



DEPARTAMENTO  
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

# Trabajo Práctico 2

## Rutas en Internet

8 de diciembre de 2015

Teoría de las comunicaciones

Integrante	LU	Correo electrónico
González, Sergio Martín	723/10	sergiogonza90@gmail.com
Ladelfa, Hernán Nahuel	318/04	nahueladelfa@gmail.com

Instancia	Docente	Nota
Primera entrega		
Segunda entrega		



**Facultad de Ciencias Exactas y Naturales**  
Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2610 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (+54 +11) 4576-3300

<http://www.exactas.uba.ar>

# Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. Primera Consigna: Caracterizando rutas</b>	<b>3</b>
2.1. Implementación de Traceroute . . . . .	3
2.2. Identificando enlaces Submarinos . . . . .	3
2.3. Rutas a explorar . . . . .	4
<b>3. Segunda Consigna: Gráficos y Análisis</b>	<b>5</b>
3.1. Rutas encontradas . . . . .	5
3.1.1. Universidad de Helsinki (Finlandia) . . . . .	5
3.1.2. Universidad de Oxford (Inglaterra) . . . . .	6
<b>4. Conclusiones</b>	<b>7</b>

# 1. Introducción

El objetivo del siguiente trabajo, es el de estudiar y poder monitorear rutas a nivel de red, y por sobre todo, encontrar rutas hacia hosts en otros continentes, por las que solo se puede acceder a través de enlaces submarinos. La idea es intentar encontrar estos enlaces, e intentar monitorear como se comportan a lo largo del día.

Para esto, es necesario implementar una herramienta *traceroute* basada en el protocolo ICMP (Internet Control Message Protocol), y que esta se pueda utilizar para la recopilación de los datos necesarios para luego poder hacer los análisis pertinentes. Este protocolo, el cual es el utilizado en la herramienta *ping* de cualquier sistema operativo, es un protocolo de control y notificación de errores, que corre sobre IP. ICMP cuenta con varios tipos de paquetes, pero nosotros solo vamos a enfocarnos en 3:

- ***echo-request***: Paquete utilizado en la herramienta *ping*. Este sirve para saber si un host se encuentra disponible o no. De estar disponible, el host receptor responde con un *echo-reply*.
- ***echo-reply***: Respuesta al envío del tipo de paquete anterior. Cuando un host recibe un paquete de tipo *echo-request*, este envía (de tenerlo habilitado) un paquete de este tipo.
- ***time-exceeded***: Los paquetes ICMP poseen un campo llamado TTL (time to live), el cual indica el tiempo de vida del paquete. Entonces, por ejemplo, si se envía un *echo-request* con un TTL de 3, luego de los 3 saltos de router, este paquete es descartado, y se envía un paquete de tipo *time-exceeded* al nodo que envió el paquete original.

Utilizando esta ultima propiedad de los paquetes ICMP, podemos implementar una herramienta que, envíe paquetes *echo-request* incrementando de a poco el TTL (inicialmente con 1), y quedarnos con las IPs origen de los paquetes *time-exceeded*, así poder averiguar la IP de cada salto que realiza el paquete al momento de ser enviado. Esto se realiza hasta que se obtiene un paquete de tipo *echo-reply*.

También se deben poder obtener los  $\Delta RTT$  entre cada hop de la ruta, para poder realizar un test de Grubbs, y encontrar posibles outliers, los cuales son potenciales enlaces submarinos. Para realizar esto, lo que se hace es acumular los RTT de cada respuesta de tipo *time-exceeded*, y luego:

$$\Delta RTT_i = RTT_i - RTT_{i-1}$$

En la etapa de análisis, una vez ya teniendo la información necesaria, se hará un análisis sobre la posible ubicación de los nodos que se encuentran entrelazados por un enlace submarino. Para esto se utilizará la herramienta Geoiptool <sup>1</sup> con la que se puede geolocalizar una IP.

Luego se realizará un análisis sobre la variación del RTT del enlace submarino, a lo largo del día. Para esto, se realizará un script que corra el traceroute cada 30 minutos, y así poder realizar un monitoreo sobre la ruta.

