Taller 2: Rutas en Internet

Teoría de las Comunicaciones
Departamento de Computación
FCEN - UBA

19.09.2023

1. Objetivos

En este trabajo práctico nos proponemos experimentar con herramientas y técnicas de uso frecuente a nivel de red. Particularmente, la versión de `traceroute` basada en los mensajes *echo request/reply* del protocolo ICMP [1, 2].

Los objetivos son múltiples. Por un lado entender los protocolos involucrados y desarrollar nuestras propias implementaciones para afianzar los conocimientos. Por otra parte, razonar sobre lo hecho y comprender mejor qué pasa con los protocolos involucrados. Para esto, se deberá realizar todo lo anterior en un marco analítico y experimental.

2. Normativa

- Fecha de entrega: hasta el **21/10/2023**.
- El trabajo práctico se deberá enviar por correo electrónico con el siguiente formato:
  to: tdc-doc at dc uba ar
  subject: debe tener el prefijo [tdc-rutas]
  body: nombres de los integrantes y las respectivas direcciones de correo electrónico. También pueden agregar una oración explicando en cual parte del trabajo tuvo mayor participación cada integrante.
  attachments: el informe en formato pdf + el código fuente en formato zip.
- No esperar confirmación a menos que reciban una respuesta indicando explícitamente que el mail fue rechazado. Notar que los avisos por exceso de tamaño no son rechazos.

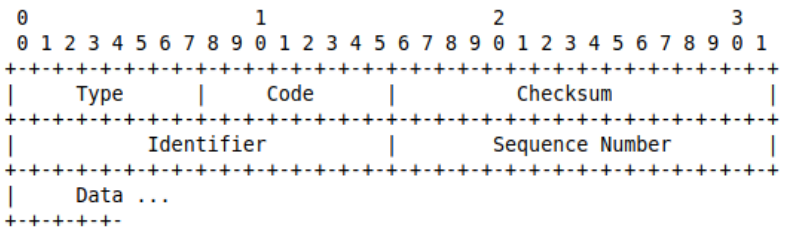
3. Enunciado

3.1. Introducción

Gracias a los protocolos de ruteo, las rutas en Internet son capaces de adaptarse a los cambios topológicos pero, a su vez, esto da lugar a un dinamismo que hace muy difícil el diagnóstico de los problemas que van surgiendo. Para lograr obtener información de forma remota acerca de los dispositivos que componen las redes se diseñó el *Internet Control Message Protocol* (ICMP) [1] que actualmente forma parte integral del núcleo de la arquitectura TCP/IP. El protocolo ICMP sólo busca proveer mensajes de control y no intercambia datos de usuario.

Si bien ICMP **debe** ser implementado por cada módulo IP, suele pasar que a la hora de configurar/administrar cada red se desactiven por cuestiones de eficiencia, dado que atender estos mensajes puede provocar retrasos en el *forwarding* de paquetes con datos. Los paquetes ICMP pueden ser enviados tanto por routers como por hosts y son generados por varias razones como errores en los datagramas IP, para comunicar información de diagnóstico, etc. Siempre se envían de regreso a la dirección origen del datagrama IP que motivó el mensaje.

Los paquetes ICMP constan de un header de 8 bytes y a continuación, una sección variable que depende de cada tipo de mensaje, como se indica a continuación:



Los primeros 3 campos son propios de todos los mensajes ICMP. El campo `Type` (1 byte), indica el tipo del mensaje y define el formato de lo que sigue. El campo `Code` (1 byte), especifica el subtipo. El campo `Checksum` (2 bytes) usa el algoritmo de IP sobre el header más los datos del paquete ICMP. Los restantes 4 bytes dependen del tipo del mensaje. Actualmente el protocolo ICMP consta de mas de 20 tipos de mensajes distintos, en el caso del ejemplo presentado arriba, se ven los campos de los mensajes `Echo Request` y `Echo Reply`, que son los que se usan para implementar la herramienta `ping`. En este caso, los 4 bytes restantes del header son, `Identifier` (2 bytes), permite asociar solicitudes con respuestas y `Sequence Number` (2 bytes), para numerar el número de intento de cada envío que se realice dentro de una misma tanda de pruebas. La sección de datos puede contener información arbitraria que debe ser devuelta en el `Echo Reply`, en general se copian los primeros bytes del `Echo Request`.

