tarjetas de red. Una de ellas, la integrada en placa, está conectada a la red de experimentación que permite realizar prácticas de *switching/routing* sin afectar a la red de producción, correspondiente a la UPV/EHU (conectada la segunda tarjeta de red).

Por lo general, las rosetas vienen identificadas por un número 1-24 y una letra (ej. 23 A) en el caso de la red de experimentación y de una nomenclatura más extensa en el caso de la red de la escuela (ej. 02-3xxB). **En ningún caso se permite la manipulación de los latiguillos de conexión entre cada equipo y las rosetas ni cambiar la configuración de la red de producción (interfaz asociado a la red EHU).**

Tal como se muestra en el diagrama de la Figura 3 cada una de las interfaces de experimentación está conectada a una roseta y ésta parcheada en el patch-panel del armario de comunicaciones.



Figura 1.- Vista frontal y trasera del patch-panel

Cada equipo dispone de dos Sistemas Operativos instalados (Windows Nativo y Sistema Recomendado con Máquinas Virtuales- VirtualBox). En casi todas las prácticas de RdA utilizaremos una máquina virtual para que cada estudiante/asignatura pueda personalizar su sistema sin interferir con el resto.

La virtualización (<http://es.wikipedia.org/wiki/Virtualizaci%C3%B3n>) permite abstraer los recursos físicos de una máquina (como CPU, disco duro, tarjetas de red, etc..) pudiendo instalar un número indeterminado de sistemas huésped (máquinas virtuales) sobre el mismo hardware. Así, cada estudiante dispone de un PC completo, pero virtual, sobre el que realizar las modificaciones/instalaciones que precise. El precio a pagar por esa flexibilidad es que los sistemas virtualizados no suelen alcanzar las velocidades de ejecución de máquinas reales.

Para acceder al sistema de virtualización (VirtualBox) se deberá arrancar el sistema Windows10 **entrando con vuestro usuario LDAP**. Después se arrancará VirtualBox (haciendo doble click sobre el icono del escritorio).

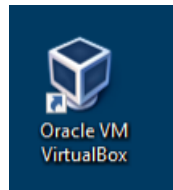


Figura 1.- Icono VirtualBox

Una vez en el sistema de máquinas virtuales, **el primer paso sera clonar las imágenes de RdA_Linux_base y RdA_Windows_base con el nombre RdA_Linux_GX y RdA_Windows_GX** y con X=1,31 en función del grupo general (p.e. RdA_Linux-G1). Se pincha con el botón derecho sobre la imagen RdA correspondiente y se pulsa en clonar, indicando el nuevo nombre. Se arrancarán las máquinas virtuales: **RdA_Linux_GX o RdA_Windows_GX**, según las indicaciones de la práctica requieran trabajar en Linux o Windows. El funcionamiento con las máquinas virtuales es sencillo. La tecla Ctrl derecha es una tecla especial que permite alternar entre diferentes modos de visualización del equipo Host/Guest. El resto de opciones las podéis ver en la Figura 5 extraída de <http://doc.ubuntu-es.org/VirtualBox>. Por precaución se aconseja no almacenar ninguna información vital en las MVs. Además, en algunos casos, para que la MV reconozca un stick USB éste debe estar conectado antes de arrancar el PC real y, aún en ese caso hay ocasiones en los que el sistema anfitrión no lo reconoce correctamente. Conclusión, es preferible utilizar métodos telemáticos (nunca mejor dicho) para enviar/recoger material al comienzo y final de la práctica.

Combinaciones de teclas

Las combinaciones de teclas más interesantes para usar en Virtualbox son:

- **CTRL(derecho)** : Para liberar el ratón y el teclado del sistema invitado y dar el control al sistema anfitrión.
- **CTRL(derecho) + F** : Para entrar o salir del modo pantalla completa.
- **CTRL(derecho) + L** : Para entrar o salir del seamless mode, un modo en el que se ve el sistema invitado maximizado, pero que no se ve el escritorio (a no ser que lo pidas explícitamente), si no sólo las ventanas de las aplicaciones y la barra de Windows/Linux. Este modo es un poco confuso. a mí no me agrada.
- **CTRL(derecho) + A** : Ajustar tamaño de ventana.
- **CTRL(derecho) + G** : Autoajustar la Ventana del anfitrión.
- **CTRL(derecho) + SUPR.** : Envía un CONTROL + ALT + SUPR. al sistema invitado (interesante si es Windows)
- **CTRL(derecho) + BACKSPACE** : Envía un CONTROL + ALT + BACKSPACE al sistema invitado (interesante si es GNU/Linux)
- **CTRL(derecho) + S** : Tomar una instantánea de la máquina virtual, como si apretáramos «Impr Pant».
- **CTRL(derecho) + I** : Inhabilitar la Integración del ratón con la máquina virtual.
- **CTRL(derecho) + R** : Para reiniciar la máquina virtual. Es el famoso «reset» windowsero. seguro que más de uno lo usa alguna vez. algooooo
- **CTRL(derecho) + P** : Para pausar la máquina virtual. Una ventaja si te está comiendo recursos en un momento en que necesitas todos los recursos de tu ordenador para el sistema anfitrión.
- **CTRL(derecho) + H** : Apagado ACPI de la máquina virtual.
- **CTRL(derecho) + Q** : Cerrar la máquina virtual.

De todos modos, tenéis los mismos accesos de forma gráfica en la barra de menús de la máquina virtual. En el extremo inferior derecho de la ventana de la máquina virtual tenéis iconos para controlar aspectos de vuestra máquina virtual y el botón que libera el ratón y el teclado del sistema invitado y da el control al sistema anfitrión (igual que Ctrl Derecho). (fig.26)

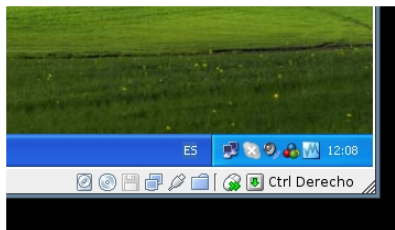


Figura 2.- Atajos de VirtualBox

2 Objetivo de la práctica

El objetivo de esta práctica consiste en el **estudio y análisis de la QoS de la red de acceso**.

Para ello, se trabajará sobre diferentes tipologías de redes de acceso: acceso a red pública (la de la casa del estudiante o similar), acceso a red corporativa (red LAN del laboratorio), etc.

Por un lado, se pretende que el estudiante **identifique y profundice en el conocimiento y análisis de la red y de los parámetros más importantes de QoS, como el throughput o el delay**. Con este fin se utilizarán diferentes herramientas de libre disposición y se desarrollará un análisis de los resultados obtenidos en las diferentes tipologías y tecnologías de redes.