---

<sup>1</sup>[www.geoiptool.com/es/](http://www.geoiptool.com/es/)

## 2. Primera Consigna: Caracterizando rutas

### 2.1. Implementacion de Traceroute

Para el desarrollo de los próximos análisis, se implemento una herramienta *traceroute* sobre Scapy en Python 3. La idea consiste en lo siguiente: Enviar iterativamente paquetes ICMP *echo-request*, empezando con un TTL de 1 y incrementándolo hasta que nos llegue un paquete de tipo *echo-reply*. Por cada paquete enviado, nos guardamos la IP del host originario del paquete, y de esta manera podemos construir la ruta que lleva al mismo.

Sin embargo, hay cosas a tener en cuenta para poder realizar esta técnica. Primero, es posible que por el tráfico de la red en el momento de hacer la captura, o por otros motivos que están fuera de nuestro control, el paquete pueda cambiar de ruta en los distintos instantes del envío de los paquetes ICMP. Otra cosa a tener en cuenta, es que si nosotros queremos hacer un estudio sobre el RTT de cada enlace ( $\Delta RTT$ ), no nos sirve el RTT medido para 1 solo intento, ya que esto pudo también haber sido influenciado por el tráfico momentáneo de la red.

Para intentar aliviar esto, lo que hacemos es, no enviar solo 1 paquete por cada TTL, sino que enviamos varios. Cuantos?, los necesarios para poder realizar un promedio lo mas confiable posible del  $\Delta RTT$ . Lo que hacemos es, establecer una cota inferior y una cota superior a la cantidad de paquetes a enviar de un mismo TTL:

- **MAX\_ATTEMPTS**: Cantidad máxima de paquetes **exitosos** a enviar de un mismo TTL. Esto quiere decir que, si mandamos MAX\_ATTEMPTS paquetes, y siempre nos llega una respuesta del mismo nodo, entonces promediamos el RRT, anotamos esta IP, y pasamos al próximo TTL.
- **MIN\_ATTEMPTS**: Cantidad mínima de paquetes provenientes del mismo nodo que debemos recibir para pasar al próximo TTL. Esto quiere decir, que si nosotros enviamos MIN\_ATTEMPTS paquetes, y todos vinieron del mismo nodo, y en MIN\_ATTEMPTS + 1, nos viene de un nodo diferente a los anteriores, de todas maneras promediamos los RTT, y avanzamos de TTL. En caso de que la IP cambie antes de llegar a los MIN\_ATTEMPTS, se descarta la información recopilada para este TTL y se vuelve a comenzar.

Cabe aclarar que este método igual tiene una falla, y es que si al momento de cambiar la IP, nosotros igual proseguimos al próximo TTL, o mismo cuando no llegamos a los MIN\_ATTEMPTS, y debemos reiniciar el TTL, va a haber un  $\Delta RTT$  que quizá no sea el de 1 enlace físico, ya que la ruta cambio. Por eso es que junto con la información recopilada, nos quedamos también con los intentos que se realizaron para cada salto, así podemos identificar cuando pase esto. Como el objetivo del TP es si es estudiar enlaces submarinos, no es tan importante que la ruta cambie en un momento determinado, siempre y cuando esto no suceda en los extremos del enlace submarino.

### 2.2. Identificando enlaces Submarinos

Para poder identificar enlaces submarinos, nos basamos en un teste denominado “Test de Grubbs”, el cual sirve para detectar outliers en una muestra aleatoria. En nuestro caso, la muestra aleatoria serán los  $\Delta RTT$  obtenidos en todo el traceroute, y el outlier que queremos encontrar, será justamente el del enlace submarino, ya que estimamos que este enlace debe tener un RTT mucho mayor a cualquier enlace que se encuentre en tierra.

Para realizar el test de Grubbs, es necesario que la muestra aleatoria sea de distribución normal, pero de todas maneras, la cátedra indico que igual usemos el test si la distribución de los  $\Delta RTT$  no nos queda normal. Ahora para verificar si la distribución de nuestras muestras es normal, utilizaremos el modulo Scipy de Python, el cual permite hacer análisis estadísticos. El test de Grubbs consiste en lo siguiente, dada una muestra  $X$  de tamaño  $N$ :