3.1.1. *TTL Time Exceeded* y `traceroute`

Otro de los tipos de paquetes famosos que tiene el protocolo ICMP es el *TTL Time Exceeded*. Este mensaje se genera cuando un determinado paquete IP alcanza en su campo TTL un valor igual a 0. Este campo sirve originalmente para descartar paquetes que transitaron por demasiados routers, que van restándolo a cada salto y cuando alcanza el valor de 0 lo descartan evitando que los paquetes ciclen infinitamente por Internet. Si el router que descarta el paquete tiene debidamente implementado el protocolo ICMP, entonces, además de descartar el paquete, envía de vuelta al origen un mensaje ICMP de tipo *TTL Time Exceeded* anunciando que el paquete fue descartado a causa del TTL excedido.

Teniendo esto en cuenta, es posible aprovechar esta funcionalidad y generar paquetes IP que vayan a una dirección en Internet, pero que tengan un TTL igual a 1. En este caso, el primer router que lo reciba lo descartará y enviará de nuevo al origen un mensaje ICMP anunciando dicho descarte. Lo interesante de este mecanismo es que el router que realiza el envío de este mensaje, lo hace poniendo como origen su propia dirección. Si ahora, hacemos lo mismo pero ponemos en el campo TTL el valor 2, entonces (si todo funciona bien) obtendremos un paquete que tiene como dirección IP de origen, la dirección del 2do router en la ruta al destino que originalmente tenía el paquete. Es posible hacer esto incrementalmente y así obtener las direcciones de todos los routers que componen una ruta.

Una de entre muchas técnicas que el comando `traceroute` implementa es la mencionada anteriormente, enviando `Echo Requests` con TTLs incrementales. Se puede ejecutar en la consola usando un nombre de dominio o una IP destino de la siguiente manera: `traceroute -M icmp 157.92.27.128` (puede requerir permisos de superusuario según la implementación). Luego de unos instantes se obtiene una salida como la siguiente:

```
 1  _gateway (192.168.0.1)  5.391 ms  5.822 ms  6.525 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  host69.181-96-120.telecom.net.ar (181.96.120.69)  28.038 ms  119.203 ms  17.449 ms
 7  * * *
 8  host63.181-96-120.telecom.net.ar (181.96.120.63)  102.367 ms  17.847 ms  19.744 ms
 9  VPN-corp.telmex.net.ar (200.49.69.157)  54.226 ms  55.475 ms  55.459 ms
10  be4-1-cf223-igw-01-bsas.claro.com.ar (131.100.186.97)  54.621 ms  54.631 ms  54.506 ms
11  * * *
12  * * *
```

```
13  * * *
14  157.92.47.53 (157.92.47.53)  22.874 ms  22.816 ms  20.675 ms
15  * * *
16  * * *
...
```

Como comentamos anteriormente puede ser que no todos los routers en la ruta implementen los mensajes *TTL Time Exceeded*, que en este caso sería lo que significan los asteriscos en los saltos que aparecen durante la ejecución. **Es posible verificar esto último usando Wireshark.** Así mismo, también es posible que el destino final no responda el `Echo Request` por lo cual estas herramientas suelen poner un límite de 30 saltos por defecto de manera de no esperar indefinidamente la respuesta.

3.2. Primera consigna: programando `traceroute`

A continuación se presenta a modo de ejemplo un script que realiza un `traceroute` usando la técnica de TTLs incrementales:

```python
#!/usr/bin/env python3

import sys
from scapy.all import *
from time import *

responses = {}
for i in range(2):
    print()
    for ttl in range(1,25):
        probe = IP(dst=sys.argv[1], ttl=ttl) / ICMP()
        t_i = time()
        ans = sr1(probe, verbose=False, timeout=0.8)
        t_f = time()
        rtt = (t_f - t_i)*1000
        if ans is not None:

            if ttl not in responses:
                responses[ttl] = []
            responses[ttl].append((ans.src, rtt))

            if ttl in responses:
                print(ttl, responses[ttl])
```

taller2.py

Para ejecutar el código, necesitamos pasar una IP destino como argumento, de la siguiente manera, por ejemplo:

```
$ python taller2.py 157.92.47.53
```