Por otro lado, se analizarán las **metodologías de algunos de los test de velocidad de la red de acceso** que existen actualmente disponibles en el mercado, con objeto de analizar los resultados que proporcionan y discernir sobre la idoneidad de su uso y sus ventajas/desventajas. A continuación, se muestra un plan de trabajo que puede ayudar al estudiante a planificar el trabajo a lo largo de las dos semanas que dura la práctica. Es necesario tener en cuenta que las tareas llevan implícitas un trabajo personal y de grupo fuera del laboratorio que incluye tareas como lectura previa de la práctica, análisis de resultados obtenidos en el trabajo desarrollado en el laboratorio, redacción de informes, etc.

<u>SEMANA</u>	<u>DESCRIPCIÓN DEL TRABAJO PREVISTO POR SEMANAS</u>
Semana 24	Presentación y familiarización con la red del laboratorio. Medidas con las diferentes herramientas de rendimiento del retardo y throughput en Linux. Análisis de resultados en la red LAN (red corporativa) del laboratorio.
Semana 25	Medidas con las diferentes herramientas de rendimiento del retardo y throughput en Windows. Análisis de resultados en la red LAN (red corporativa) del laboratorio.
Semana 26	Pruebas con test de velocidad . Capturas para el análisis de la pruebas con Wireshark. Análisis de metodología y resultados obtenidos en las pruebas desarrolladas con los test de velocidad.
Trabajo "offline"	<ul style="list-style-type: none">• Cumplimentación de cuestionarios previos a la clase (red pública: red de acceso en la residencia del estudiante).• Desarrollo de pruebas adicionales a las desarrolladas en el laboratorio en el entorno de la red pública residencial del estudiante.• Cumplimentación del informe, de acuerdo con patrón entregado, para ser entregado 2 semanas después de la finalización de la práctica.

3 Conociendo la red y medidas de QoS en Linux

3.1 Introducción


En esta primera parte de la práctica nos familiarizaremos con el entorno y la red del laboratorio así como con el uso de herramientas básicas para la medida de los parámetros de QoS más importantes en las redes de acceso (ver en Anexo resumen global de parámetros).

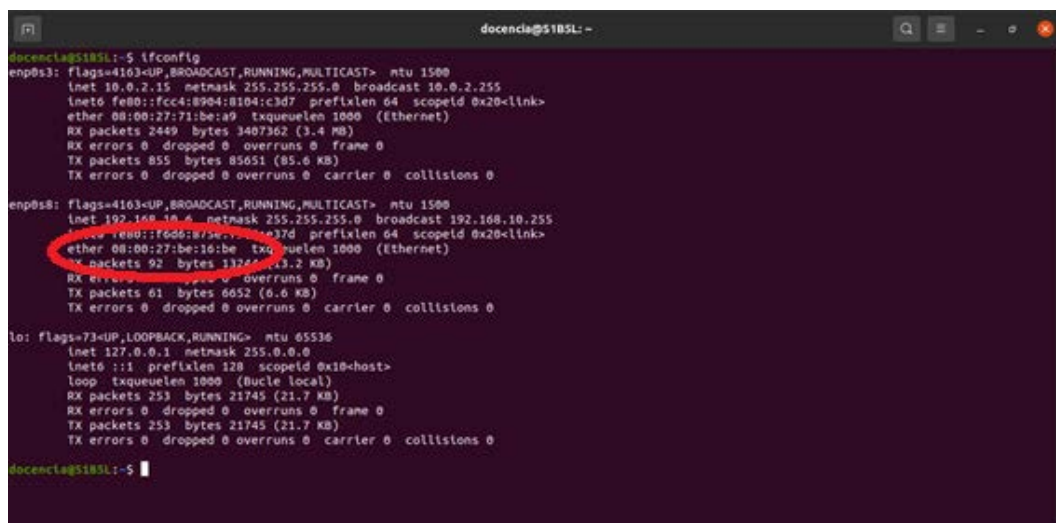
Para el desarrollo de esta parte de la práctica el estudiante arrancará el ordenador (**entrada con usuario LDAP**) y arrancará la máquina virtual **RdA_Linux_GX** del VirtualBox. El usuario que tenemos habilitado en esta máquina es:

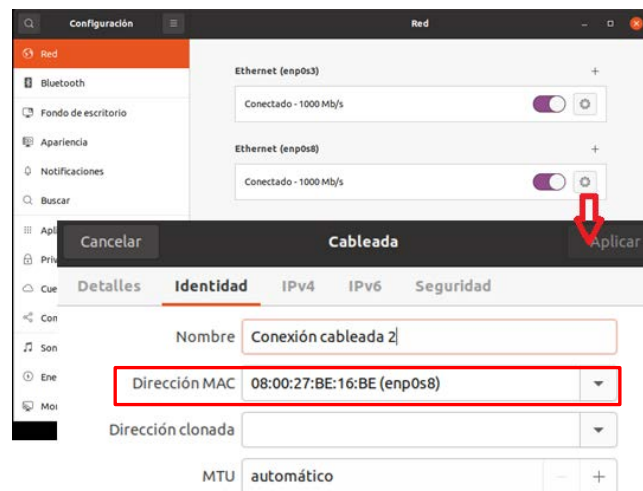
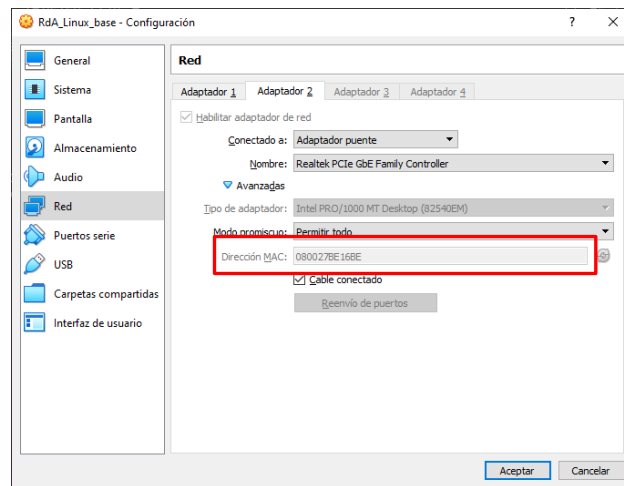
Username: docencia
Password: Docencia

3.2 Conociendo la red

Los puestos del laboratorio cuentan con varios interfaces de acceso a red (UNI). El objetivo de esta primera parte de la práctica es obtener la información necesaria para saber la configuración de sus correspondientes interfaces y la conectividad que nos proporcionan las mismas.

