REDES:
Ejercicios de Definiciones de Rendimiento y Mediciones
==========

Notas:
---
* Comandos precedidos con "$" implican que Ud. debe ejecutar el comando como un usuario general no como root.
* Comandos precedidos con "#" implican que Ud. debe ejecutar el comando como root
* Comandos con lineas de comando mas especificas (ej. "GW-RTR>" o "mysql>") implican que Ud. esta ejecutando el comando en un dispositivo remoto, o dentro de otro programa
* Si un comando termina con "\" indica que el comando continua en la linea siguiente, y que Ud. debe considerar las dos lineas como una sola secuencia

Ejercicio Parte I
==========

0. Login a su servidor virtual

NOTA: Durante el ejercicio si Ud. encuentra que el comando
apt-get da error, entonces debe actualizar la base de datos de apt. Para ello:

```
$ sudo apt-get update
```

Metricas de Rendimiento de Red
---

1. ping
---

ping es un programa wue envia paquetes ICMP de tipo "solicitud de eco", y espera "respuesta de eco" procedente de la entidad encuestada. Dependiendo del sistema operativo usado, puede ser que vea la demora de retorno minima, maxima, y mediana, y en algunas casos la desviacion estandar de la mediana en las respuestas ICMP de la entidad encuestada.
Para mas detalles:

http://en.wikipedia.org/wiki/Ping

Bloquear ping a nivel de firewall es generalmente una mala idea.
Tomando en cuenta lo anterior, trate de usar ping de varias formas:

```
$ ping localhost
```

Presione ctrl-c para detener el proceso. Aqui va la respuesta tipica a este comando:

```
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time0.020 ms
    64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.006 ms
    64 bytes from localhost (127.0.0.1): icmp_seq=3 ttl=64 time=0.006 ms
    64 bytes from localhost (127.0.0.1): icmp_seq=4 ttl=64 time=0.006 ms
    64 bytes from localhost (127.0.0.1): icmp_seq=5 ttl=64 time=0.006 ms
    64 bytes from localhost (127.0.0.1): icmp_seq=6 ttl=64 time=0.009 ms
```

```
64 bytes from localhost (127.0.0.1): icmp_seq=7 ttl=64 time=0.007 ms
^C
--- localhost ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 5994ms
rtt min/avg/max/mdev = 0.006/0.008/0.020/0.005 ms
```

Pregunta: Por que la respuesta al primer paquete toma 20ms meintras que el resto de las respuestas es mucho mas rapido? Este es un tipo de demora. Que tipo de demora es?

2. traceroute

Alguna vez ha usado traceroute? Ha revisado en detalle como funciona?
Si no lea aqui:

http://en.wikipedia.org/wiki/Traceroute

Puede ser que necesite instalar el paquete traceroute antes de poder usarlo. De esta forma:

```
$ sudo apt-get install traceroute
```

Una vez installado:

```
$ traceroute nsrc.org
```

Aqui vemos una muestra de respuesta de traceroute a nsrc.org (lineas se parten debido a la longitud de linea):

```
traceroute to nsrc.org (128.223.157.19), 64 hops max, 52 byte packets
 1  gw.ws.nsrc.org (10.10.0.254)  1.490 ms  1.069 ms  1.055 ms
 2  192.248.5.2 (192.248.5.2)  2.741 ms  2.450 ms  3.182 ms
 3  192.248.1.126 (192.248.1.126)  2.473 ms  2.497 ms  2.618 ms
 4  mb-t3-01-v4.bb.tein3.net (202.179.249.93)  26.324 ms  28.049 ms  27.403 ms
 5  sg-so-06-v4.bb.tein3.net (202.179.249.81)  103.321 ms  91.072 ms  91.674 ms
 6  jp-pop-sg-v4.bb.tein3.net (202.179.249.50)  168.948 ms  168.712 ms  168.903 ms
 7  tpr5-ge0-0-0-4.jp.apan.net (203.181.248.250)  172.789 ms  170.367 ms  188.689 ms
 8  losa-tokyo-tp2.transpac2.net (192.203.116.145)  579.586 ms  284.736 ms  284.202 ms
 9  abilene-1-lo-jmb-702.lsanca.pacificwave.net (207.231.240.131)  303.736 ms  284.884 ms
530.854 ms
10  vl-101.xe-0-0-0.core0-gw.pdx.oregon-gigapop.net (198.32.165.65)  328.082 ms  305.800
ms  533.644 ms
11  vl-105.uonet9-gw.eug.oregon-gigapop.net (198.32.165.92)  336.680 ms  617.267 ms
495.685 ms
12  vl-3.uonet2-gw.uoregon.edu (128.223.3.2)  310.552 ms  421.638 ms  612.399 ms
13  nsrc.org (128.223.157.19)  309.548 ms  612.151 ms  611.505 ms
```