Luego de ejecutar el código y esperar unos instantes debería obtenerse una salida como la siguiente:

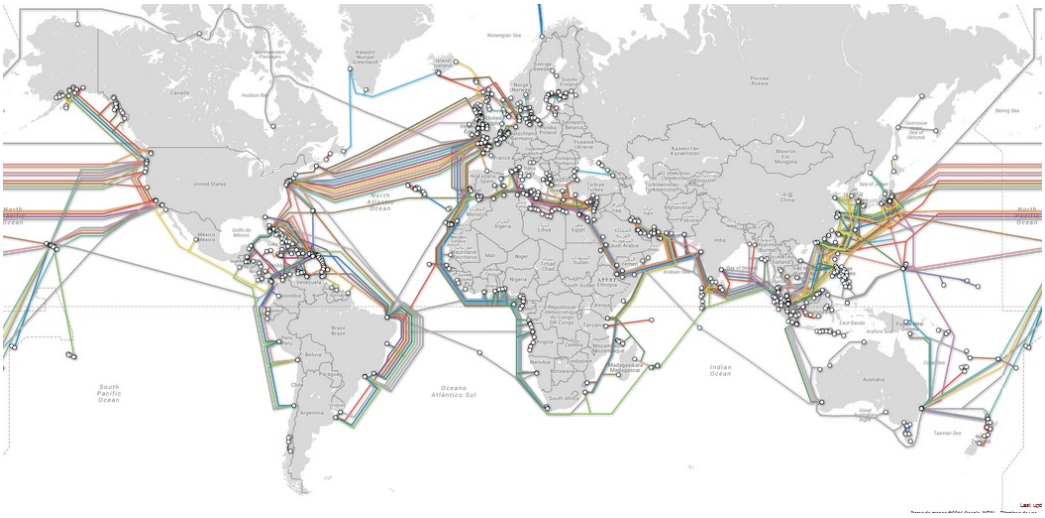
```

1 [('192.168.0.1', 102.68282890319824)]
6 [('181.96.120.69', 199.4168758392334)]
8 [('181.96.120.63', 51.299095153808594)]
9 [('200.49.69.157', 63.28415870666504)]
10 [('131.100.186.97', 55.377960205078125)]
14 [('157.92.47.53', 71.39706611633301)]
```

```
1 [('192.168.0.1', 102.68282890319824)]
6 [('181.96.120.69', 199.4168758392334), ('181.96.120.69', 55.33790588378906)]
8 [('181.96.120.63', 51.299095153808594), ('181.96.120.63', 59.3411922454834)]
9 [('200.49.69.157', 63.28415870666504), ('200.49.69.157', 71.3341236114502)]
10 [('131.100.186.97', 55.377960205078125), ('131.100.186.97', 67.03495979309082)]
14 [('157.92.47.53', 71.39706611633301), ('157.92.47.53', 71.37084007263184)]
```

Adaptar el código anterior como se crea necesario de manera que permita calcular los RTTs entre cada salto para los que se reciba una respuesta ICMP de tipo *Time exceeded*. Se recomienda enviar como mínimo 30 paquetes para un mismo TTL (ráfagas) y analizar las respuestas tanto para distinguir entre varias rutas como para obtener un valor de RTT promediado seleccionando para cada TTL las respuestas de la IP que más veces responda (en caso de haya más de una). Además, calcular el *RTT entre salto* restando los valores de RTT de saltos sucesivos. Tener en cuenta que esta resta puede dar un número negativo, en este caso se puede obviar el cálculo de *RTT entre saltos* y calcularlo con el próximo salto que de positivo.

3.3. Segunda consigna: Experimentación e Informe

Usando la herramienta desarrollada, analizar rutas a sitios web de universidades en diferentes continentes. Las rutas por las que pasen los paquetes en Internet, pueden atravesar más de un continente, y esos caminos hasta pueden llegar a pasar por enlaces que cruzan océanos, como se puede ver en la figura a continuación.