1. Arranca un terminal de comandos y averigua el nombre de tu ordenador en la red mediante el comando:
\$ hostname
2. Mediante el comando **\$ ifconfig** averigua los interfaces de red (Ethernet) habilitados en tu PC. Indica las características más relevantes de los mismos.
3. Analiza ahora esa misma información, pero utilizando la interfaz gráfica de Linux (desde el icono de la parte inferior  y con el botón derecho mostrar “preferencias de red” y desde la opción de Linux “Configuración -> Red”)
4. Comprueba que la dirección MAC del interfaz que no tiene acceso a la red de internet (la que no empieza por 10.0.2.X), es la misma en los resultados de los apartados 2 y 3 (ver figuras adjuntas). Si no fuera así debes cambiarla para ponerla congruente.





5. Identifica físicamente el acceso a los interfaces encontrados en apartados anteriores.

6. Utiliza el comando ping para enviar paquetes a un servidor web.

\$ ping SERVIDOR (p.e. www.google.com)

- a. ¿Has obtenido respuesta? (utilizar Wireshark para todos los análisis)
- b. ¿Por qué interfaz crees que se ha llevado a cabo la comunicación?
- c. ¿Qué velocidad de envío de datos te permite cada uno de los interfaces? Utiliza los comandos*:

\$ sudo mii-tool INTERFAZ

\$ sudo ethtool INTERFAZ

para averiguarlo y compara los resultados que te dan. ¿De qué velocidad estamos hablando: propagación/transmisión/procesamiento?

*Para ver las opciones de los comandos recuerda que puedes utilizar \$man ethtool

- d. Si dispones de un portátil (u otro dispositivo de datos), compara los interfaces que te aparecen en el ordenador del laboratorio y en tu dispositivo particular y comenta las diferencias más significativas.

Checkpoint 1: Muestra los resultados de este apartado al profesor/a e incorpóralos al informe de la práctica.

3.3 Configurando la red interna

En este apartado de la práctica vamos a configurar la interfaz de red de la red interna del laboratorio con una dirección IP Privada (**IMPORTANTE: configurar la que NO corresponde con la red EHU que nos proporciona el acceso a internet**).

Utilizad para ello la interfaz gráfica: en Configuración-> centro de redes y recursos compartidos.

Configuraremos en la interfaz de la red del Laboratorio la dirección: **192.168.10.X¹/24**

¹donde **X** es el número que figura en la toma de red interna correspondiente a tu PC.

(Nota- En el caso de que se quisiera hacer mediante comando se utilizaría el comando:


```
$ sudo ifconfig "nombre de interfaz" 192.168.10.X1/24
```

7. Comprueba que la interfaz se ha configurado correctamente (comando ifconfig).
8. Comprueba de nuevo la velocidad de los interfaces con el comando utilizado en el apartado 4c. ¿Puedes cambiar de alguna manera la velocidad que aparece en "speed"? Justifica tu respuesta (Pista: busca en internet que sucede al utilizar esos comando sobre máquinas virtuales).

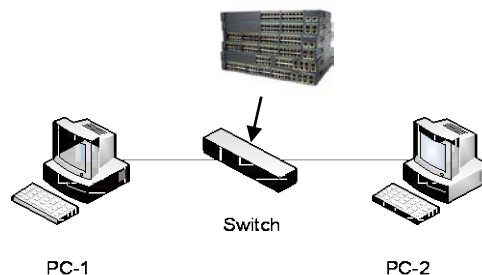
3.4 Midiendo retardos (delay)

3.4.1 Midiendo retardos en la red interna

En este apartado vamos a familiarizarnos con la medida y análisis del retardo en la red. Para ello analizaremos el envío de tramas Ethernet mediante el envío de tráfico utilizando el "ping" y **analizando lo que ocurre mediante Wireshark** (todas las pruebas deben ser capturadas con Wireshark²).

9. Comprueba que puedes hacer ping a la dirección IP configurada en la interfaz de tu ordenador que no pertenece a la red con conexión a Internet (EHU) y observa los tiempos de respuesta y explica el significado de todos los valores que te devuelve este comando. Vuelve a lanzar el ping pero aumentando el tamaño del paquete a 2000 bytes y compara los tiempos de respuesta (utiliza para ello el comando `$ ping -s 2000 192.168.10.X`. ¿Ha variado el tiempo en recibir la respuesta? Justifica la respuesta apoyándote en lo observado en tus capturas de Wireshark como base para las explicaciones (utiliza el filtro adecuado para capturar SÓLO el tráfico con origen/destino tu dirección IP). Puedes establecer el número de solicitudes de eco ICMP con la opción `-n` y con la opción `-i` será posible establecer el intervalo en segundos entre el envío de cada paquete y debes recoger algún resultado de la respuesta obtenida en la tabla que se adjunta en el informe. Prueba otras opciones del ping que tienes disponibles (ayuda del manual: `$man ping`
10. Conecta ahora tu PC al de tu pareja de laboratorio, utilizando el patch panel de interconexión del laboratorio para conectar las interfaces de red de la red interna. (la boca del patch panel corresponde con el número de tu toma de red interna)
 - a. Comprueba que puedes hacer ping a la dirección IP configurada en la interfaz de red de tu pareja de laboratorio y observa los tiempos de respuesta.

- b. Comprueba nuevamente la velocidad de las interfaces de red mediante el comando `$ sudo ethtool INTERFAZ` e indica las diferencias que observas frente a la velocidad que te aparece en la tarjeta de red asociada a este interfaz en la máquina anfitriona (Windows nativo) .
 - c. Envía de nuevo un ping al equipo de tu pareja de laboratorio pero con un paquete de 20000 bytes y compara los resultados. ¿Ha variado el tiempo en recibir la respuesta? Justifica la respuesta e indica qué tipos de retardos se están considerando. Indica así mismo si se produce fragmentación y a partir de qué valor.
11. Conecta ahora tu PC y el de tu pareja de laboratorio a dos bocas del switch asignado por la profesora (utilizar cada estudiante la boca de número igual que el de su toma red).



- a. *Comprueba que puedes hacer ping a la dirección IP configurada en la interfaz de red de tu pareja de laboratorio y observa los tiempos de respuesta. Analiza, como siempre, con Wireshark y compara los retardos que mide Wireshark con lo obtenidos con el ping. Analiza también los tamaños de los paquetes enviados y si existe fragmentación.*
- b. Envía de nuevo un ping al equipo de tu pareja de laboratorio pero con un paquete de 20000 bytes. Compara los resultados. ¿Ha variado el retardo? Justifica la respuesta.
- c. Realiza la misma prueba pero en este caso que sea tu pareja de laboratorio el que realiza el ping. ¿Ha variado el tiempo en recibir la respuesta? Justifica la respuesta.
- d. Repite la prueba pero estando ahora conectados tú y tu pareja de laboratorio a distintos switches (no olvides interconectar los switches).
- e. En el diagrama de tiempos incorporado en el informe, representa los tiempos o retardos obtenidos, representando los retardos que consideras pueden ser de: transmisión, propagación, procesamiento y cola.

