

Trabajo Práctico

TP 1 - Herramientas de Diagnóstico de Redes

Fecha de entrega: 15/09/2021

Franco, Juan Martín 149.615

juanmartin_franco@hotmail.com

ping (RTT)

A pesar de su simpleza ping continúa siendo una herramienta realmente útil para obtener una estimación del tiempo de ida y vuelta (Round-Trip Time) entre dos hosts.

1. Destine cinco minutos a elegir siete destinos contra los cuales realizar una medición de RTT. Deberán estar ubicados: uno en su misma provincia, el segundo en otra provincia, y los restantes en cada uno de los cinco continentes. Verifique mediante algún método (ej.: geoip) que cada host remoto se encuentra en el lugar geográfico correspondiente. Configure ping para enviar Echo Request cada 0.5 segundos hasta alcanzar 600 mensajes. Refiérase al manual de la herramienta (man ping) para determinar qué parámetros permiten establecer tal configuración.

Host Destino	Dirección IP	Ubicación	Latencia Mínima	Latencia Promedio	Latencia Máxima	Desvío	% Pérdida
www.samsung.com	92.123.85.197	BS.AS.,ARG	16.44ms	20.69ms	27.07ms	4.119ms	0%
www.unc.edu.ar	200.16.16.170	CBA,ARG	25.30ms	29.33ms	31.46ms	2.435ms	0%
www.reddit.com	151.101.93.140	BRASIL	48.34ms	51.04ms	53.98ms	2.054ms	0%
www.amazon.co.uk	178.236.7.220	IRLANDA	223.05ms	224.59ms	226.48ms	1.469ms	0%
www.mega.nz	66.203.127.18	NZ	236.73ms	239.51ms	242.19ms	2.052ms	0%
www.amazonaws.com	52.68.160.2	JAPÓN	309.31ms	311.07ms	312.43ms	1.175ms	0%
www.gov.za	164.151.129.20	SUDÁFRICA	392.77ms	394.35ms	395.95ms	1.125ms	0%

Los parámetros utilizados fueron:

El parámetro -c para definir la cantidad de pings a realizar.

El parámetro -i para definir el intervalo entre cada ping.

2. ¿Qué observa a partir de las mediciones reflejadas en la tabla? ¿A qué podría deberse?

En las mediciones que se ven reflejadas en la tabla, se puede observar una amplia variación en la latencia. Esto, puede deberse a múltiples factores, entre los que se pueden destacar la distancia entre el host origen y el host destino.

Pero cabe destacar que este no es el único factor que hace variar la latencia ya que, por ejemplo, al realizar múltiples pings hacia www.mega.nz (ubicado en Nueva Zelanda, a aproximadamente 9.300 km de Buenos Aires) obtuve una latencia promedio de 239.51ms, y cuando realicé el mismo procedimiento hacia el host de www.gov.za (ubicado en Sudáfrica, a aproximadamente 7.700 km de distancia, mucho más cerca que Nueva Zelanda) obtuve una latencia promedio de 394.35ms, lo cual explica que no siempre a mayor distancia mayor latencia.

Otro factor que puede hacer variar la latencia puede ser: los saltos intermedios por los que tienen que pasar los paquetes.

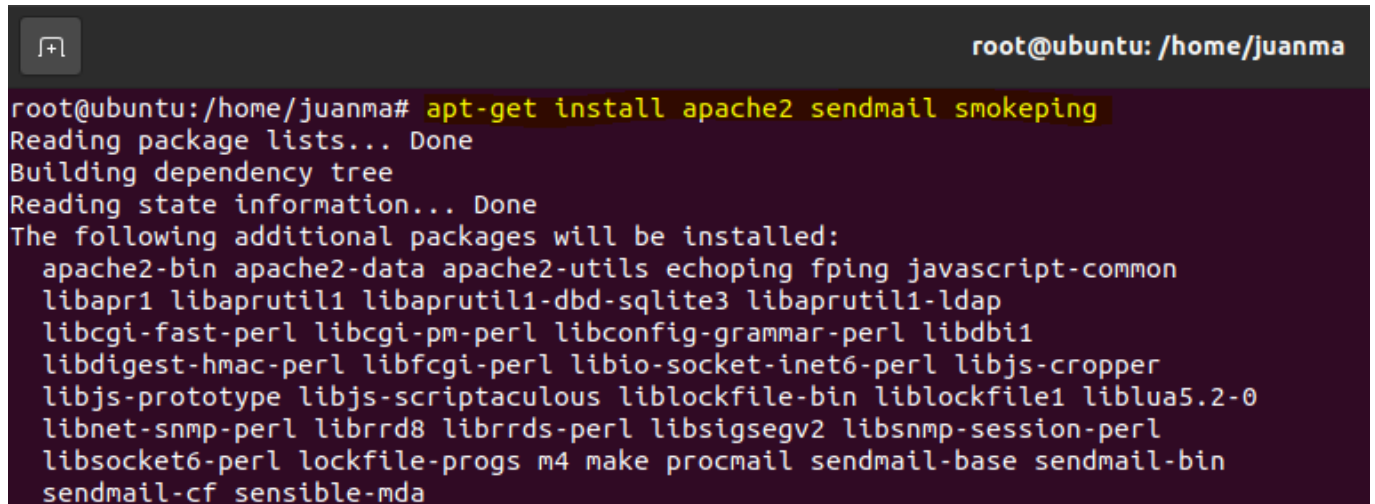
3. Instale y configure la herramienta SmokePing. Mida durante al menos dos horas contra 3 hosts localizados en distintos continentes (puede emplear aquellos de la consigna)

previa). Adjunte los gráficos correspondientes a las mediciones realizadas y comente los comportamientos que puede observar a partir de ellos.

Encontrará una breve guía de configuración de la herramienta adjunta a esta práctica.

Para realizar esta consigna, procederé a instalar la herramienta SmokePing mediante el uso del comando:

```
apt-get install apache2 sendmail smokeping
```



```
root@ubuntu: /home/juanma
root@ubuntu:/home/juanma# apt-get install apache2 sendmail smokeping
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils echoping fping javascript-common
  libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap
  libcgi-fast-perl libcgi-pm-perl libconfig-grammar-perl libdbi1
  libdigest-hmac-perl libfcgi-perl libio-socket-inet6-perl libjs-cropper
  libjs-prototype libjs-scriptaculous liblockfile-bin liblockfile1 liblua5.2-0
  libnet-snmp-perl librrd8 librrds-perl libsigsegv2 libsnmp-session-perl
  libsocket6-perl lockfile-progs m4 make procmail sendmail-base sendmail-bin
  sendmail-cf sensible-mda
```

Una vez ejecutado dicho comando, se instalará la herramienta.

Ahora, siguiendo el manual de configuración de SmokePing provisto por los docentes, ejecuto el siguiente comando:

```
nano /etc/smokeping/config.d/General
```

Y reemplazo el contenido de dicho archivo con lo siguiente:



```
GNU nano 4.8 /etc/smokeping/config.d/General
*** General ***

owner = Administración y Gestión de Redes
contact = aygr@unlu.edu.ar
mailhost = localhost
cgiurl = http://localhost/cgi-bin/smokeping.cgi
syslogfacility = local0

@include /etc/smokeping/config.d/pathnames
```

Una vez guardados los cambios, procedo a realizar lo mismo en el archivo Targets, utilizando el siguiente comando:

```
nano /etc/smokeping/config.d/Targets
```

Y reemplazo el contenido del archivo con lo siguiente:

```
root@ubuntu: /home/juanma
GNU nano 4.8 /etc/smokeping/config.d/Targets Modified
*** Targets ***

probe = FPing
menu = Top
title = Monitor de Latencia de la Red
remark = Bienvenido a Smokeping. \
        Aquí encontrará información acerca de la Latencia de la Red

+ WAN #

menu = Conexión a Internet
title = Monitorización Enlace WAN

++ Google

menu = Google.com
title = Google
host = www.google.com
```

Por último, reinicio los servicios de SmokePing y Apache2 mediante los siguientes comandos:

```
service smokeping restart
service apache2 reload
```

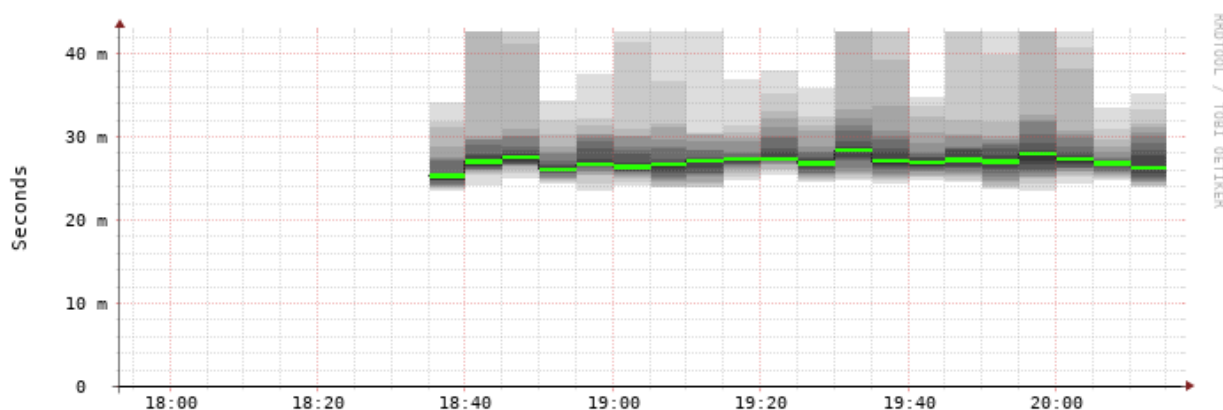
Universidad Nacional de Cordoba

Navigator Graph

Time range: 2021-09-09 17:53

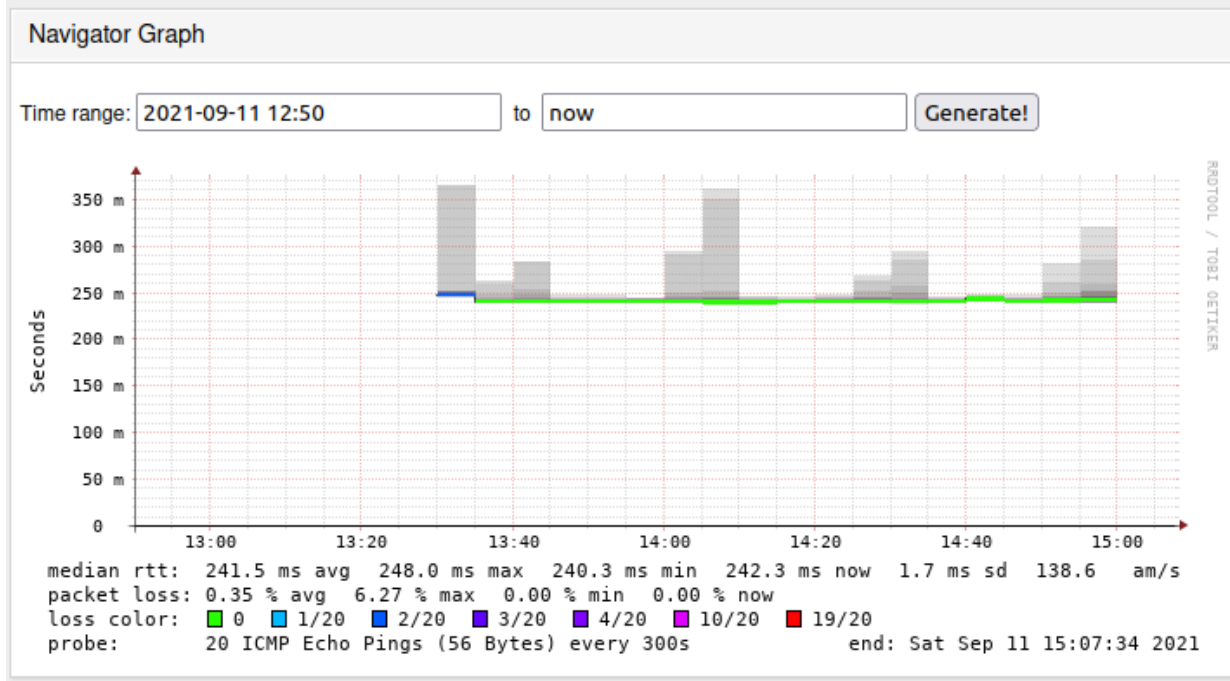
to now

Generate!