Ud. comprende lo que cada linea significa? Si no, vea la pagina de Wikipedia y ademas lea el manual en linea

```
$ man traceroute
```

para mas informacion. Que significa si ve lineas como estas?

```
15  * * *
16  * * *
17  * * *
```

Si ve lo anterior, significa que el dispositivo remoto no responde a peticion de eco ICMP, o esta configurado con direcciones privadas (vea RFC 1918.)

Como puede ver, traceroute puede ser usado para determinar en que lugar(es) hay problemas en la conexion entre dos puntos de la red.

Trate de ejecutar traceroute otra vez al mismo host (nsrc.org). Esta vez tomara' menos tiempo. Por que?

3. mtr

La herramienta mtr combina ping y traceroute en una sola pantalla que se actualiza dinamicamente. Antes de usarlo es posible que deba instalarlo:

```
$ sudo apt-get install mtr-tiny
```

Ahora pruebe:

```
$ mtr nsrc.org
```

La respuesta del programa se ve diferente en diferentes versiones de Linux y UNIX, pero en esencia vera' un resumen de la perdida de paquetes en cada paso en el camino al dispositivo encuestado, numero de paquetes de solicitud de eco ICMP enviado, el tiempo de retorno (RTT) mas reciente, y el mejor, peor, y average tiempo de retorno, asi como la desviacion estandar de estos.
Al mostrar el % de paquetes perdidos en este formato, es mucho mas facil detectar en que punto puede ser que tenga problemas su red (o conexion)

Ejercicios Parte II
==========

Analisis de Redes
---

1. lsof y netstat

Para verificar que servicios estan corriendo en su servidor, puede utilizar un grupo de herramientas con multitud de opciones.
Utilizando el manual del sistema (man), puede ver lo que los comandos proveen como

informacion. Note que hay muchas opciones y combinaciones!

Para visualizar las paginas de manual utilize "man lsof", "man netstat", o tambien "lsof -h" and "netstat -h". Recuerde ejecutar estos comandoos como root para tener los suficientes privilegios y permisos necesarios.
Puede ser que necesite instalar lsof antes de usarlo. Para ello:

```
$ sudo apt-get install lsof
```

* Using lsof, what IPv4 services are listening on your machine?
* Using netstat, what IPv4 and IPv6 services are listening on your machine?

When you run lsof and netstat you should run them as root:

```
$ sudo lsof
$ sudo netstat
```

2. tcpdump and tshark

Primero, instalemos estos paquetes:

```
$ sudo apt-get install tcpdump tshark
```

Use tcpdump asi:

```
$ sudo tcpdump -i lo -A -s1500 -w /tmp/tcpdump.log
```

Ahora genere algun trafico en su interface de loopback (lo) en otra terminal. Esto es, abra una sesion de ssh a su servidor virtual, o enviiele solicitudes de eco ICMP.

Por ejemplo:

```
$ ping localhost
$ ssh localhost
```

etc. Para terminar presione CTRL-C

Nota: ssh genera informacion mucho mas interesante. Leamos el fichero de salida resultante con tshark:

```
$ sudo tshark -r /tmp/tcpdump.log | less
```

Que vee? Puede seguir la sesion de ssh originada anteriormente paso a paso?