Estos saltos tienen un alto tiempo de propagación comparado con otros saltos de la ruta debido a las grandes distancias que recorren. Analizar datos de una ruta por cada integrante del grupo, intentando ubicar los saltos interoceánicos basandose solamente en los RTTs obtenidos y comparándolos con herramientas de geolocalización IP [3][4][5] de manera de verificar si los resultados obtenidos son consistentes. **Pueden no serlo y es parte del trabajo identificar dichos casos**.

El informe debe seguir la siguiente estructura, intentando cumplir con los límites de palabras sugeridos:

- **Introducción (máximo 200 palabras):** Breve explicación de los experimentos que se van a realizar.
- **Métodos y condiciones de los experimentos (máximo 400 palabras):** Explicación del código implementado y descripciones de las rutas. Se debe detallar la localización geográfica de cada universidad y las características de las pruebas -horario, día de la semana, etc.-
- **Resultados de los experimentos (máximo 600 palabras):** En esta sección deben presentarse figuras y/o tablas que muestren de manera integral los resultados observados. A modo de sugerencia, se puede mostrar un gráfico de *RTT entre saltos* que se deduce de restar los valores promediados a cada salto y/o *RTT total* a cada salto.

- **Conclusiones (máximo 200 palabras):** Breve reseña que sintetize las principales dificultades y descubrimientos.

A continuación se sugieren preguntas que se pueden intentar responder una vez obtenidas las rutas. Tener a bien transcribir en el informe en la sección de resultados, aquellas que hayan podido ser respondidas. Se valorará significativamente el planteo de nuevas preguntas.

- ¿Qué porcentaje de saltos no responden los *Time exceeded*? ¿Cuál es el largo de la ruta en terminos de los saltos que si responden?
- ¿La ruta tiene enlaces intercontinentales? ¿Cuántos?
- ¿Se observaron comportamientos anómalos del tipo descripto en la bibliografía sugerida [6]?
- ¿Se observaron otros comportamientos anómalos? Proponga hipótesis que permitan explicarlos.

3.4. Tercera consigna (OPCIONAL): Detección de enlaces interoceánicos

Extender la herramienta y el informe para que una vez terminada la determinación de la ruta, prediga automáticamente los enlaces intercontinentales recorridos basandose en la técnica de estimación de outliers propuesta por Cimbala [7]. Para esto deben usarse los valores de *RTT entre saltos* como la distribución probabilística sobre la cual se detectan los outliers. Los enlaces intercontinentales pueden llegar a distinguirse del resto como outliers dentro de dicha distribución. Algunas preguntas que se pueden intentar responder serían:

- ¿La distribución de *RTT entre saltos* presenta outliers según el método de Cimbala? ¿Cuántos?
- ¿Se corresponden los outliers con los enlaces intercontinentales? ¿Cuántos falsos positivos y falsos negativos hay?
- ¿Se aprecia alguna diferencia en la capacidad de detectar enlaces intercontinentales según el largo de la ruta?
- ¿Es posible mejorar las predicciones usando un valor de corte fijo para el valor $(X_i - \bar{X})/S$ en lugar del valor en la tabla τ?

A modo de sugerencia, en esta actividad se puede graficar, el valor $(X_i - \bar{X})/S$ (ver [7]) para cada salto en la ruta con respecto a la distribución de los *RTT entre saltos* de la ruta (los valores ya promediados).

Referencias

[1] RFC 792 (ICMP) `https://www.ietf.org/rfc/rfc792.txt`

[2] Traceroute (Wikipedia) `https://en.wikipedia.org/wiki/Traceroute`

[3] `https://www.geoiptool.com/es/`

[4] `https://www.ip2location.com/free/traceroute`

[5] `https://dazzlepod.com/ip/`

[6] `https://www.net.in.tum.de/fileadmin/TUM/NET/NET-2012-08-1/NET-2012-08-1_02.pdf`

[7] `https://www.me.psu.edu/cimbala/me345/Lectures/Outliers.pdf`