median rtt: 27.0 ms avg 28.4 ms max 25.3 ms min 26.3 ms now 0.6 ms sd 41.5 am/s
packet loss: 0.00 % avg 0.00 % max 0.00 % min 0.00 % now
loss color: 0 1/20 2/20 3/20 4/20 10/20 19/20
probe: 20 ICMP Echo Pings (56 Bytes) every 300s end: Thu Sep 9 20:16:56 2021

Mega



4. ¿Qué comportamiento se observa? ¿Qué implicaría un incremento/disminución de la latencia a partir de un patrón establecido? ¿Qué otras utilidades ofrece esta herramienta?

Para analizar el comportamiento primero me parece óptimo dejar en claro que representa cada cosa en el gráfico.

La línea de color verde (en caso de no haber pérdida de paquetes) representa a la media de las latencias y, si hay pérdida de paquetes, su color variará en base a la cantidad de paquetes perdidos.

El “humo” representa el resto de los valores de latencia, o sea, marca la variabilidad de la media.

En ambos casos, el desvío de la media no fue muy grande (0.6 en la UNC y 1.7ms en Mega.nz).

Si bien parece que el gráfico de Mega.nz es menos variable esto se debe a que, como la latencia del servidor de Mega maneja valores superiores, el gráfico adapta la escala en valores de a 50ms.

Las variaciones se pueden apreciar mejor en el gráfico de la UNC ya que la escala utiliza valores de a 10ms.

Un incremento/disminución de la latencia a partir de un patrón establecido implicaría que la red está muy congestionada/poco congestionada.

Las otras utilidades que ofrece esta herramienta se basan en medir el retardo de los siguientes servicios:

- DNS,
- HTTP y HTTPS,
- Whois,
- SMTP,
- FPing6 (equivalente al realizado en este trabajo práctico pero con ICMPv6),

entre otros.

5. ¿De qué manera afecta la latencia a las aplicaciones? Describa y brinde ejemplos.

La latencia afecta de manera considerable la experiencia de uso de aplicaciones que requieren el envío y la recepción de paquetes.

Por poner un ejemplo, si estoy jugando un juego online (cualquiera sea el juego, suponiendo que no es un juego de estrategia donde el tiempo de respuesta no suele ser tan vital) es indispensable que la latencia

sea lo más baja posible, ya que mientras más latencia haya (o sea, mientras más tarde un paquete de datos en completar un recorrido de ida y vuelta), más tardará en llegar a mi pantalla lo que realmente está pasando en la partida online y esto genera que otros jugadores con menor latencia puedan aprovecharse de dicha situación.

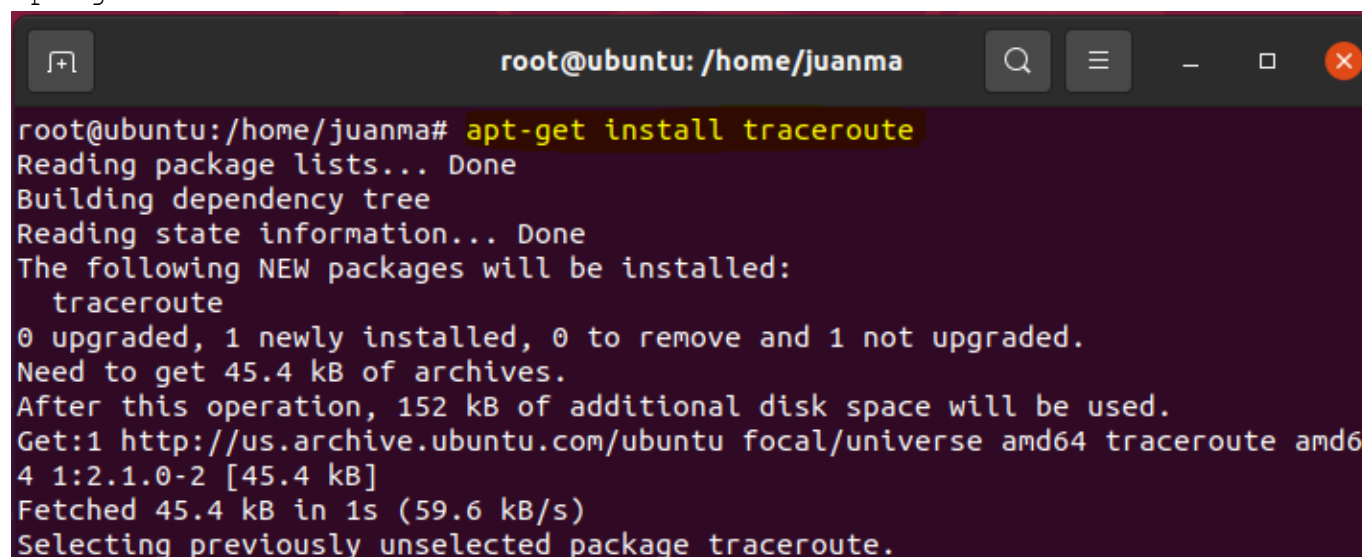
traceroute

El comando **traceroute** aprovecha ciertas particularidades del protocolo IP para intentar obtener y mostrar en pantalla el camino que toma un paquete al ir de un host a otro. Además, si recibe respuesta, muestra el RTT percibido hasta cada uno de los dispositivos que forman el camino.

1. Instale la herramienta y explique cómo funciona. ¿Qué significan las líneas *.*.* en la salida del programa?

Para instalar la herramienta, procedo a utilizar el siguiente comando dentro de la terminal:

```
apt-get install traceroute
```

A screenshot of a terminal window with a dark background. The title bar shows 'root@ubuntu: /home/juanma'. The terminal text shows the command 'apt-get install traceroute' being executed. The output indicates that the package is being installed, showing details like package lists, dependency tree, and disk space requirements. It also shows the source of the package and the selection of the package.

```
root@ubuntu:/home/juanma# apt-get install traceroute
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  traceroute
0 upgraded, 1 newly installed, 0 to remove and 1 not upgraded.
Need to get 45.4 kB of archives.
After this operation, 152 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 traceroute amd64 1:2.1.0-2 [45.4 kB]
Fetched 45.4 kB in 1s (59.6 kB/s)
Selecting previously unselected package traceroute.
```

El funcionamiento de traceroute es bastante simple:

Traceroute envía paquetes al destino con el campo TTL (Time-To-Live) igual al número de saltos. Cada router disminuye dicho valor de un paquete entrante y si visualiza un paquete entrante con TTL = 0 lo descarta, de lo contrario disminuye el valor y lo envía a otro router. Al mismo tiempo, se envía información al origen sobre la identidad del router, dicha información es de diagnóstico. Este proceso se repite hasta que se alcanza el destino o hasta que se alcanza el TTL máximo permitido. Si el router no responde dentro de un tiempo de espera, traceroute imprime un *. Como generalmente se envían 3 paquetes a cada máquina, si no obtengo respuesta de ningún paquete se visualizará *.*.*. Con esto puedo concluir que * equivale a no obtener respuesta.

2. Verifique la ruta que podría llegar a seguir un paquete IP hacia el host 8.8.8.8 (servidor DNS público de Google). Ejecute la misma consulta varias veces y en momentos distintos ¿Qué conclusión puede obtener?

Para realizar esto, utilizo el comando:

```
traceroute -I 8.8.8.8
```

Obteniendo los siguientes resultados:

(traceroute ejecutado el 4/9 a las 11:53 am)

```
juanma@ubuntu:~$ traceroute -I 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1  _gateway (192.168.66.2)  0.136 ms  0.076 ms  0.124 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  * * *
 8  * * *
 9  * * *
10  * * *
11  dns.google (8.8.8.8)  26.288 ms  26.858 ms  20.472 ms
```

(traceroute ejecutado el 6/9 a las 17:28)

```
juanma@ubuntu:~$ traceroute -I 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1  _gateway (192.168.66.2)  0.122 ms  0.052 ms  0.079 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  * * *
 8  * * *
 9  * * *
10  * * *
11  dns.google (8.8.8.8)  24.167 ms  24.080 ms  18.715 ms
```

La conclusión que puedo sacar es que la latencia entre que se envía y vuelve un paquete de datos es un valor que varía en funciones de varios factores entre los que se pueden encontrar la distancia entre ambos hosts, lo congestionada que esté la red, etc.

En este caso, se puede observar que el traceroute realizado el 6/9 me indica una menor latencia, por lo que puedo suponer que el camino que tomaron los 3 paquetes estaba menos congestionado que el camino que tomaron los 3 paquetes del traceroute realizado el 4/9.

3. En qué situaciones puede llegar a ser útil esta herramienta. Ejemplifique.

El comando traceroute puede llegar a ser muy útil para averiguar si hay algún problema en el camino hacia un equipo de mi propia red o de una red externa.

Por ejemplo: Si quiero entrar al sitio web www.google.com y no me carga correctamente dicho sitio, aparte de realizar un ping hacia el mismo, puedo ejecutar el comando traceroute para averiguar si la comunicación hacia el host de destino se pierde o se interrumpe en algún nodo del camino. Con estos resultados, puedo averiguar donde ocurre el problema, o solucionarlo en caso de que el problema sea a nivel de red local.

Este comando puede tener utilidad también en una red local muy grande, en donde un paquete puede llegar a tomar múltiples caminos para llegar al host de destino y con esta herramienta puedo saber por cuales nodos está pasando el paquete que se envía.

4. Una vez que finaliza el descubrimiento de la ruta con traceroute, ¿Se puede afirmar que todos los paquetes IP (de la prueba que ejecutó esa instancia del programa) siguieron exactamente esa misma ruta? Justifique su respuesta.

No, ya que los 3 paquetes que recibieron respuesta (dentro de la fila 11) poseen diferente latencia, lo que hace pensar que los paquetes fueron por diferentes rutas (y eso generó que varíe la latencia de los mismos).