Ahora usaremos ftp. Primero instalemos un cliente de FTP:

```
$ sudo apt-get install ftp
```

Ahora trataremos esto:

```
$ sudo rm /tmp/tcpdump.log
$ sudo tcpdump -i eth0 -A -s1500 -w /tmp/tcpdump.log
```

En la otra terminal, ejecute:

```
$ ftp limestone.uoregon.edu

Connected to limestone.uoregon.edu.
220 FTP Server ready.
Name (limestone.uoregon.edu:sysadmin): anonymous
Password: <anything you want>
ftp> exit
```

Termine la session de tcpdump en la otra terminal (CTRL-C). Ahora vea el contenido del fichero de salida:

```
$ sudo tshark -r /tmp/tcpdump.log | less
```

Puede ver su password?

Si Ud. tiene una buena cantidad de trafico en su red, entonces la cantidad de informacion compilada en tcpdump.log puede ser bastante grande. Puede buscar patrones de su sesion de FTP de esta forma:

```
"/FTP"
```

en la pantalla de salida. Dado que ud. redirecciono' via una "tuberia" la salida del comando tshark al utilitario paginador less, el uso del buscador de patrones "/" del paginador funciona bien
Ahora presione "n" para la proxima ocurrencia del patron "FTP".
Debe ver una linea con la cadena:

```
"FTP Request: PASS PasswordYouTypedIn"
```

Detectar passwords no encriptados en una LAN inalambrica es facil con una herramienta como esta

Recuerde limpiar el fichero de coleccion:

```
$ rm /tmp/tcpdump.log
```

3. Usando iperf

Instale iperf:

```
$ sudo apt-get install iperf
```

Use "man iperf" o "iperf -h" como ayuda.

Pida a su colega de al lado ejecutar:

```
$ iperf -s
```

Conectese al servidor de su colega usando:

```
$ iperf -c ipNeighbor
```

Si no conoce la direccion IP del servidor de su colega, pidale que la determine de esta forma con el comado ifconfig:

```
$ ifconfig eth0
```

Cuanto ancho de banda hay entre sus dos servidores?
Ud. puede repetir este ejercicio con cualquier otro servidor donde iperf este' instalado y usted tenga una cuenta de usuario. Es una forma rapida y simple de determinar el ancho de banda entre dos puntos.

Para detener el servidor de iperf que Ud ejecuto' como "iperf -s" simplemete presione CTRL-c

Si tiene tiempo para seguir probando opciones de iperf: si tiene un servidor remoto fuera del taller donde pueda instalar iperf, puede determinar el ancho de banda entre estos dos puntos.

Otras cosas interesantes a probar:

* Pruebe TCP usando varios tallas de ventana (-2).

* Verifique TCP MSS (-m). Como afecta esto el rendimiento? Que cosa es "descubrimiento del MTU del camino" (Path MTU discovery)

* Pruebe con dos threads en paralelo (-P) y compare los totales. Hay alguna diferencia? Por que?

* Pruebe con diferentes tallas de paquetes y la opcion TCP_NODELAY (-N).

Opcionales/Pruebas Avanzadas

A) ping con talla de paquete variable

Por defecto, ping envia datagramas IP de talla 84 bytes:

* 20 bytes encabezamiento IP
* 8 bytes encabezamiento ICMP
* 56 bytes rellenos de datos

Sin embargo, Ud. puede enviar paquetes mas grandes usando la opcion -s. Usando "-s 1472" podra enviar un datagrama IP de 1500 bytes, que es el maximo en redes Ethernet (la suya). Por encima de 1500, el paquete es mas grande qe la talla maxima de paquete (MTU = Maximum Transmission
Unit). Este simple mecanismo permite diagnosticar muchos tipos de problemas, e incluso

distinguir entre demora de transmision y demora de propagacion (alguien recuerda la diferencia?)
Encontremos un servidor que este relativamente cerca de nosotros.
(Use traceroute ppara ver algun servidor cerca)

```
$ traceroute nsrc.org
```

Busque otro servidor que este al menos 2 o mas saltos de distancia (pues como su enrutador es virtual, la herramienta no funciona adecuadamente con servidores virtualizados)
El segundo salto es el enrutador que es el gateway de nuestro laboratorio al exterior. Este esta muy cerca para usarlo en la medicion, no dara' diferencia substanciales. Nos referiremos al servidor que selecciono' como PING_MACHINE.