1. Se calcula el promedio muestral de  $X$ :  $\mu$
2. Se calcula la desviación standard de  $X$ :  $\sigma$
3. Se calcula el estadístico del test:  $G = (\max(X) - \mu)/\sigma$
4. Se calcula el valor de rechazo del test:  $C = \frac{n-1}{\sqrt{N}} * \sqrt{\frac{t_{\alpha/N, N-2}^2}{N-2+t_{\alpha/N, N-2}^2}}$
5. Si el valor de  $G$  es mayor a  $C$ , entonces  $\max(X)$  es un outlier de  $X$

El valor  $t_{\alpha/N, N-2}$  hace referencia al valor crítico de la distribución  $t$  para  $N-2$  grados de libertad, y nivel significativo  $\alpha$

### 2.3. Rutas a explorar

Teniendo implementadas las herramientas para recopilar la información y realizar los cálculos anteriormente dichos anteriormente, debemos elegir 2 IPs ubicadas en otro continente para poder realizar el *traceroute*. Las IPs elegidas son las siguientes:

- Finlandia: University of Helsinki - [www.helsinki.fi](http://www.helsinki.fi)
- Inglaterra: University of Oxford - [www.ox.ac.uk](http://www.ox.ac.uk)

Una vez corrido el traceroute a estas IPs, tendremos una ruta aproximada hacia las mismas. Cabe aclarar que la herramienta no es exacta, sino que solo es una aproximación de la ruta real, ya que para poder encontrar la ruta exacta con todos los nodos involucrados, es necesario usar técnicas mas complejas.

Una vez tengamos encontrada una ruta, lo que hacemos es geolocalizar las IPs de la misma, para poder comparar esto con el resultado que nos devuelva el Test de Grubbs, y así corroborar que RTT del enlace que vemos como outlier, tenga sus extremos en distintos continentes. Para esto, utilizaremos Geo IP Tool ([www.geoiptool.com](http://www.geoiptool.com)), junto con IP Location ([www.iplocation.net](http://www.iplocation.net)), las cuales nos permiten ubicar geográficamente una IP. IP Location, a diferencia de Geo IP Tool, presenta 4 posibles opciones de donde puede estar la IP, sacadas de otras paginas Web.

### 3. Segunda Consigna: Gráficos y Análisis

Luego de la breve explicación de como se han implementado las herramientas a utilizar, y de haber propuesto 2 rutas a analizar, procedemos a mostrar los resultados obtenidos de las mismas. Para poder realizar los siguientes análisis, lo que hicimos fue: Correr la herramienta *traceroute* a lo largo del día, cada media hora con un script, durante 12 horas.

#### 3.1. Rutas encontradas

Primero, vamos a presentar las rutas encontradas para las IPs propuestas. Se presentaran en formato tabla, y se indicara la ubicación de cada IP según Geo IP Tool.

##### 3.1.1. Universidad de Helsinki (Finlandia)

El siguiente cuadro muestra como el enlace submarino se debería encontrar entre Argentina y Italia. Esta ruta, fue el resultado de 27 corridas de *traceroute* a lo largo del día, por lo que es lo más certero que da nuestra herramienta. En la ruta se ve como al llegar al continente europeo, las IPs van saltando entre IPs locales de Suecia y Finlandia, hasta llegar al host destino.

Una cosa a destacar, es la aparición de varios  $\Delta RTT$  negativos, incluso habiendo promediado los RTT de 10 paquetes distintos, y que además como se puede ver, todos fueron exitosos, y solo hubo 1 cambio de IP al inicio del *traceroute*, en donde luego de 8 intentos cambio la IP. Esto muestra como por más de que se promedien varios RTT, la cercanía de estos nodos, hacen que los RTT varíen mucho. Para eso está bueno también tener el valor del desvío standard de los intentos, así tenemos una aproximación de entre que valores puede estar verdaderamente el RTT, con alta probabilidad.