Por ejemplo, el paquete 1 (es decir, el primero de la izquierda) tuvo una latencia de 26.288ms, mientras que el paquete 2 (el del medio) tuvo una latencia de 26.858ms, mientras que el paquete 3 (el último, el de la derecha) tuvo una latencia de 20.472, lo que me permite deducir que puede haber tomado un camino más corto o simplemente el camino menos congestionado.

5. Realice traceroute a los hosts definidos en el ejercicio de ping anterior. Adicione a la tabla previa una columna con la cantidad de dispositivos intermedios.

Para realizar este ejercicio, utilizaré el comando

```
traceroute -I [host destino]
```

El cual permite utilizar paquetes ICMP para los probes (en lugar de utilizar UDP, lo que utiliza Linux por defecto).

Host Destino	Dirección IP	Ubicación	Latencia Mínima	Latencia Promedio	Latencia Máxima	Cantidad de Dispositivos Intermedios
www.samsung.com	92.123.85.197	BS.AS.,ARG	16.44ms	20.69ms	27.07ms	11
www.unc.edu.ar	200.16.16.170	CBA,ARG	25.30ms	29.33ms	31.46ms	14
www.reddit.com	151.101.93.140	BRASIL	48.34ms	51.04ms	53.98ms	12
www.amazon.co.uk	178.236.7.220	IRLANDA	223.05ms	224.59ms	226.48ms	16
www.mega.nz	66.203.127.18	NZ	236.73ms	239.51ms	242.19ms	20
www.amazonaws.com	52.68.160.2	JAPÓN	309.31ms	311.07ms	312.43ms	13
www.gov.za	164.151.129.20	SUDÁFRICA	392.77ms	394.35ms	395.95ms	16

6. En una red externa a la Universidad, realice traceroute al sitio web www.unlu.edu.ar y otro a www.ut.ee . Indique el ISP que provee el servicio de conectividad a Internet en ese momento. Adjunte la salida del traceroute.

Para realizar este ejercicio, utilicé el siguiente comando:

```
traceroute -I www.unlu.edu.ar
```

Obteniendo el siguiente resultado:


```

juanma@ubuntu:~$ traceroute -I www.unlu.edu.ar
traceroute to www.unlu.edu.ar (190.104.80.1), 30 hops max, 60 byte packets
 1  _gateway (192.168.66.2)  0.613 ms  0.544 ms  0.475 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  * * *
 8  * * *
 9  * * *
10  * * *
11  * * *
12  unlu1.unlu.edu.ar (190.104.80.1)  19.136 ms  19.001 ms  22.854 ms

```

Luego, utilicé el comando:

```
traceroute -I www.ut.ee
```

Obteniendo el siguiente resultado:

```

juanma@ubuntu:~$ traceroute -I www.ut.ee
traceroute to www.ut.ee (193.40.5.73), 30 hops max, 60 byte packets
 1  _gateway (192.168.66.2)  1.788 ms  1.615 ms  1.444 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  * * *
 8  * * *
 9  * * *
10  * * *
11  * * *
12  * * *
13  * * *
14  * * *
15  * * *
16  * * *
17  * * *
18  * * *
19  www.ut.ee (193.40.5.73)  282.693 ms  289.083 ms  288.979 ms

```

ISP en el momento de la ejecución de los comandos: Fibertel.

nmap (exploración de la Red)

Herramienta para escaneo de puertos y exploración de redes. Las referencias básicas son el manual (man nmap) y el sitio oficial <http://nmap.org>.

1. Instale nmap en su equipo. Ejecute un escaneo básico contra el equipo indicado por el docente y a su propio equipo.

```
$ nmap -Pn 190.104.80.3
```

```
$ nmap localhost
```

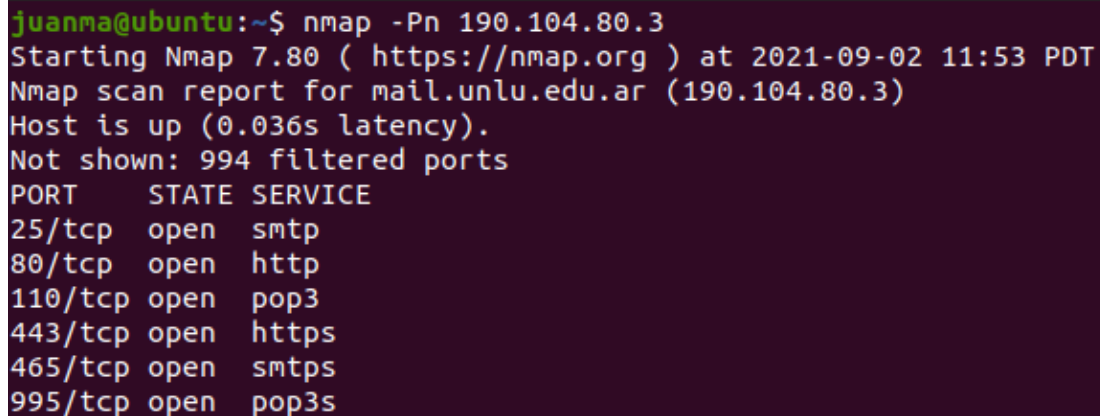

¿Qué información brinda la salida del comando? ¿Qué rol tiene ese host en la organización? ¿En qué medida esta información puede ser útil o peligrosa para una organización?

Para resolver el enunciado, primero procedo a instalar la herramienta nmap mediante la ejecución del siguiente comando:

```
sudo apt install nmap
```

Una vez hecho esto, procedo a ejecutar los comandos que están en el enunciado:

```
nmap -Pn 190.104.80.3
```



```
juanma@ubuntu:~$ nmap -Pn 190.104.80.3
Starting Nmap 7.80 ( https://nmap.org ) at 2021-09-02 11:53 PDT
Nmap scan report for mail.unlu.edu.ar (190.104.80.3)
Host is up (0.036s latency).
Not shown: 994 filtered ports
PORT      STATE SERVICE
25/tcp    open  smtp
80/tcp    open  http
110/tcp   open  pop3
443/tcp   open  https
465/tcp   open  smtps
995/tcp   open  pop3s
```

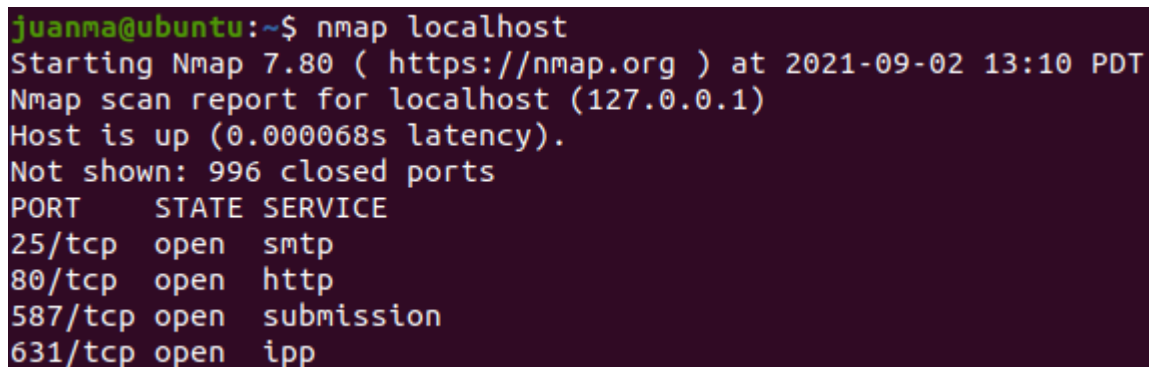
La herramienta nmap escanea la dirección provista o el bloque de direcciones provisto (en este caso, solo la dirección 190.104.80.3) y, mediante el parámetro -Pn le estoy indicando a la herramienta que no realice ping.

Mediante los datos provistos por la herramienta se puede ver claramente que la ip 190.104.80.3 pertenece al servidor de correo electrónico de la UNLu (por ejemplo, se puede visualizar que dicha ip está asociada a mail.unlu.edu.ar y que los entre los puertos abiertos se encuentran SMTP, POP3, etc. relativos al correo electrónico).

Esta información puede ser útil para un atacante ya que, mediante el escaneo de la red puede recrear la topología de la organización y conocer más a detalle cómo proceder para realizar los ataques, y además mediante el escaneo de cada host puede ver que puerto abierto tiene cada host, lo que permite verificar vulnerabilidades que pueden ser futuros puntos de entrada para ataques maliciosos.

Ahora, procedo a realizar el escaneo a mi propio host, utilizando el comando:

```
nmap localhost
```



```
juanma@ubuntu:~$ nmap localhost
Starting Nmap 7.80 ( https://nmap.org ) at 2021-09-02 13:10 PDT
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000068s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE
25/tcp    open  smtp
80/tcp    open  http
587/tcp   open  submission
631/tcp   open  ipp
```

En dicho escaneo se pueden ver abiertos los puertos 25 (con su servicio SMTP, el protocolo utilizado para la transferencia de correo electrónico), 80 (con su servicio HTTP, el protocolo utilizado para la transferencia de información en la web), 587 (con su servicio submission, es utilizado para enviar los

emails de forma segura y garantizar que lleguen a su destino) y 631 (con su servicio ipp, utilizado para permitir la impresión remota desde una PC a cualquier impresora accesible).

2. Lea el manual de la herramienta y ejecute el Ejemplo 1 del mismo contra localhost. Comente que información adicional visualiza respecto al ejercicio anterior. Compárelo con la ejecución del ejemplo 1 al dominio de la UNLu, y comente brevemente por qué una mala configuración puede representar un riesgo de seguridad.

En este caso, utilizando el comando

```
nmap -Pn localhost
```

Obtuve el mismo resultado que la ejecución del ejemplo 1:

```
juanma@ubuntu:~$ nmap -Pn localhost
Starting Nmap 7.80 ( https://nmap.org ) at 2021-09-11 13:27 PDT
Nmap scan report for localhost (127.0.0.1)
Host is up (0.00010s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE
25/tcp    open  smtp
80/tcp    open  http
587/tcp   open  submission
631/tcp   open  ipp

Nmap done: 1 IP address (1 host up) scanned in 0.06 seconds
```

El parámetro -Pn me permite omitir el ping que realiza previo al escaneo.

Esto puede resultar útil ya que a veces, algún host puede tener desactivado el protocolo ICMP y, al no recibir respuesta, interpreta que el host está inactivo.

Además de esto, el uso de este comando reduce el “ruido” que se realiza con el escaneo.

3. Una de las ventajas de nmap es que permite, mediante comodines o con formato CIDR, hacer un escaneo completo de un segmento de red para descubrir dispositivos presentes en la misma. Busque en el manual la sección “TARGET SPECIFICATION” (o bien en español: ESPECIFICACIÓN DE OBJETIVOS) y deduzca como puede encontrar todos los dispositivos conectados a su red. Puede ver su dirección IP actual mediante el comando ip addr show.

Para realizar un escaneo completo de mi red, primero tengo que saber cuál es la dirección de la misma.