3.4.2 Midiendo retardos en Internet

12. Ejecuta las siguientes pruebas (capturando con wireshark) e incorpora los resultados al informe.
- a. Comprueba que puedes hacer ping a las direcciones www.ehu.eus, www.ofcom.org.uk y www.whois.us y observa los tiempos de respuesta.
 - b. ¿A qué crees que se deben las diferencias? ¿Qué comando podrías utilizar para ver la ruta que siguen los paquetes?

- c. En la tabla incorporada al informe, representa los tiempos o retardos obtenidos, representando los retardos que consideras pueden ser de: transmisión, propagación, procesamiento y cola.
- d. Arranca el navegador y mide mediante la herramienta disponible en <http://pingtool.org> los mismos retardos que los desarrollados en los apartados anteriores. Compara los resultados y justifica si hubiera diferencias (incorpora los resultados a la tabla del apartado anterior).

Checkpoint 2: Muestra al profesor/a los resultados de este apartado e incorpóralos al informe de la práctica (no olvidar incorporar capturas de wireshark).

3.5 Midiendo throughput

En este apartado de la práctica vamos a analizar el rendimiento del throughput (capacidad/caudal/velocidad) en un entorno de laboratorio. Para ello, inicialmente comprobaremos el límite de velocidad de la tarjeta Ethernet mediante la red más sencilla establecida entre nuestros ordenadores con un cable y, posteriormente, desarrollaremos el análisis en redes más complejas con equipamiento de red intermedios para estimar el efecto que los elementos de red pueden tener en el cómputo del throughput.

En este caso vamos a utilizar la herramienta **iperf3 y jperf** para transmitir datos continuamente entre los dos ordenadores y poder observar el tráfico y medir el throughput. La herramienta **iperf3**, debe ejecutarse en los dos ordenadores, ya que uno actúa en modo cliente y el otro en modo servidor. Esta herramienta nos permite enviar y recibir datos, estableciendo si queremos utilizar el protocolo fiable TCP o no fiable UDP, así como otros parámetros de envío (velocidad, tamaño paquete...). En base a estos envío de datos y su recepción el throughput utilizando **Wireshark junto con iperf3**. Si el programa no se encuentra instalado en nuestro ordenador, comenzaremos por la instalación del programa de acuerdo con lo especificado en el Anexo (sección 2).

13. Ejecutaremos, al menos, las siguientes pruebas:

- a. Configura en un ordenador un servidor **iperf3** para **recibir datos con transporte fiable en el puerto 10000**. Eso se hace con el comando:

```
$ iperf3 -s -p 10000
```

El ordenador se quedará a la escucha de recibir datos.

- b. En este mismo ordenador pero en otro terminal configuraremos también otro servidor **iperf3** para **recibir datos con transporte no fiable en el puerto 20000**. Eso se hace con:

```
$ iperf3 -s -p 20000
```

(el modo TCP o UDP se establece en el cliente y cuando intercambia los parámetros para el test con el servidor es cuando acuerdan el uso de TCP o UDP).

- c. Configuraremos ahora el **envío de datos en el otro ordenador**. Configuramos uno de los ordenadores para enviar datos durante 15 segundos hacia el ordenador de tu pareja de laboratorio mediante los comandos (TCP y UDP):

```
$ iperf3 -c 192.168.10.X -p 10000 -t 15
```

```
$ iperf3 -c 192.168.10.X -u -p 20000 -t 15
```

Analiza la información que nos proporciona y para qué sirven cada una de las opciones utilizadas en el comando (-c, -p, -t...) e indica otras opciones que pueden ser interesantes de cara a la utilización del comando. Indica qué diferencias encuentras en los resultados utilizando UDP y TCP y justifica a qué se deben realizando las modificaciones necesarias para obtener datos comparativos.

- d. Una vez comprobado cómo funciona iperf3 vamos a lanzar diferentes envíos de datos **en ambos ordenadores** con Wireshark arrancado y comparar los datos de throughput que nos da iperf3 y wireshark tanto en TCP como en UDP (recordad configurar adecuadamente las máquinas como servidores en el puerto correspondiente). Para ello lanzar, al menos, los siguientes envíos:

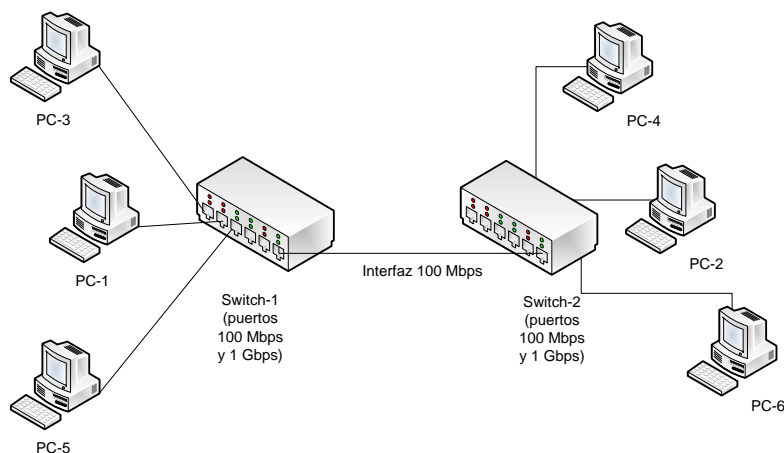
```
$ iperf3 -c 192.168.10.X -p 10000 -t 10 -b 1Mbps
```

```
$ iperf3 -c 192.168.10.X -p 10000 -t 10 -b 100Gbps
```

```
$ iperf3 -c 192.168.10.X -u -p 10000 -t 10 -b 1Mbps
```

```
$ iperf3 -c 192.168.10.X -u -p 10000 -t 10 -b 100Gbps
```

14. De acuerdo con otra(s) pareja(s) de laboratorio (grupos de seis por ejemplo), desarrollaremos ahora la misma prueba pero utilizando todos los ordenadores los mismo switches. Comparar los resultados obtenidos con las otras parejas de laboratorio realizando envíos también.



Checkpoint 3: Muestra al profesor/a los resultados de este apartado e incorpóralos al informe de la práctica (no olvidar incorporar capturas de wireshark).

4 Conociendo la red y medidas de QoS en Windows

4.1 Introducción

En esta segunda parte de la práctica con el uso de herramientas básicas para la medida de los parámetros de QoS más importantes en las redes de acceso pero ahora sobre Windows. Para el desarrollo de esta parte de la práctica el estudiante arrancará la máquina virtual **RdA_Windows_GX** del VirtualBox El usuario que tenemos habilitado en esta máquina es:

Username: docencia

Password: Docencia.

4.2 Conociendo la red

Comenzaremos arrancando la máquina virtual **RdA_Windows_GX** y desarrollaremos las siguientes pruebas.