Envia 20 pings standard a esa direccion:

```
$ ping -c20 PING_MACHINE
```

Anote el tiempo de retorno (RTT) *average* (t1).

Ahora envie 20 pings de talla maxima:

```
$ ping -c20 -s1472 PING_MACHINE
```

Otra vez, Anote el tiempo de retorno (RTT) *average* (t1).

La demora de propagacion en ambos casos es la misma (mismo camino), por tanto el tiempo de retorno mas grande debe ser debido a la demora de transmision.
Ahora puede estimar la demora de transmision y de ahi el ancho de banda entre dos puntos:

```
    aumento en tiempo de transmission  =  t2 - t1
    aumento en bits enviados           =  (1500-84) * 8 * 2  = 22656
```

(multiplique por 2 porque el tiempo de retorno equivale a eniar el pauqete dos veces, la ida y la vuelta)

Divida los bits por el tiempo para obtener un estimado de los bits por segundo, Recuerde convertir de segundos a milisegundos primero:

Ejemplo:

t2 = 1.71
t1 = 1.14

t2-t1 = 0.57

0.57 ms = 0.00057 sec

22656 bits / 0.00057 sec = 39747368.42 bps

Ahora puede convertir a Kbps, Mbps, etc.

Haciendo esto para diferentes saltos, es posible estimar el ancho de banda en cada tramo, aun cuando esten remotos.
Existe una herramienta que facilita esto: "pathchar" , pero debe de compilarla de codigo fuente

Algunos binarios para OS especificos aqui:

ftp://ftp.ee.lbl.gov/pathchar/

La pagina web con documentacion aqui:

http://www.caida.org/tools/utilities/others/pathchar/

B) tcpdump parte II

Puede utilizar tcpdump como una herramienta de analsis forense en tiempo-real. Tcpdump tiene muchas opciones, y tomaria mucho tiempo explicar todas las posibilidades. Utilicemos un caso practico:

Observemos una solicitud de DHCP de una PC y las respuestas recibidas

Primero conectese a su servidor virtual, como root:

```
$ sudo bash
```

Usemos el utilitario screen (multi-pantallas)

```
# apt-get install screen
```

Ejecute screen:

```
# screen
```

Ahora podemos tener mutiples terminales. Empecemos a colectar datos en ua pantalla:

```
# tcpdump -s0 -ni eth0 port 67 or port 68
```

Abramos otra pantalla con screen:

```
Press ctrl-a c
```

Lea el manual de tcpdump para saber que hacen las opciones "-s0", "-n" and "-i":

```
# man tcpdump
```

(Busque "-s" escribiendo "/" y entonces "-s" y presione ENTER. Presione "n" para ver la proxima ocurrencia de la cadena -s")

Ahora haga una solicitud de DHCP para una nueva direccion IP para eth0 en su servidor:

```
# dhclient
```

Retorne a la pantalla anterior para ver la informacion de salida de tcpdump (presione "ctrl-a p") (p = previa, n= proxima)

Debe ver algo como esto:

```
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
18:03:05.003190 IP 0.0.0.0.68 > 255.255.255.255.67: BOOTP/DHCP, Request from 52:54:4a:5e:
68:77, length 300
18:03:05.004349 IP 10.10.0.254.67 > 10.10.0.250.68: BOOTP/DHCP, Reply, length 300
```

Para detener la sesion de tcpdump presione "ctrl-c"

Usted sabe lo que esto significa? Por que especificamos a tcpdump que "escuchara" en los puertos 67 y 68? Si busca en el fichero /etc/services encontrara' las definiciones de ambos puertos como "bien conocidos":

```
bootps          67/udp     # Bootstrap Protocol Server
bootps          67/tcp     # Bootstrap Protocol Server
bootpc          68/udp     # Bootstrap Protocol Client
bootpc          68/tcp     # Bootstrap Protocol Client
```

Puede retornar a la pantalla donde corrio' dhclient
Y teclee:

ctrl-a-n

Y:

```
# exit
```

Si le interesa la herramienta "screen", vea:

http://www.howtoforge.com/linux_screen