Para esto, utilizo el comando:

```
ip addr show (o ip a s)
```

```
juanma@ubuntu:~$ ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:cb:57:b1 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.66.128/24 brd 192.168.66.255 scope global dynamic noprefixroute ens33
        valid_lft 1473sec preferred_lft 1473sec
    inet6 fe80::6a00:228e:f04e:d6e5/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

Una vez obtengo mi dirección de host y la máscara, puedo deducir que la dirección de mi red es:

192.168.66.0

ya que máscara 24 equivale al uso de 24 bits para definir la red y los restantes, en este caso 8, para los hosts.

Una vez sé que mi dirección de red es 192.168.66.0, utilizo el siguiente comando:

`nmap 192.168.66.0/24` (o sea, en formato CIDR).

```
juanma@ubuntu:~$ nmap 192.168.66.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2021-09-02 13:44 PDT
Nmap scan report for _gateway (192.168.66.2)
Host is up (0.00071s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE
53/tcp    open  domain

Nmap scan report for ubuntu (192.168.66.128)
Host is up (0.00033s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE
80/tcp    open  http

Nmap done: 256 IP addresses (2 hosts up) scanned in 2.90 seconds
```

En dicho escaneo, puedo ver que hay 2 hosts dentro de la red.

4. Investigue que opción permite hacer escaneo de puertos UDP y luego utilícela contra un host particular. ¿Por qué podría resultar útil realizar un análisis de esta característica, si la mayoría de los servicios de red utilizan TCP?

La opción que me permite realizar un escaneo de puertos UDP es el parámetro `-sU`

```
SCAN TECHNIQUES:
-sS/sT/sA/sW/sM: TCP SYN/Connect()/ACK/Window/Maimon scans
-sU: UDP Scan
-sN/sF/sX: TCP Null, FIN, and Xmas scans
--scanflags <flags>: Customize TCP scan flags
-sI <zombie host[:probeport]>: Idle scan
-sY/sZ: SCTP INIT/COOKIE-ECHO scans
-sO: IP protocol scan
-b <FTP relay host>: FTP bounce scan
```

`nmap 192.168.66.128 -sU`

Utilizando dicho parámetro obtuve el siguiente resultado:

```
root@ubuntu:/home/juanma# nmap 192.168.66.128 -sU
Starting Nmap 7.80 ( https://nmap.org ) at 2021-09-09 17:41 PDT
Nmap scan report for ubuntu (192.168.66.128)
Host is up (0.000021s latency).
Not shown: 998 closed ports
PORT      STATE      SERVICE
631/udp   open|filtered ipp
5353/udp  open|filtered zeroconf

Nmap done: 1 IP address (1 host up) scanned in 1.35 seconds
root@ubuntu:/home/juanma#
```

Este tipo de escaneos resulta de gran utilidad ya que, si bien es cierto que la gran mayoría de los servicios de red utilizan el protocolo TCP, existen servicios no menos importantes que operan sobre UDP como por ejemplo DNS y DHCP, los cuales podrían presentar vulnerabilidades que pueden llegar a ser aprovechadas por un posible atacante.

5. Instale en un equipo de su hogar la aplicación nmap y realice un escaneo a toda la red de su hogar. ¿Qué ha logrado descubrir? ¿Existen otros host aparte de su equipo? ¿Qué puertos poseen en escucha? ¿se corresponden con los servicios que usted esperaba?

Para realizar el escaneo primero verifico cual es la dirección ip de mi red, mediante el comando:

`ip a s`

```
juanma@ubuntu:~$ ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:cb:57:b1 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.66.128/24 brd 192.168.66.255 scope global dynamic noprefixroute ens33
        valid_lft 1159sec preferred_lft 1159sec
    inet6 fe80::6a00:228e:f04e:d6e5/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

Con la salida de este comando, puedo visualizar que la dirección ip de mi máquina es 192.168.66.128 y que la máscara de subred /24 me indica que 24 bits son para la red y los restantes para el host.

Con estos datos, es fácil saber que la dirección de mi red es 192.168.66.0 y, por lo tanto, puedo utilizar el siguiente comando en nmap:

`nmap 192.168.66.0/24`

Obteniendo el siguiente resultado:

```
juanma@ubuntu:~$ nmap 192.168.66.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2021-09-09 17:50 PDT
Nmap scan report for _gateway (192.168.66.2)
Host is up (0.00028s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE
53/tcp    open  domain

Nmap scan report for ubuntu (192.168.66.128)
Host is up (0.00029s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE
80/tcp    open  http

Nmap done: 256 IP addresses (2 hosts up) scanned in 2.91 seconds
```

En la última imagen se puede apreciar que hay 2 hosts activos en mi red (mi router y mi notebook, desde la cual estoy ejecutando los comandos).

Mi router tiene el puerto 53 abierto, el cual es utilizado por el servicio de DNS.

Mi notebook tiene el puerto 80 abierto, el cual es utilizado por el servicio HTTP.

iperf (throughput)

Constituye una herramienta que permite medir el throughput y la calidad de un enlace. Para ello, emplea un esquema cliente-servidor.

Este punto lo puede resolver utilizando máquinas virtuales o bien si así lo dispone utilizando dos equipos físicos. También puede consultar el listado de servidores iperf públicos.

Para comenzar con la resolución del enunciado, procedo a instalar la herramienta iperf, mediante el uso del comando:

```
apt-get install iperf
```

```
root@ubuntu:/home/juanma# apt-get install iperf
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  iperf
0 upgraded, 1 newly installed, 0 to remove and 4 not upgraded.
Need to get 76.5 kB of archives.
After this operation, 213 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 iperf amd64 2.0.13+dfsg1-1build1 [76.5 kB]
Fetched 76.5 kB in 1s (81.9 kB/s)
Selecting previously unselected package iperf.
(Reading database ... 186747 files and directories currently installed.)
Preparing to unpack .../iperf_2.0.13+dfsg1-1build1_amd64.deb ...
Unpacking iperf (2.0.13+dfsg1-1build1) ...
Setting up iperf (2.0.13+dfsg1-1build1) ...
Processing triggers for man-db (2.9.1-1) ...
```

1. Iniciar el servidor correspondiente para que reciba peticiones en un puerto diferente al definido por defecto. Verifique si la operación fue exitosa empleando el comando netstat (paquete net-tools) o bien mediante el comando ss -tnlp . Adjuntar salida del comando.

Para iniciar el servidor correspondiente y que este reciba peticiones en un puerto diferente al definido por defecto (por defecto es el 5001), tengo que utilizar el comando:

```
iperf -s -p [puerto]
```

Por ejemplo:

```
iperf -s -p 6000
```

El parámetro -p es para definir el puerto donde se van a recibir peticiones, y el parámetro -s es utilizado para que se ejecute en modo servidor.

```
root@ubuntu:/home/juanma# iperf -s -p 6000
-----
Server listening on TCP port 6000
TCP window size: 128 KByte (default)
-----
```

Ahora, verifico que se esté escuchando en ese puerto mediante el uso del comando

```
ss -ltnp
```



```

juanma@ubuntu:~$ ss -ltnp
State      Recv-Q    Send-Q    Local Address:Port    Peer Address:Port    Process
LISTEN     0         10        127.0.0.1:587         0.0.0.0:*
LISTEN     0        4096      0.0.0.0:6000         0.0.0.0:*
LISTEN     0        4096      127.0.0.53%lo:53     0.0.0.0:*
LISTEN     0         5        127.0.0.1:631        0.0.0.0:*
LISTEN     0        10       127.0.0.1:25         0.0.0.0:*
LISTEN     0        511      *:80                 *:.*
LISTEN     0         5        [::1]:631           [::]:.*

```

Efectivamente, se está escuchando en el puerto definido previamente.

2. Verificar el throughput existente con otro equipo perteneciente a la red del laboratorio bajo los protocolos TCP y UDP durante 60 segundos en intervalos de 5 segundos.

Para realizar este procedimiento, ejecutaré el siguiente comando:

```
iperf-2.0.13-win.exe -c 192.168.66.128 -p 6000 -i 5 -t 60
```

El parámetro -c [ip servidor] me permite ejecutar en modo cliente.

El parámetro -p [puerto] me permite especificar el puerto en el cual la aplicación está escuchando.

El parámetro -i [segundos] me permite especificar el intervalo (en segundos) entre los reportes.

El parámetro -t [segundos] me permite especificar el tiempo (en segundos) que se ejecutará el análisis.

```

C:\Users\juanm\Downloads>iperf-2.0.13-win.exe -c 192.168.66.128 -p 6000 -i 5 -t 60
-----
Client connecting to 192.168.66.128, TCP port 6000
TCP window size: 1.00 MByte (default)
-----
[328] local 192.168.66.1 port 55068 connected with 192.168.66.128 port 6000
[ ID] Interval      Transfer    Bandwidth
[328]  0.0- 5.0 sec  1.30 GBytes  2.23 Gbits/sec
[328]  5.0-10.0 sec  1.38 GBytes  2.37 Gbits/sec
[328] 10.0-15.0 sec  1.53 GBytes  2.62 Gbits/sec
[328] 15.0-20.0 sec  1.46 GBytes  2.50 Gbits/sec
[328] 20.0-25.0 sec  1.58 GBytes  2.71 Gbits/sec
[328] 25.0-30.0 sec  1.56 GBytes  2.68 Gbits/sec
[328] 30.0-35.0 sec  1.57 GBytes  2.70 Gbits/sec
[328] 35.0-40.0 sec  1.59 GBytes  2.73 Gbits/sec
[328] 40.0-45.0 sec  1.60 GBytes  2.75 Gbits/sec
[328] 45.0-50.0 sec  1.61 GBytes  2.76 Gbits/sec
[328] 50.0-55.0 sec  1.47 GBytes  2.52 Gbits/sec
[328] 55.0-60.0 sec  1.57 GBytes  2.70 Gbits/sec
[328]  0.0-60.0 sec 18.2 GBytes  2.61 Gbits/sec

```

Ahora, para realizar el mismo procedimiento pero bajo el protocolo UDP, utilizo el comando:

```
iperf-2.0.13-win.exe -c 192.168.66.128 -p 6000 -i 5 -t 60 -u
```

cuyos parámetros son iguales al comando anterior, salvo la diferencia en que a este último se agrega el parámetro -u el cual me permite especificar que se utilizará el protocolo UDP.

```

C:\Users\juanm\Downloads>iperf-2.0.13-win.exe -c 192.168.66.128 -p 6000 -i 5 -t 60 -u
-----
Client connecting to 192.168.66.128, UDP port 6000
Sending 1470 byte datagrams, IPG target: 11215.21 us (kalman adjust)
UDP buffer size: 64.0 KByte (default)
-----
[328] local 192.168.66.1 port 51431 connected with 192.168.66.128 port 6000
[ ID] Interval      Transfer    Bandwidth
[328]  0.0- 5.0 sec   642 KBytes  1.05 Mbits/sec
[328]  5.0-10.0 sec   640 KBytes  1.05 Mbits/sec
[328] 10.0-15.0 sec   640 KBytes  1.05 Mbits/sec
[328] 15.0-20.0 sec   640 KBytes  1.05 Mbits/sec
[328] 20.0-25.0 sec   640 KBytes  1.05 Mbits/sec
[328] 25.0-30.0 sec   640 KBytes  1.05 Mbits/sec
[328] 30.0-35.0 sec   639 KBytes  1.05 Mbits/sec
[328] 35.0-40.0 sec   640 KBytes  1.05 Mbits/sec
[328] 40.0-45.0 sec   640 KBytes  1.05 Mbits/sec
[328] 45.0-50.0 sec   640 KBytes  1.05 Mbits/sec
[328] 50.0-55.0 sec   640 KBytes  1.05 Mbits/sec
[328] 55.0-60.0 sec   639 KBytes  1.05 Mbits/sec
[328] WARNING: did not receive ack of last datagram after 10 tries.
[328]  0.0-60.0 sec  7.50 MBytes  1.05 Mbits/sec
[328] Sent 5351 datagrams

```

3. En el caso de TCP: Realice mediciones empleando diversos tamaños de ventana. Considerando valores: 1kb, 2kb, 16kb, 128kb, 320kb, 10mb Confeccione una gráfica que represente el throughput respecto del tamaño de ventana efectivamente asignado por el programa.