15. Mediante el comando **\$ipconfig** averigua los interfaces de red (Ethernet) habilitados en tu PC. Indica las características más relevantes de los mismos y compáralos con la información obtenida desde LINUX y con la obtenida a partir de la información de la red que nos proporciona windows (Panel de control-> ver propiedades de red-> mostrar propiedades de cada una de ellas con botón derecho). Verifica y prueba algunas de las opciones del comando **\$ipconfig** (como la opción **\$ipconfig/all**).
16. Identifica físicamente el acceso a los interfaces encontrados en el apartado anterior en tu ordenador.
17. Utiliza el comando ping para enviar paquetes a un servidor web (p.e. www.google.com).
 - a. ¿Has obtenido respuesta?
 - b. ¿Por qué interfaz crees que se ha llevado a cabo la comunicación? Si no estás seguro prueba a desactivar un interfaz de red y una vez deshabilitado uno de ellos prueba a realizar de nuevo la prueba del ping y comprueba el resultado (no olvides volver a habilitarlo).
 - c. ¿Qué velocidad de envío de datos te permite cada uno de los interfaces (Revisar nuevamente en propiedades de red) ¿De qué velocidad estamos hablando: propagación/transmisión/procesamiento?
 - d. Comprueba que la dirección MAC de tus interfaces está congruente (es decir, la información que aparece con el comando **\$ ipconfig** es igual que la que aparece en la opción de la interfaz gráfica-> propiedades de red).

Checkpoint 4: Muestra los resultados de este apartado al profesor/a e incorpóralos al informe de la práctica.

4.3 Configurando la red

En este apartado de la práctica verificaremos cómo es posible cambiar la información y configuración de mis interfaces de red en Windows (de nuevo en la máquina virtual de Windows RdA_Windows_GX).

18. Incorpora al informe la pantalla desde donde establecerías la configuración de tus interfaces de red en Windows. Si se cuenta con el tiempo necesario, configurar la interfaz de la red interna del laboartorio como se hizo en la parte de Linux.

4.4 Midiendo retardos

19. Ejecuta las siguientes pruebas (capturando con wireshark) e incorpora los resultados al informe.
 - a. Comprueba que puedes hacer ping (con la herramienta ping de windows) a las direcciones www.ehu.eus, www.ofcom.org.uk y www.whois.us y observa los tiempos de respuesta. Compara los resultados obtenidos a los obtenidos en linux.
 - b. En la tabla incorporada al informe del apartado 12, representa los tiempos o retardos obtenidos, representando los retardos que consideras pueden ser de: transmisión, propagación, procesamiento y cola. Si se cuenta con el tiempo necesario, realizar las pruebas de ping con la pareja como se hizo en la parte de Linux .
 - c. Arranca el navegador y mide mediante la herramienta disponible en <http://pingtool.org> los mismos retardos que los desarrollados en el apartado anterior. Compara resultados.

4.5 Midiendo throughput

De acuerdo con lo recogido en el Anexo, instalaremos el programa iperf3 y jperf en windows y buscaremos un servidor iperf activo (consultar con el profesor/a si no se encuentra ninguno).

20. Ejecuta pruebas similares a las ejecutadas en la parte de Linux (capturando con wireshark) e incorpora los resultados al informe.

- Resultados obtenidos

Checkpoint 5: Muestra los resultados de este apartado al profesor/a e incorpóralos al informe de la práctica.

5 Análisis de test de velocidad para estimación de la calidad en la red de acceso

5.1 Conceptos

En la guía de referencia [ETSI EG 202 057-4 V1.2.1](#), accesible desde la página web del ETSI www.etsi.org y presentada en el seminario 2 por algunos estudiantes, se especifican los indicadores de QoS y metodologías de medida que se contemplan para el servicio de acceso a Internet. A continuación se recoge un extracto de estos parámetros, entre los que se encuentran los analizados en el apartado anterior:

Parameter	Measure	Measurement Method	Application
Login time	number of successful log-ins	Test calls	all IAP services that are accessed via a login process
Data transmission speed achieved	a) The maximum data transmission rate in kbit/s achieved. b) The minimum data transmission rate in kbit/s achieved. c) The mean value and standard deviation of the data transmission rate in kbit/s.	Test calls	all IAP
Unsuccessful data transmissions ratio	% unsuccessful data transmission	Test calls	all IAP
Successful log-in ratio	% Successful log-ins	Test calls	all IAP services that are accessed via a login process
Delay (one way transmission time)	a) The mean values of the delay in milliseconds. b) The standard deviation of the delay.	Test calls	all IAP

Sin embargo, tal y como también se presentó en el seminario 2, nuestro regulador, la [CNMC](#), ha establecido que, de acuerdo con la regulación establecida en la orden de calidad [IET/1090/2014](#), a partir de 2014, de estos parámetros recogidos en la guía ETSI específicos para el servicio de acceso a Internet, se medirá y publicará (ver en [Publicaciones QoS](#)) sólo **la velocidad de transmisión conseguida**. Se indica así mismo en esta orden ministerial que, para la medida de este parámetro, serán de aplicación la definición y métodos de medidas recogidos en la sección 5.2 del documento [ETSI EG 202 057-4](#), con la siguiente matización: *“La velocidad de transferencia de datos conseguida se calculará con relación a la velocidad de transmisión de datos obtenida por los usuarios tanto desde un servidor remoto a su ordenador o equipo terminal como en sentido contrario, es decir, desde su ordenador o equipo terminal hacia un servidor remoto.”*

Aún cuando para estas medidas de QoS los operadores utilizan sondas de prueba ([documentación de referencia](#) documento: *“Criterios adicionales para la medición de los parámetros de calidad de servicio específicos para el servicio de acceso a Internet”*), existen actualmente multitud de sistemas o *“Test de Velocidad”* a disposición de los usuarios que nos proporcionan datos y resultados relacionados con la “velocidad de transferencia de datos conseguida” (throughput) y también con los retardos.

En la segunda parte de esta práctica desarrollaremos un **análisis de varias de estas herramientas de este tipo con el objetivo de que el estudiante sea consciente de las diferentes metodologías utilizadas** y la razón por la que los resultados obtenidos con unas y otras pueden ser diferentes. Así mismo, el estudiante reflexionará para realizar una comparativa de estas metodologías y las utilizadas en las secciones 2 y 3 de la práctica.

5.2 Desarrollo de la práctica de test de velocidad

Para el desarrollo de esta parte de la práctica el estudiante trabajará sobre el Windows nativo.

En las páginas que se indican a continuación se encuentran disponibles los enlaces a algunas de las alternativas disponibles en Internet para estimar la QoS de nuestra conexión a Internet.