Cuadro 1: Ruta para Universidad de Helsinki

TTL	IP	Intentos	RTT Promedio	Desvío Standard	Delta RTT	Ubicación
1	10.0.0.1	8	59.12ms	15.03ms	59.12ms	Router Local (Argentina)
2	10.24.128.1	10	55.40ms	4.30ms	-3.73ms	Router Local (Argentina)
3	181.47.254.85	10	58.20ms	6.09ms	2.80ms	Argentina (Bs As)
4	195.22.220.93	10	60.10ms	6.01ms	1.90ms	Argentina (Tigre)
5	195.22.220.92	10	63.11ms	7.41ms	3.01ms	Argentina (Tigre)
<b>6</b>	<b>149.3.183.11</b>	<b>10</b>	<b>271.50ms</b>	<b>4.70ms</b>	<b>208.39ms</b>	<b>Italia</b>
7	No hubo respuesta	-	-	-	-	-
8	109.105.97.126	10	324.10ms	2.60ms	52.60ms	Suecia
9	109.105.102.102	10	336.80ms	8.99ms	12.70ms	Suecia
10	109.105.102.103	10	333.20ms	17.01ms	-3.60ms	Suecia
11	193.167.253.9	10	333.80ms	8.51ms	0.60ms	Finlandia
12	128.214.173.242	10	336.40ms	3.44ms	2.60ms	Finlandia (Helsinki)
13	128.214.173.10	10	333.22ms	12.11ms	-3.18ms	Finlandia (Helsinki)
14	128.214.189.85	10	328.40ms	5.15ms	-4.82ms	Finlandia (Helsinki)
15	128.214.189.90	10	337.20ms	5.81ms	8.80ms	Finlandia (Helsinki)

##### 3.1.2. Universidad de Oxford (Inglaterra)

A diferencia de la ruta provista en el punto anterior, esta resultó ser bastante más errática. Para empezar, el enlace que tiene el mayor  $\Delta RTT$ , tiene ambos extremos en Estados Unidos, uno en Virginia y el otro en Kansas (según Geo IP Tool y IP Location). Por otro lado, hay otro salto grande que se observa, que es entre el hop 7 y 9, y aquí si ambos extremos están en distintos continentes. El hop 8, en escasa ejecución realizadas a lo largo del día obtuvimos algunas respuestas de él, pero las herramientas de geolocalización indicaban que la IP podía estar en Estados Unidos, o en Alemania, lo cual nos dejaba con un grado de incertidumbre bastante grande.

De todas maneras, según se ve, el enlace debe estar entre Estados Unidos y Inglaterra, o quizá entre Estados Unidos y Alemania, pero no sabemos certeramente cuáles son las IPs de sus extremos.

Cuadro 2: Ruta para Universidad de Helsinki

TTL	IP	Intentos	RTT Promedio	Desvio Standard	Delta RTT	Ubicación
1	10.0.0.1	8	55.88ms	17.37ms	55.88ms	Router Local (Argentina)
2	10.24.128.1	10	65.80ms	6.48ms	9.92ms	Router Local (Argentina)
3	181.47.254.85	10	60.90ms	6.49ms	-4.90ms	Argentina (Bs As)
4	208.178.195.214	10	65.50ms	6.02ms	4.60ms	Estados Unidos (Virginia)
5	208.178.195.213	10	60.50ms	9.51ms	-5.00ms	Estados Unidos (Virginia)
6	67.17.75.66	10	223.20ms	53.45ms	162.70ms	Estados Unidos (Kansas)
7	4.68.111.121	10	202.20ms	16.02ms	-21.00ms	Estados Unidos (Chicago)
8	No hubo respuesta	-	-	-	-	-
9	212.187.139.166	10	276.00ms	7.33ms	-26.43ms	Inglaterra (Londres)
10	146.97.33.2	10	283.70ms	7.07ms	7.70ms	Inglaterra (Londres)
11	146.97.37.194	10	293.50ms	15.26ms	9.80ms	Inglaterra (Londres)
12	193.63.108.94	10	275.20ms	4.10ms	-18.30ms	Inglaterra (Gales)
13	193.63.108.98	10	285.00ms	7.50ms	9.80ms	Inglaterra (Gales)
14	193.63.109.42	10	294.70ms	10.61ms	9.70ms	Inglaterra (Gales)
15	192.76.21.71	10	285.30ms	4.92ms	-9.40ms	Inglaterra (Oxford)
16	192.76.22.200	10	289.90ms	6.54ms	4.60ms	Inglaterra (Oxford)
17	192.76.32.62	10	285.90ms	4.86ms	-4.00ms	Inglaterra (Oxford)
18	129.67.242.154	10	289.50ms	8.90ms	3.60ms	Inglaterra (Oxford)

## 4. Conclusiones