Tamaño de la Ventana	Throughput (GB/s)
1kb	0,383
2kb	0,395
16kb	0,407
128kb	0,406
320kb	0,418
10mb	0,401

4. ¿Qué permite establecer la opción -M ? ¿Cómo afecta esto al throughput? Investigue la técnica “Path MTU discovery”.

La opción -M me permite establecer el tamaño máximo del segmento.

Básicamente esto me indica el tamaño máximo (medido en bytes) que un dispositivo puede recibir en un determinado momento sin fragmentar.

Esta opción afecta al throughput ya que:

Cuando se trata de rendimiento, es importante elegir un tamaño óptimo para la ventana de recepción. Elegir un tamaño demasiado chico, limitará al remitente.

Elegir un tamaño demasiado grande, puede provocar retransmisiones ineficaces de paquetes en caso de pérdida de paquetes.

Para encontrar el tamaño óptimo debo calcular el BDP (o producto retardo x ancho de banda):

$BDP = \text{Ancho de banda} * RTT$.

Calculando el BDP nos puede dar un número superior a 65.535bytes ($n > 65.535\text{bytes}$).

Esto nos dice que seríamos capaces de recibir n bytes (valor del BDP) si queremos utilizar la capacidad de red al máximo posible, pero esto no es posible en la ventana de recepción, ya que excede el tamaño máximo de ventana de TCP (65.535 bytes).

Para resolver esto se utiliza la opción TCP Windows scale (la cual está definida en el RFC 1323).

El valor de dicha opción especifica cuantas veces el tamaño de la ventana de recepción necesita desplazarse a la izquierda para obtener el tamaño de ventana utilizable.

Resumiendo, cuando el Throughput no es el esperado, al buscar posibles cuellos de botella debo investigar la configuración de TCP utilizada.

La optimización del tamaño de ventana de recepción, la habilitación de la opción Windows Scale y el uso de SACK (Selective ACK) pueden mejorar el rendimiento.

El PMTU Discovery es una de las únicas formas en las que TCP intenta adaptar explícitamente el tamaño de su segmento después de que se haya iniciado una conexión, al menos cuando haya grandes cantidades de datos a transferir. El tamaño máximo de un segmento puede afectar el rendimiento general, al igual que el tamaño de la ventana. [Stevens - TCP IP Illustrated].

Explicación de Path MTU Discovery:

Cuando un host necesita transmitir datos a través de una interfaz, hace referencia a la MTU de la interfaz para saber la cantidad de datos que puede transmitir en c/paquete.

Por ejemplo, las interfaces Ethernet tienen un MTU predeterminado de 1500bytes.

Pero no todos los enlaces de internet tienen la misma MTU, ya que este puede variar según el tipo de medio físico, por ejemplo.

Cuando un router decide reenviar un paquete a una interfaz pero comprueba que el tamaño del paquete excede la MTU de la interfaz, debe fragmentar el paquete para transmitirlo como 2 (o más) piezas individuales.

La fragmentación es costosa (hablando tanto de recursos del router como de utilización del ancho de banda).

Para utilizar una ruta de la manera más eficiente, los hosts deben encontrar la ruta MTU, es decir, la MTU más pequeña de cualquier enlace en el camino hacia el extremo distante.

En el RFC 1191 se define el descubrimiento de ruta MTU (es decir, Path MTU Discovery), el cual es un proceso simple donde un host puede detectar una ruta MTU más chica que su interfaz MTU.

Para este proceso hay 2 componentes claves: el bit Don't Fragment (DF) de la cabecera IP y un código del mensaje ICMP → Destination Unreachable, Fragmentation Needed.

5. ¿Qué efecto presenta la opción -N ? ¿Qué tipo de aplicaciones pueden requerir tal utilidad?

La opción -N me permite activar la opción TCP_NODELAY, lo que deshabilita el Algoritmo de Nagle.

El algoritmo de Nagle intenta evitar la congestión que los paquetes pequeños (denominados tinygrams) pueden ocasionar en la red reteniendo por poco tiempo la transmisión de datos TCP en algunas circunstancias.

Las aplicaciones que pueden sacar provecho de dicha utilidad puede ser por ejemplo, cuando una aplicación no pueda esperar o cuando se penalice mucho la pérdida de un paquete, esto puede ser el uso del ratón en un escritorio manejado de forma remota, o por ejemplo aplicaciones interactivas que transmiten pulsaciones de teclas únicas, como puede ser Telnet.

iptraf (estadísticas de uso de la Red)

iptraf es una herramienta para monitorizar redes IP. Intercepta los paquetes que cursan la red y presenta varias estadísticas acerca del tráfico actual en ella.

Para comenzar con la resolución del enunciado, primero procedo a instalar la herramienta iptraf, mediante el uso del comando

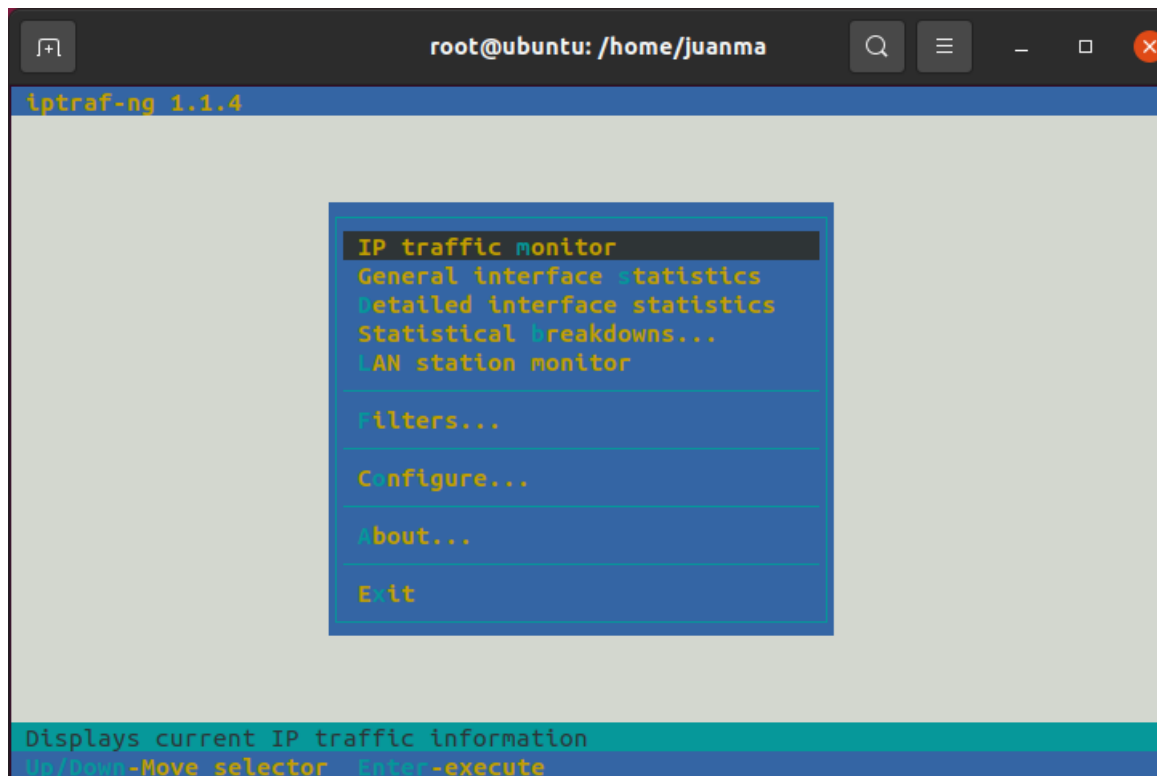
```
apt-get install iptraf
```

```
root@ubuntu:/home/juanma# apt-get install iptraf
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  iptraf-ng
The following NEW packages will be installed:
  iptraf iptraf-ng
0 upgraded, 2 newly installed, 0 to remove and 4 not upgraded.
Need to get 293 kB of archives.
After this operation, 763 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us.archive.ubuntu.com/ubuntu focal/main amd64 iptraf-ng amd64 1:1.1.4-6build1 [291 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 iptraf all 1:1.1.4-6build1 [1,912 B]
Fetched 293 kB in 2s (183 kB/s)
Selecting previously unselected package iptraf-ng.
(Reading database ... 186808 files and directories currently installed.)
Preparing to unpack .../iptraf-ng_1%3a1.1.4-6build1_amd64.deb ...
```

1. Inicie la utilidad mediante el comando iptraf-ng (como usuario root).

Para realizar esto, utilizo el comando

```
sudo iptraf-ng
```



2. Consulte las opciones “IP Traffic Monitor”, “Detailed Interface Statistics” y visite el sitio web de la UNLu y otros sitios. ¿Qué información proporciona cada opción?

```

juanma@ubuntu: ~
iptraf-ng 1.1.4
TCP Connections (Source Host:Port) ----- Packets --- Bytes Flag Iface
192.168.66.128:33212 = 6 559 --A- ens33
34.107.221.82:80 = 4 480 --A- ens33
192.168.66.128:33214 = 6 561 --A- ens33
34.107.221.82:80 = 4 398 --A- ens33
192.168.66.128:36438 = 12 1560 -PA- ens33
34.117.237.239:443 = 11 5942 --A- ens33
192.168.66.128:33826 = 9 1318 --A- ens33
13.227.92.110:443 = 7 6577 --A- ens33
192.168.66.128:36884 = 6 681 --A- ens33
72.247.40.235:80 = 4 1066 --A- ens33
190.104.80.1:80 = 4 640 -PA- ens33
192.168.66.128:57662 = 5 592 CLOS ens33
TCP: 18 entries ----- Active

UDP (89 bytes) from 192.168.66.2:53 to 192.168.66.128:44042 on ens33
UDP (101 bytes) from 192.168.66.2:53 to 192.168.66.128:33711 on ens33
UDP (98 bytes) from 192.168.66.2:53 to 192.168.66.128:59941 on ens33
UDP (118 bytes) from 192.168.66.2:53 to 192.168.66.128:39287 on ens33
UDP (83 bytes) from 192.168.66.2:53 to 192.168.66.128:47266 on ens33
Bottom ----- Elapsed time: 0:00
Packets captured: 610 | TCP flow rate: 0.12 kbps
Up/Dn/PgUp/PgDn-scroll B-more TCP info B-chg actv win S-sort TCP X-exit

```

La imagen anterior muestra las conexiones que hay en dicha interfaz (antes de visualizar el gráfico se debe seleccionar la interfaz) en formato de Socket Host Origen:Puerto, la cantidad de paquetes transmitidos, la cantidad de bytes que pesan esos paquetes, las flags activas, la interfaz utilizada, etc.