<http://www.speedtest.net/>

<https://www.netztest.at/en/Test>

Se requiere desarrollar el análisis que se recoge en los apartados siguientes relativos a estas herramientas. Adicionalmente, el estudiante podrá desarrollar un análisis similar sobre otras herramientas que él utilice habitualmente o que le suministra su proveedor. Las pruebas se desarrollarán en el Windows Virtual Box.

5.2.1 Test de Ookla: <http://www.speedtest.net/>

La metodología que utiliza este test se encuentra especificada en: <https://support.speedtest.net/hc/en-us/articles/203845400-How-does-the-test-itself-work-How-is-the-result-calculated-> y vamos a analizarla con un ejemplo en base a vuestro análisis sobre Wireshark.

Se responderán en el informe, al menos, a las siguientes cuestiones para ayudar en la definición de la metodología:

21. ¿Qué parámetros mide la herramienta?
22. ¿Cuál es el procedimiento de medida para cada parámetro?
23. ¿Has encontrado algún problema a la hora de analizar con Wireshark la metodología que utiliza?
24. ¿Contra qué recurso/recursos se realiza la medida? Si es seleccionable, desarrolla pruebas contra el más cercano y otros dos (uno en Europa y otro en EEUU) y compara resultados e incorpóralos a la tabla del informe. Contrasta los resultados obtenidos con los resultados que podrás obtener con las herramientas analizadas en la sección 3 de la práctica.
25. ¿Puedes saber si se utilizan conexiones en paralelo durante la medida o una sola conexión TCP? Justifícalo.
26. ¿Existe alguna fase de “pre-test” para alcanzar una situación de estabilidad (recordad el slow-start) antes de que se tomen valores para la medida?
27. Justifica la diferencia que ves con las pruebas realizadas en los apartados anteriores y qué ventajas/desventajas puede tener esta metodología frente a las anteriores.
28. ¿Qué tratamiento se hace de los resultados obtenidos en la medida de cara a presentar un resultado final al usuario del test?
29. Realizar en la red de acceso de vuestras casas/residencia una prueba con este test y contrastar con las pruebas y resultados desarrollados para el cuestionario inicial de la práctica. Incorpora todos los resultados al informe.

Checkpoint 6: Muestra los resultados de este apartado al profesor/a e incorpóralos al informe de la práctica.

5.2.2 Test de RTR (Regulador Austríaco): <https://www.netztest.at/en/Test>

La metodología utilizada en este test se encuentra especificada en <https://www.netztest.at/doc/> y como queda descrito en <https://www.rtr.at/en/tk/netztestmethodik> también podéis acceder a la descripción del test opcional **NDT** en <https://www.rtr.at/en/tk/netztestmethodik>. Este test está basado en la metodología descrita en el capítulo 6 del estándar [ETSI TS 103 222-1](#).

Partiendo de estos datos, **desarrolla diferentes pruebas de QoS** con esta herramienta para cada una de las tecnologías de red de acceso disponibles en el laboratorio, realiza capturas con el Wireshark y compara los resultados obtenidos con la metodología propuesta en el estándar respondiendo a las siguientes preguntas:

30. ¿Qué parámetros mide?
31. ¿Cuál es el procedimiento de medida para cada parámetro en base a lo que se describe en el estándar? ¿Has encontrado algún problema a la hora de comprobar con Wireshark que, efectivamente, esa es la metodología que se utiliza?
32. ¿Contra qué recurso/recursos se realiza la medida?
33. ¿Existe alguna fase de “pre-test” para alcanzar una situación de estabilidad (recordad el slow-start) antes de que se tomen valores para la medida?
34. Justifica la diferencia que ves con las pruebas realizadas en los apartados anteriores.
 - Desarrolla un estudio de los resultados obtenidos similar al desarrollado en el último punto de las pruebas del apartado anterior, comparándolos.

5.2.3 Conclusiones

Teniendo en cuenta las pruebas realizadas con los test de

35. Resume las conclusiones del estudio realizado en la comparativa de los test de velocidad utilizados en base a la tabla que se recoge en el informe.
36. Desarrolla un estudio de los resultados obtenidos, tanto en las pruebas desarrolladas en el laboratorio, como en las pruebas indicadas en el cuestionario previo a la práctica u otras similares a las desarrolladas en el laboratorio pero sobre vuestra red de acceso en casa con la tecnología cableada (XDSL, HFC, FTTx) y también extendida a vuestras redes LAN (Wifi, PLC...) y extrae las conclusiones que consideres de interés, incluyendo, al menos:
 - Conclusiones respecto a los resultados obtenidos en la red corporativa y en vuestra red pública residencial.
 - Conclusiones respecto a los resultados obtenidos con esta metodología en tu red pública y las publicadas por tu proveedor en la web del regulador.

Checkpoint 7: Muestra los resultados de este apartado al profesor/a e incorpóralos al informe de la práctica.

Anexo Práctica 1



Grado en Ingeniería en Tecnología de
Telecomunicación
(3^{er} Curso- Especialidad Telemática)

Área de Ingeniería Telemática

Índice

1	Estándares de QoS	1
2	IPERF/JPERF	3
2.1	Introducción a Iperf.....	3
2.2	Instalación IPERF	4
2.2.1	Instalación Iperf en LINUX.....	4
2.2.2	Instalación Iperf en WINDOWS	5
2.2.3	Instalación Jperf en WINDOWS.....	5
2.3	Ejemplos de uso de pruebas con iperf	5
2.4	Ejemplos de uso de pruebas con jperf	5
2.5	Algunas pruebas propuestas	6

Parámetros de QoS



1 Estándares de QoS

En esta práctica probaremos el software `iperf3` para la generación de flujos de tráfico. Este es un software muy simple pero que nos servirá en las prácticas y asignaturas posteriores para la evaluación de la QoS por lo que el objetivo en esta práctica es que el estudiante se familiarice con su uso y prestaciones. Emplearemos la versión 3 del programa, la cual es incompatible con versiones anteriores. El motivo de emplear esta versión es que ofrece algunas opciones no disponibles en versiones anteriores.

extenderemos el estudio desarrollado sobre algunos de los parámetros de la guía de referencia [ETSI EG 202 057-4](#), a otros parámetros que nos permiten estimar la QoS de forma más precisa en Internet, como los contemplados en la recomendación de la serie [Y.1540](#):

- Retardo - IP packet transfer delay (IPTD)
- Tasa de pérdidas de paquetes - IP packet error ratio (IPER)
- Tasa de paquetes erróneos - packet loss ratio (IPLR)
- Tasa de paquetes espurios - Spurious IP packet rate (SIPR)
- Variación del retardo- IP packet delay variation (IPDV)
- Disponibilidad - IP service availability

También debemos tener en cuenta la extensión más amplia de parámetros recogidos en la [ETSI EG 765-3](#) que engloba los parámetros recogidos en algunos de los estándares más importantes:

	IETF RFCs	ITU-T Recommendations
Framework	RFC 2330 [i.3]	Y.1540 [i.1], sections 1 through 5
Loss	RFC 2680 [i.6]	Y.1540 [i.1], section 5.5.6 G.1020 [i.23]
Delay	RFC 2679 [i.5] (One-way) RFC 2681 [i.7] (Round Trip)	Y.1540 [i.1], section 6.2 G.1020 [i.23] G.114 [i.22] (One-way)
Delay Variation	RFC 3393 [i.10]	Y.1540 [i.1], section 6.2.2 G.1020 [i.23]
Connectivity / Availability	RFC 2678 [i.4]	Y.1540 [i.1], section 7
Loss Patterns	RFC 3357 [i.9]	G.1020 [i.23]
Packet Reordering Packet Duplication	RFC 4737 [i.15]	Y.1540 [i.1], sections 5.5.8.1 and 6.6 Y.1540 [i.1], sections 5.5.8.3, 5.5.8.4, 6.8, and 6.9
Link/Path Bandwidth Capacity, Link Utilization, Available Capacity	RFC 5136 [i.31]	
Bulk Transport Capacity	RFC 3148 [i.8], RFC 5136 [i.31]	

Tabla 1.1. Resumen de estándares analizados en la guía ETSI EG 765-3

Por último, y dado que estamos que queremos centrar nuestro interés en la medida de rendimiento para las redes de acceso, destacar también la lista de parámetros que se recogen en la recomendación de la UIT-T Y.1292 que lleva por título “Redes IP Parametrizables (RIP): Marco para los requerimientos y aptitudes relacionadas con la parametrización de las redes para los servicios IP por parte de los usuarios finales”. En esta recomendación se consideran

relevantes, en el contexto de las aptitudes de la red que deben poderse establecer en base a los requerimientos de los usuarios, la siguiente lista de parámetros:

- Tasa de pérdidas de paquetes/Tasa de paquetes erróneos
- Retardo en un sentido/Retardo de ida y vuelta/Variación del retardo
- Disponibilidad (p.e.: Tiempo de puesta en marcha, tiempo medio entre fallos, tiempo medio de reparación del servicio)
- Valor máximo del ancho de banda/Ancho de banda disponible/Valor mínimo de ancho de banda
- Retardo de ida y vuelta en la localización e identificación para la movilidad
- Probabilidad de bloqueo en el acceso y de finalización del servicio
- Estadísticas relacionadas con el tráfico (p.e.: número de paquetes a transmitir o recibir, número de paquetes rechazados o recibidos con error)

En definitiva, de todo lo anteriormente mencionado se puede concluir que, en cuanto a los parámetros a tener en cuenta para la medida del funcionamiento de la red en Internet, las normativas y estándares parecen coincidir en que los más importantes son: **conectividad, disponibilidad, retardo, variación del retardo, pérdidas de paquetes y velocidad de transmisión.**

Por otro lado, en la Recomendación [Y.1541](#) se establecen los objetivos en cuanto los parámetros anteriormente enunciados para los diferentes servicios y clases de servicio.

Network performance parameter	Nature of network performance objective	QoS Classes					
		Class 0	Class 1	Class 2	Class 3	Class 4	Class 5 Unspecified
IPTD	Upper bound on the mean IPTD (Note 1)	100 ms	400 ms	100 ms	400 ms	1 s	U
IPDV	Upper bound on the $1 - 10^{-3}$ quantile of IPTD minus the minimum IPTD (Note 2)	50 ms (Note 3)	50 ms (Note 3)	U	U	U	U
IPLR	Upper bound on the packet loss probability	1×10^{-3} (Note 4)	1×10^{-3} (Note 4)	1×10^{-3}	1×10^{-3}	1×10^{-3}	U
IPER	Upper bound	1×10^{-4} (Note 5)					U

Tabla 1.2. Clases de de calidad de servicio y objetivos de funcionamiento para servicios IP de la Rec. Y.1541

IPERF/JPERF



2 IPERF/JPERF

En esta práctica probaremos el software `iperf3` para la generación de flujos de tráfico. Este es un software muy simple pero que nos servirá en las prácticas y asignaturas posteriores para la evaluación de la QoS por lo que el objetivo en esta práctica es que el estudiante se familiarice con su uso y prestaciones. Emplearemos la versión 3 del programa, la cual es incompatible con versiones anteriores. El motivo de emplear esta versión es que ofrece algunas opciones no disponibles en versiones anteriores.

2.1 Introducción a Iperf

Este programa se emplea para generar flujos de paquetes en una red IP que nos permitirán obtener estadísticas para la medida de algunos de los parámetros de QoS: throughput, latencia, jitter, pérdidas. Los paquetes los envía a otro proceso ejecutando el mismo programa `iperf`, normalmente en otra máquina. A uno de los extremos se le llama el “cliente” y al otro el “servidor”, y siguiendo esa filosofía cliente-servidor se lanza primero el segundo en una máquina y a continuación el cliente en otra. El cliente contacta con el servidor (mantienen una conexión de control) y a continuación hacen los envíos de tráfico. Si queréis ver de manera más gráfica los resultados de `iperf`, podéis utilizar una aplicación creada explícitamente para ello, se llama **JPerf**. JPerf puede ser asociado con `Iperf` para tener una interfaz gráfica escrita en Java.

Antes de desarrollar las pruebas, es conveniente haber consultado la documenta relacionada con los comandos y el funcionamiento del programa. Se indican a continuación el enlace de ayuda al manual:

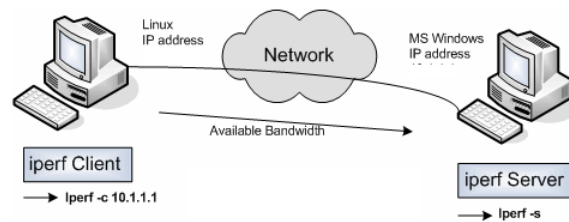
- <https://code.google.com/p/iperf/wiki/ManPage>

Y otras páginas de ayuda:

- <https://iperf.fr/>
- <https://fasterdata.es.net/performance-testing/network-troubleshooting-tools/iperf-and-iperf3/>
- <http://castello.guifi.net/content/servidor-iperf-en-castelloguifinet-testeando-la-velocidad-y-el-estado-de-la-red>
- <http://www.redeszone.net/redes/iperf-manual-para-medir-ancho-de-banda-entre-dos-ordenadores-en-lan/>

2.2 Instalación IPERF

Iperf puede ser instalado muy fácilmente en cualquier sistema basado en UNIX/Linux o Microsoft Windows. Un host debe tener la función de cliente y otro la de servidor.