Abajo se detalla la cantidad de paquetes capturados.

```

juanma@ubuntu: ~
iptraf-ng 1.1.4
- Statistics for ens33 -----
Total      Total      Incoming   Incoming   Outgoing    Outgoing
Packets    Bytes      Packets    Bytes      Packets      Bytes
Total:      607       74799      304        23522      303         51277
IPv4:       601       72903      300        21802      301         51101
IPv6:        6         600        4          424        2           176
TCP:        525      63300      261        15569      264         47731
UDP:        82       10203      43         6657       39          3546
ICMP:        0          0          0           0          0           0
Other IP:    0          0          0           0          0           0
Non-IP:      0          0          0           0          0           0

Total rates:      0.00 kbps      Broadcast packets:      0
                  0 pps      Broadcast bytes:       0

Incoming rates:   0.00 kbps
                  0 pps

Outgoing rates:   0.00 kbps
                  0 pps
Elapsed time: 0
X-exit

```

En la segunda imagen, se muestra un resumen de los protocolos utilizados en el tráfico de dicha interfaz. La primera columna divide al total de paquetes en el tipo de protocolo utilizado, por ejemplo:

- Cuantos paquetes utilizaron el protocolo IPv4, cuantos utilizaron el protocolo IPv6 (ambos de capa de red) y cuantos utilizaron el protocolo ICMP (protocolo auxiliar de capa 3).
- Cuantos paquetes utilizaron el protocolo TCP y cuantos el protocolo UDP (ambos de capa de transporte).

Luego, la siguiente columna muestra el total de paquetes (salientes + entrantes) por cada protocolo.

La tercera columna detalla el total de bytes que pesaban dichos paquetes.

Las siguientes columnas muestran la separación del total anterior en paquetes entrantes y paquetes salientes acompañados del peso de c/u respectivamente.

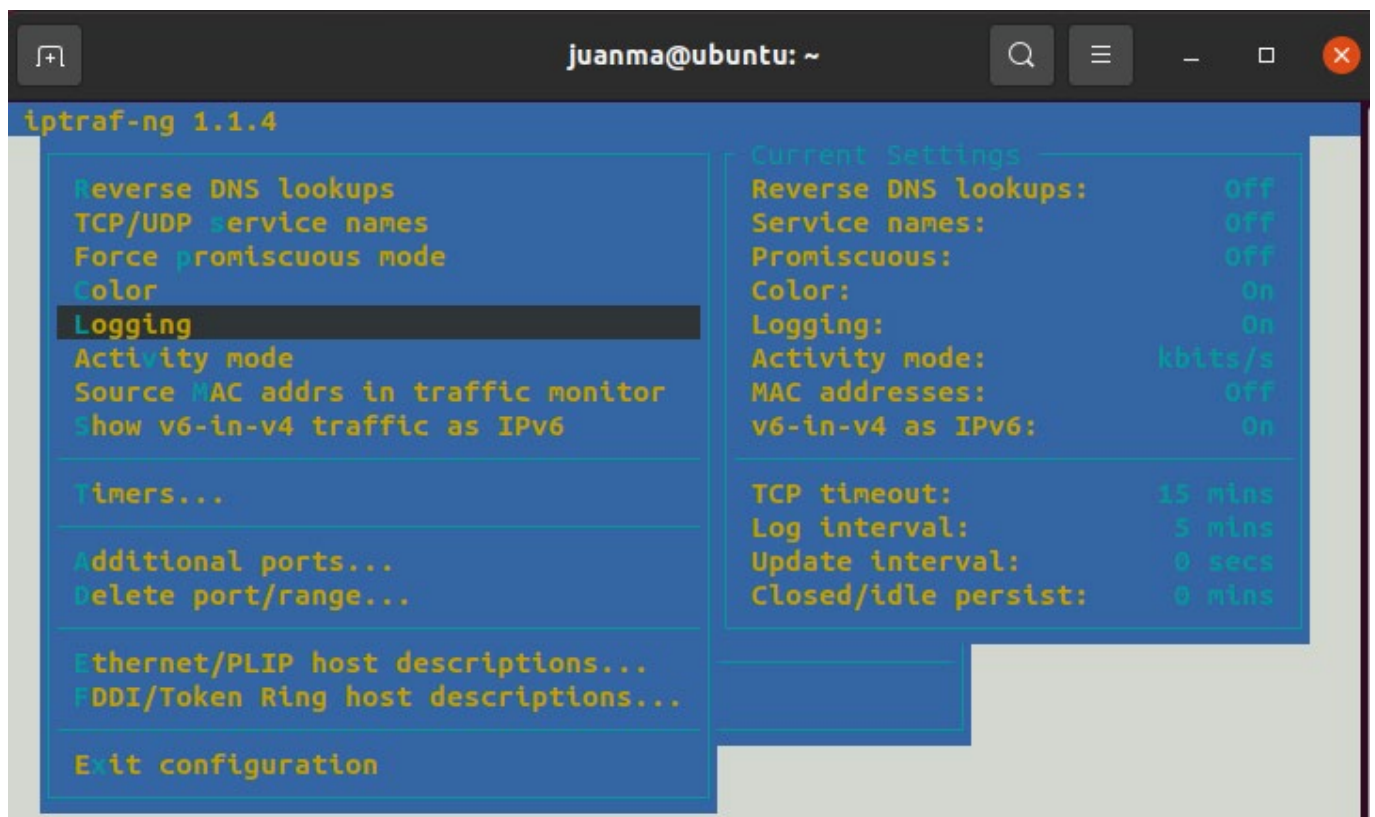
Mas abajo se muestran datos como cantidad de errores en el checksum de paquetes IP, cantidad de paquetes broadcast y bytes de los mismos, etc.

3. Configure la herramienta para que genere un archivo log de la información recuperada.

Vuelva a consultar las opciones de la consigna anterior. ¿En qué ruta por defecto se almacena tal información? ¿Para qué podría utilizarse?

Para realizar esto, debo ir a la opción “configure ...” dentro del menú principal.

Luego, busco la opción Logging y la activo:



La ruta por defecto en la que se almacena dicha información es dentro del directorio /var/log/iptraf/.

En mi caso, la herramienta no me dejaba utilizar dicho directorio (me tiraba un error) ya que no existía, por lo que tuve que crearlo manualmente.

Esto podría utilizarse para analizar todo el tráfico de la red durante un periodo más grande, analizar si existe alguna conexión sospechosa, algo fuera de lo normal.

ab (Test de Estrés para Servidores Web)

Herramienta para realizar benchmarking de Servidores Web. Se encuentra diseñada para proporcionar una aproximación del rendimiento actual del servidor, exhibiendo específicamente cuántas peticiones por segundo el mismo es capaz de servir.

1. Instale la herramienta ab (paquete apache2-utils en Debian).

En mi caso, la herramienta ya estaba instalada, pero en caso de no estarla, se puede instalar mediante el comando:

```
sudo apt-get install apache2-utils
```

```
juanma@ubuntu:~$ sudo apt-get install apache2-utils
[sudo] password for juanma:
Reading package lists... Done
Building dependency tree
Reading state information... Done
apache2-utils is already the newest version (2.4.41-4ubuntu3.4).
apache2-utils set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 4 not upgraded.
```

2. Realice una prueba contra el servidor web <https://eula-gtec.unlu.edu.ar/> efectuando 1000 peticiones con una concurrencia de 10 peticiones simultáneas. Nota: no omita la barra final de la dirección web.

Para realizar esto, debo utilizar el comando:

```
ab -n 1000 -c 10 https://eula-gtec.unlu.edu.ar/
```

El parámetro -n me permite especificar la cantidad de peticiones.

El parámetro -c me permite especificar la concurrencia de solicitudes simultaneas.

```
Server Software:      Apache
Server Hostname:      eula-gtec.unlu.edu.ar
Server Port:          443
SSL/TLS Protocol:     TLSv1.2,ECDHE-RSA-CHACHA20-POLY1305,2048,256
Server Temp Key:      X25519 253 bits
TLS Server Name:      eula-gtec.unlu.edu.ar

Document Path:        /
Document Length:      45966 bytes

Concurrency Level:     10
Time taken for tests:  134.648 seconds
Complete requests:     1000
Failed requests:       0
Total transferred:     46355000 bytes
HTML transferred:     45966000 bytes
Requests per second:   7.43 [#/sec] (mean)
Time per request:      1346.478 [ms] (mean)
Time per request:      134.648 [ms] (mean, across all concurrent requests)
Transfer rate:         336.20 [Kbytes/sec] received

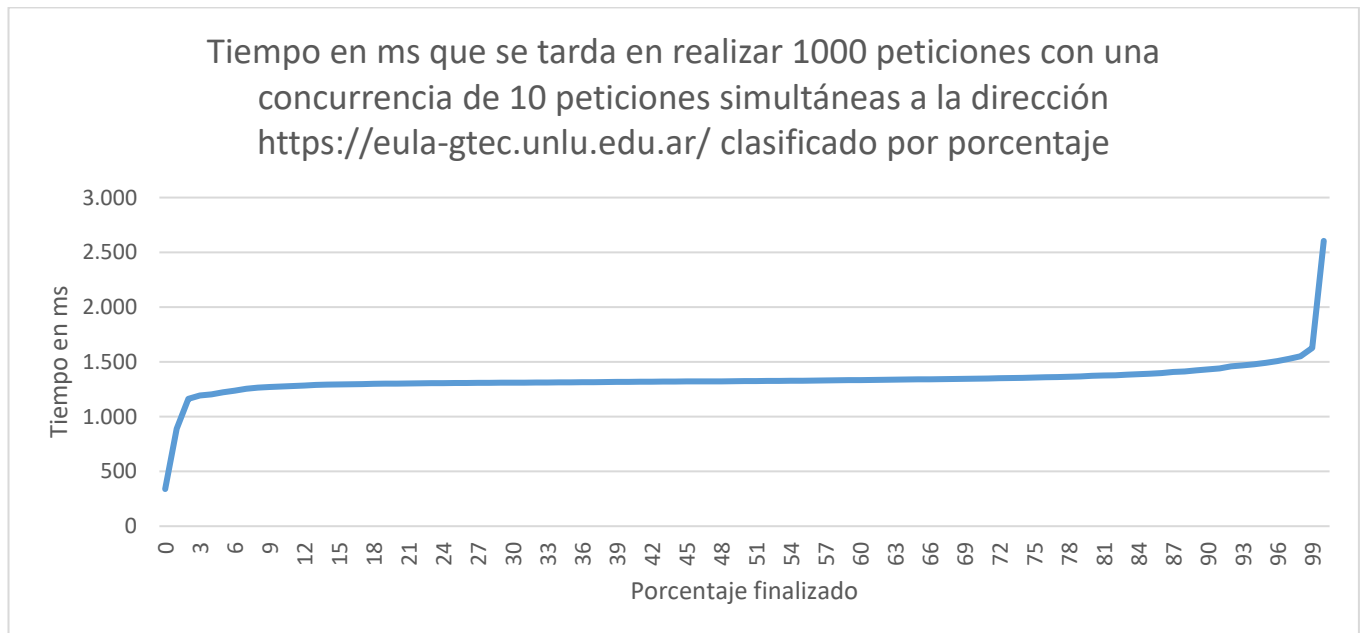
Connection Times (ms)
      min    mean[+/-sd] median    max
Connect:    67     88  34.7      85   1080
Processing: 411   1252 135.3    1233   2314
Waiting:    144    806 112.1     790   1858
Total:      512   1340 138.4    1318   2457
```


3. ¿Qué información proporciona la herramienta? Grafique la prueba realizada (Ayuda: Opciones -e , -g)

La información que proporciona la herramienta entre otras cosas es el retardo mínimo/promedio/máximo de la realización de las peticiones.