2.2.1 Instalación Iperf en LINUX

- Instalación sencilla:

```
$ sudo apt-get install iperf3
```

- Compilación manual (iperf):

Se clona el repositorio de software mediante la herramienta git. En caso de que no se disponga del comando git se instalará mediante apt-get (como super usuario o usando sudo).

```
$ sudo apt-get install git  
$ git clone http://github.com/esnet/iperf.git  
$ cd iperf  
$ ./configure && make
```

Si se produce un error en el comando anterior es posible que no se haya instalado el compilador o entorno de desarrollo.

```
$ sudo apt-get install build-essential
```

AVISO IMPORTANTE : No se instalará (**NO hacer \$ sudo make install**) para disponer de él en el directorio correspondiente por cada grupo. Concretamente los binarios generados se sitúan en el directorio \$HOME/iperf/src

Para empezar a trabajar con iperf se accederá a la carpeta en la que está el programa a ejecutar.

```
-$ cd $HOME/iperf/src
```

Dado que el funcionamiento de la herramienta iperf es cliente-servidor, en el equipo que funcionará como servidor se ejecutará el siguiente comando, para poner el iperf3 a la escucha de conexiones entrantes.

```
$ ./iperf -s
```

Desde el equipo cliente, se ejecutarán las pruebas utilizando las diferentes opciones que ofrece el comando.

2.2.2 Instalación Iperf en WINDOWS

En la página <https://iperf.fr/> o <http://sourceforge.net/projects/iperf/files/iperf-2.0.5.tar.gz/download> tenéis el software disponible para la instalación en windows.

2.2.3 Instalación Jperf en WINDOWS

En la página <https://code.google.com/p/xjperf/downloads/list> tenéis el software disponible para la instalación en windows.

Instalación:

- Descomprimos el .zip.
- Ejecutamos el .bat

2.3 Ejemplos de uso de pruebas con iperf

TCP test: `$./iperf3 -c iperf.scottlinux.com`

UDP test: `$./iperf3 -c iperf.scottlinux.com -u`

Specify IPv4 or IPv6:

`$./iperf3 -4 -c iperf.scottlinux.com`

`$./iperf3 -6 -c iperf.scottlinux.com`

Ejemplo de subida TCP básica (sustituir IPSEVIDOR e Y por lo que corresponda)

`$./iperf3 -4 -c IPSEVIDOR -p 510Y`

Ejemplo de bajada con N conexiones TCP en sentido descendente (sustituir IPSEVIDOR e Y por lo que corresponda)

`$./iperf3 -4 -c IPSEVIDOR -P N -R -p 510Y`

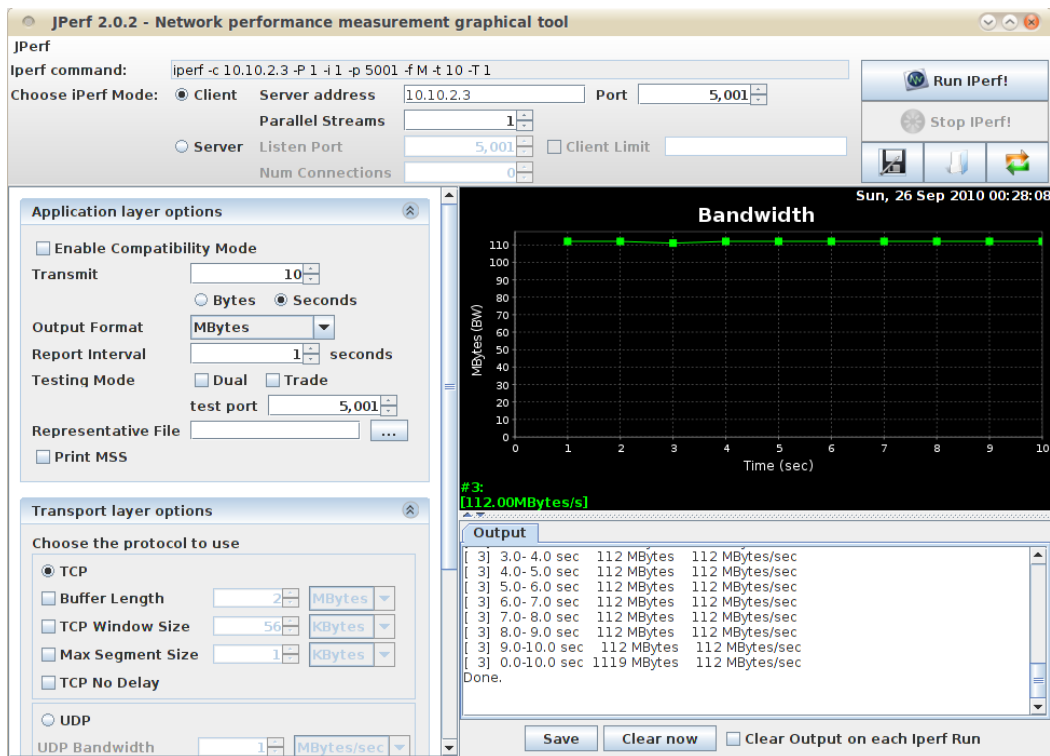
- [Servidores Iperf públicos:](#)

En <http://en.wikipedia.org/wiki/Iperf> y <https://iperf.fr/iperf-servers.php> podéis encontrar algunos servidores de Iperf públicos.

Nota- No siempre están disponibles. Si no lo están, intentar montaros vuestro propio servidor a través de los espacios que os pueden proporcionar entornos de colaboración como Github)

2.4 Ejemplos de uso de pruebas con jperf

En la interfaz del Jperf, seleccionaremos entre cliente y servidor e introducimos el comando iperf correspondiente, cuando lo hayamos seleccionado le damos a **RUN JPERF**.



2.5 Algunas pruebas propuestas

\$ iperf -c IP_Address -P 1 -i 1 -p 5001 -f m -t 10

- -c significa que estás ejecutando en modo cliente, conectándote al "host" que tiene la IP_Address que sigue al -c.
 - -P 1 indica el nº de conexiones en paralelo que estás lanzando.
 - -i 1 signica los segundos entre graficado y reporte del BW.
 - -p 5001 identifica el puerto cliente/servidor en el que se está escuchando o al que se está conectado.
 - -fm es el formato en el que te gustaría obtener los resultados in Kbits, Mbits, KBytes, Mbytes, and Gbits.
 - -t 10 tiempo en segundos entre pruebas. Por defecto 10 segundos.
- Compara el rendimiento de TCP realizando la misma prueba con el valor por defecto del tamaño máximo del buffer de recepción y otros valores 16KB 48KB 128KB...
 - Realiza un test de descarga. ¿Qué valor da de throughput?. Repite el procedimiento con varias conexiones TCP en paralelo y explica el resultado en caso de que sea diferente.