También el software utilizado por el servidor, el puerto en el que atiende solicitudes, versión del protocolo SSL/TLS, path del recurso que estoy solicitando, tiempo que tardo el test (el benchmarking), cantidad de solicitudes completas y fallidas, bytes transferidos, etc.

La herramienta detalla un análisis general de cuantas peticiones aproximadamente puede recibir por segundo un sitio web.



4. ¿Qué implica la utilización del parámetro -i? ¿Qué diferencia encuentra con la prueba de la consigna previa?

El parámetro -i me permite utilizar el método HEAD en lugar de GET.

Es decir, me cambia el tipo de petición que le estoy realizando al host destino.

La diferencia con la prueba anterior es que el método HEAD solo pide la cabecera, o sea, omite el cuerpo de la respuesta.

Esto me devuelve los valores de los campos content-length, content-type, etc.

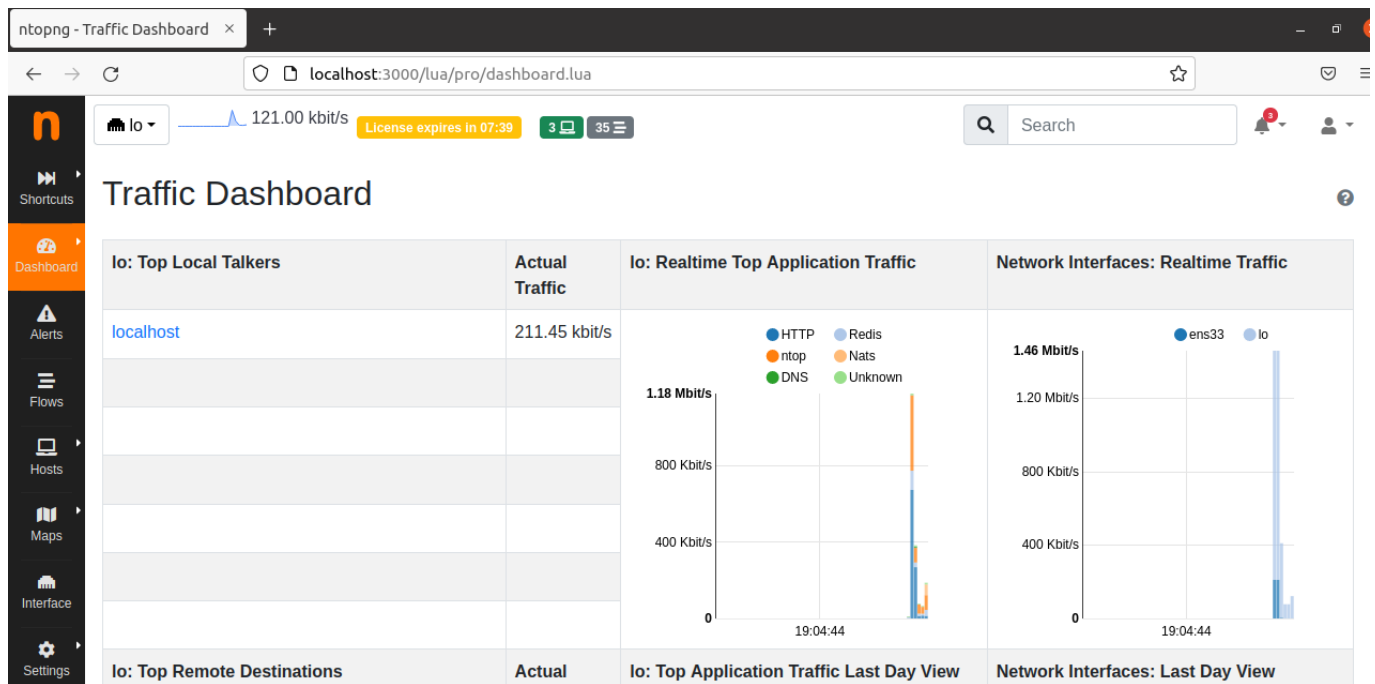
Utilizar el parámetro -i para utilizar el método HEAD reduce considerablemente el tiempo que tarda en realizarse la prueba.

ntop (estadísticas de uso de la Red)

Herramienta para el monitoreo y análisis de tráfico en la red. Provee una interfaz web para los reportes muy completa e intuitiva.

1. Instalar ntop en su distribución. El servicio levanta automáticamente, si no lo hace Iniciar ntop en su forma básica: como root o con sudo:
`ntop -i <interfaz de red>`

2. Luego ingrese vía web browser a <http://localhost:3000>.
Debería estar visualizando la interfaz de ntop.



Como se ve en la imagen anterior, efectivamente se visualiza la interfaz de Ntop.

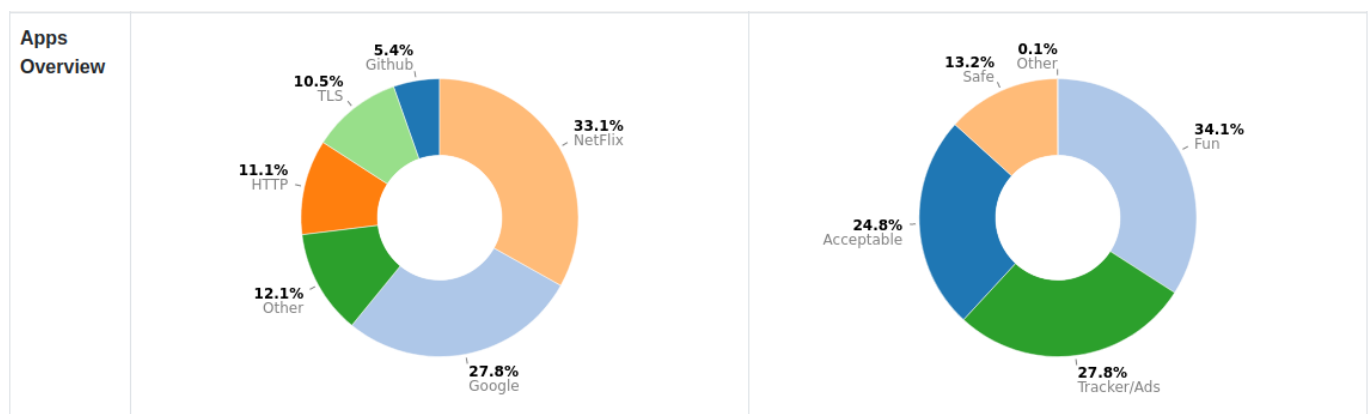
3. Revise los menús “Summary”, “All protocols” e “IP”. Comente muy brevemente las opciones que le resulten más útiles o interesantes. Si visualiza poca información, navegue por un par de sitios externos y vuelva a recargar la página de Ntop (F5).

Una de las opciones más útiles que encontré en la herramienta fue dentro del menú

Hosts → Selecciono el host deseado (localhost en este caso) → Apps

Se puede visualizar gran cantidad de datos acerca del tráfico de datos enviados y recibidos de cada aplicación.

Un gráfico que resume el % que representa cada servicio de la capa de aplicación, y una categoría que mide el grado de confiabilidad (supongo) de las aplicaciones.



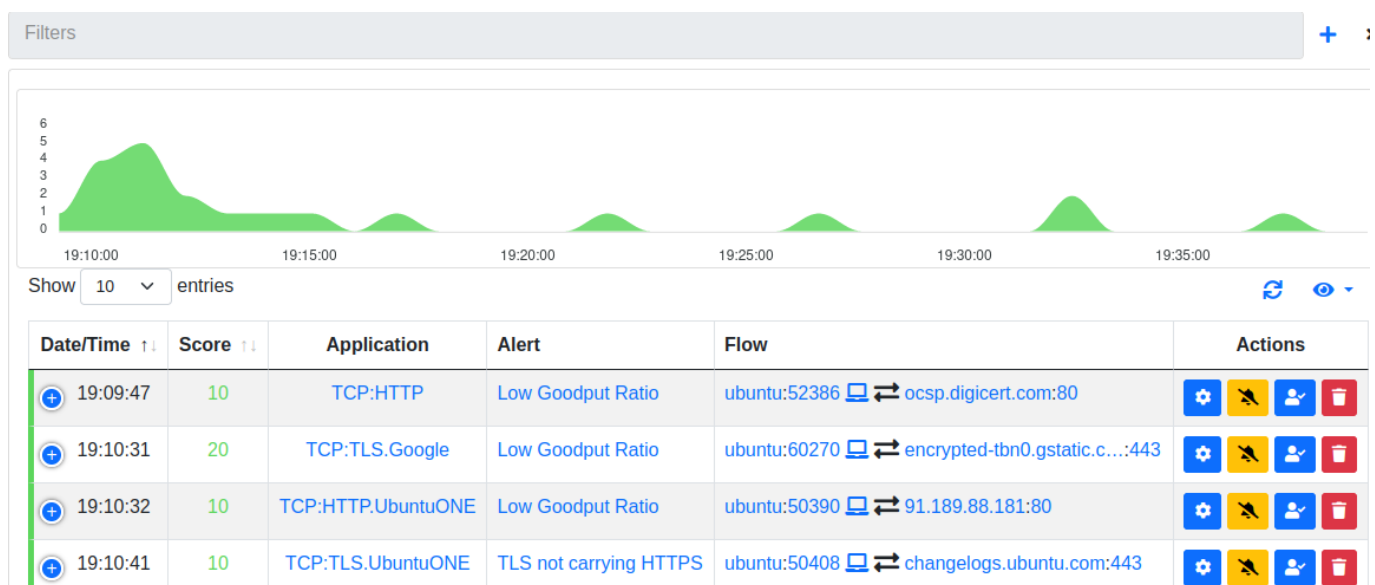
Luego, la parte que más me gustó, fue el cuadro que le sigue a la imagen de arriba:

Application	Duration	Sent	Received	Breakdown	Total	
Total	12:20	690.34 KB	5.61 MB	<div><div>Sent</div><div>Rcvd</div></div>	6.29 MB	
Amazon	03:15	51.65 KB	91.83 KB	<div><div>Sent</div><div>Rcvd</div></div>	143.48 KB	2.23 %
Cloudflare	00:45 sec	20.74 KB	139.87 KB	<div><div>Sent</div><div>Rcvd</div></div>	160.61 KB	2.49 %
DHCP	00:10 sec	692 Bytes	724 Bytes	<div><div>Sent</div><div>Rcvd</div></div>	1.38 KB	0.02 %
DNS	05:45	74.26 KB	118.44 KB	<div><div>Sent</div><div>Rcvd</div></div>	192.69 KB	2.99 %
Github	00:25 sec	56.79 KB	290.55 KB	<div><div>Sent</div><div>Rcvd</div></div>	347.33 KB	5.4 %
Google	03:10	208.58 KB	1.54 MB	<div><div>Sent</div><div>Rcvd</div></div>	1.75 MB	27.78 %
GoogleServices	00:15 sec	6.69 KB	7.92 KB	<div><div>Sent</div><div>Rcvd</div></div>	14.6 KB	0.23 %
HTTP	04:00	58.35 KB	654.18 KB	<div><div>Sent</div><div>Rcvd</div></div>	712.53 KB	11.07 %
MDNS	00:30 sec	627 Bytes	0 Bytes	<div><div>Sent</div><div>Rcvd</div></div>	627 Bytes	0.01 %
NetFlix 😊	01:25	113.96 KB	1.97 MB	<div><div>Sent</div><div>Rcvd</div></div>	2.08 MB	33.06 %
TLS	04:35	64.82 KB	608.94 KB	<div><div>Sent</div><div>Rcvd</div></div>	673.75 KB	10.46 %
UbuntuONE	01:00	5.58 KB	24.75 KB	<div><div>Sent</div><div>Rcvd</div></div>	30.33 KB	0.47 %
Unknown	01:50	3.93 KB	1.67 KB	<div><div>Sent</div><div>Rcvd</div></div>	5.61 KB	0.09 %
Wikipedia	00:20 sec	2.88 KB	25.58 KB	<div><div>Sent</div><div>Rcvd</div></div>	28.46 KB	0.44 %
YouTube 😊	00:10 sec	11.93 KB	50.59 KB	<div><div>Sent</div><div>Rcvd</div></div>	62.52 KB	0.97 %
ntop	00:20 sec	8.9 KB	138.6 KB	<div><div>Sent</div><div>Rcvd</div></div>	147.51 KB	2.29 %

La tabla de arriba muestra un resumen de cada aplicación, con datos como la duración, datos enviados, datos recibidos, la balanza entre estas 2 ultimas columnas, el total (la combinación de ambos) y el % que representan del total.

Luego, otra opción que me pareció muy útil fue, dentro de la categoría de Alerts → Tipo de alerta

Por ejemplo, si voy a Alerts → Flow, puedo visualizar lo siguiente:



El cual me muestra las alarmas dentro de la red, y me da la posibilidad de clasificar como atacante a algún host (en este caso no porque son de tipo Flow, pero si son Warnings lo permite).

4. ¿Porque cree que se necesita ejecutar con permisos de root?

Creo que se necesita ejecutar con permisos de root porque la información de la red debe ser accesible por pocas personas.

Por ejemplo, cualquier usuario podría acceder a las conexiones de dicha red y saber que aplicaciones está corriendo o que puertos tiene abiertos.

Además de esto, cualquier usuario podría acceder a la información referente a que dispositivos están conectados a la red, y podría reconstruir la topología de la red.

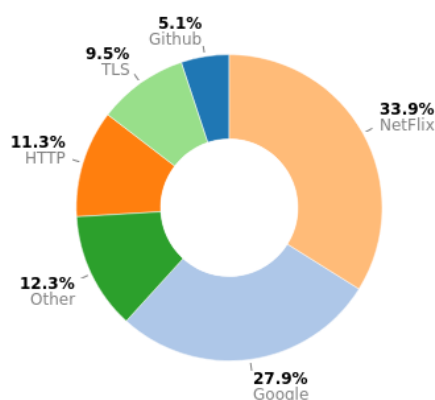
5. Para los siguientes requerimientos de información, aclare que opción de ntop es la mejor, y comente brevemente que información ofrece.

• Si necesita ver con ntop un resumen del tráfico de los protocolos de la Capa de Aplicación del stack TCP/IP, ¿a qué opción debería dirigirse?

Para ver esto, debería ir a la opción:

Dashboard → Traffic Dashboard → Applications

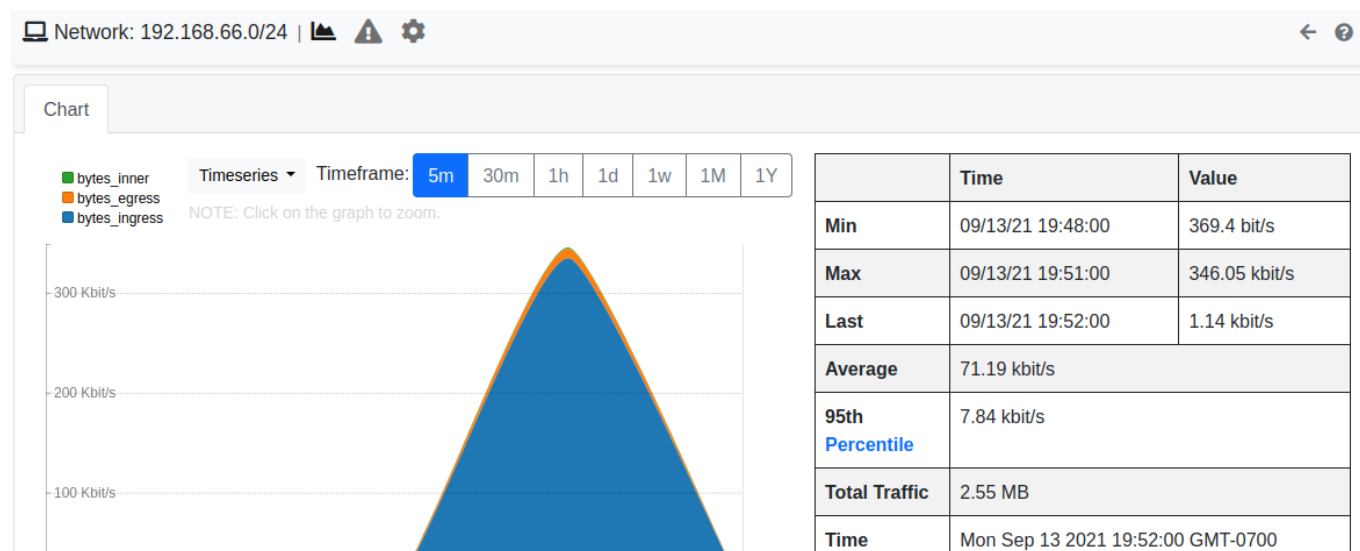
Top Application Protocols



• Si el administrador necesita revisar la actividad de la red por periodo de tiempo, cual listado de ntop ofrece una mejor visualización al respecto.

Para visualizar esto, debo ir a Hosts → Networks → Seleccionar la red → Historical

Se visualizará algo como lo siguiente:



En Timeframe puedo configurar el periodo de tiempo que quiero ver.

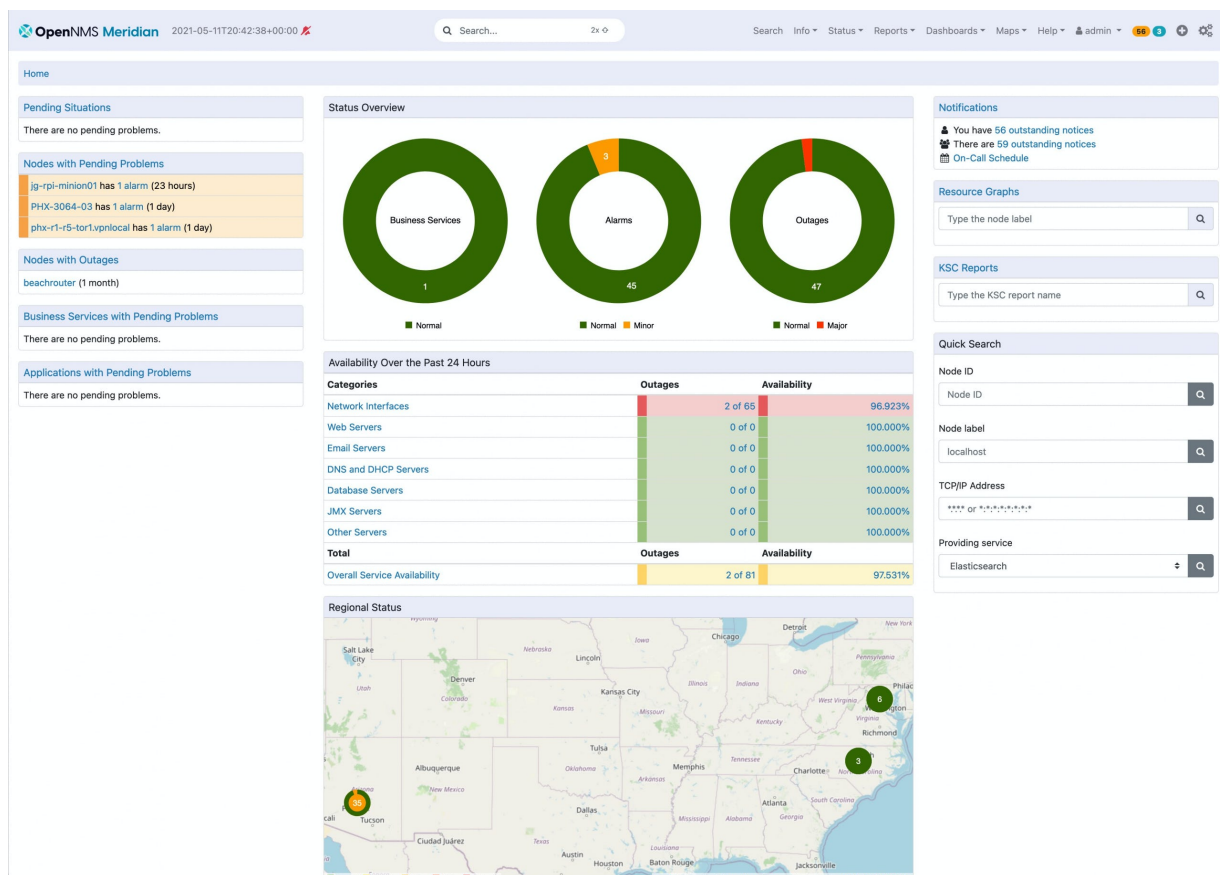
Herramientas gráficas

1. Investigue qué herramientas gráficas existen para monitorear redes y centros de datos. Seleccione una y comente sus funcionalidades.

Existen varias herramientas para monitorear redes y centros de datos, entre las cuales podemos destacar:

- Nagios Network Analyzer.
- ntopng.
- PRTG.
- Pandora FMS.
- NetCrunch.
- OpenNMS.
- Capsa

En mi caso, opté por elegir OpenNMS.



Esta herramienta, cuenta con múltiples funcionalidades como pueden ser:

- Actualizaciones en tiempo real, para atender cualquier problema lo antes posible y visualizar el estado actual de la red,
- Alertas de correo electrónico, para obtener información al instante en caso de que haya alguna alarma,
- Creación de diagramas, en caso de que quiera realizar un informe la herramienta permite realizar diagramas de manera automática,
- Registros de eventos, para poder visualizar que eventos ocurrieron y en que determinado momento, entre otras funciones más.