Rutas en redes: Utilizando traceroute y métodos estadísticos para detectar enlaces submarinos

Gonzalo Ciruelos Axel Maddonni Federico Patané Gabriel Thibeault

***Resumen*—En este trabajo implementaremos una herramienta para trazar la ruta de paquetes ICMP, es decir un `traceroute`. Además, utilizando los tiempos de los RTT de los paquetes, desarrollaremos un método estadístico para detectar los enlaces submarinos.**

***Index Terms*—`traceroute`, enlaces submarino, ICMP, RTT, IP**

I. Introducción teórica

En este trabajo nos proponemos usar un traceroute junto con herramientas estadísticas para intentar detectar enlaces submarinos en las rutas IP.

Nuestra implementación de traceroute se basará en paquetes ICMP de tipo `echo request`. Enviaremos, al destino especificado, paquetes ICMP echo request con distintos TTL, esperando que el paquete se quede sin tiempo en un nodo intermedio, y este nodo intermedio nos mande otro paquete ICMP (esta vez de tipo `time exceeded`) y de esta manera sabremos la dirección IP del nodo intermedio.

Más adelante introduciremos más en detalle cómo fue hecha la implementación de traceroute. Sin embargo, lo que más nos interesa del traceroute es el RTT entre cada nodo: como podemos tomar el tiempo que tarda un paquete en ir y volver de cada nodo intermedio de la ruta, podemos estimar el tiempo que tarda el salto entre cada nodo de la ruta. De esta manera, como nuestro objetivo es detectar enlaces submarinos, podemos pensar que una buena forma de detectarlos sería detectar cuando el salto tarda mucho.

Esto es efectivamente lo que haremos, tendremos una variable aleatoria que será el RTT relativo, es decir, entre saltos, y tomaremos todos los RTT relativos de nuestro traceroute como muestras de esa variable aleatoria. Luego, usando algún modelo de detección de outliers, intentaremos detectar mediciones que corresponden con enlaces submarinos.

Como método de detección de outliers utilizaremos el método descripto en [2]. Este método se basa en normalizar la variable aleatoria, para obtener una nueva variable aleatoria que llamaremos ZRTT. Es decir, supondremos que los RTT relativos son una $N(\mu,\sigma)$ para algún μ, σ y normalizaremos para obtener una $N(0,1)$.

G. Ciruelos e-mail: gonzalo.ciruelos@gmail.com
A. Maddonni, e-mail: axel.maddonni@gmail.com
F. Patané, e-mail: fedepatane20@gmail.com
G. Thibeault, e-mail: gabriel.eric.thibeault@gmail.com

$$ZRTT = \frac{RTT_i - \overline{RTT}}{s_{RTT}}$$

Notar que sin embargo este método es imperfecto, porque la variable aleatoria podría no ser (y de hecho no es) una normal.

Ahora bien, tenemos que elegir algún valor límite tal que si $ZRTT$ es mayor que ese valor, diremos que es un outlier, y en consecuencia ese enlace es un enlace submarino. Aquí, es donde el paper [2] propone utilizar la variable τ de Thompson. La τ de Thompson se calcula como sigue

$$\tau = \frac{t_{\frac{\alpha}{2}}(n-1)}{\sqrt{n}\sqrt{n-2+t^2_{\frac{\alpha}{2}}}}$$

Donde n es la cantidad de mediciones.

Todo esto resume la parte teórica del problema. De aquí en adelante nos adentraremos en la implementación de la herramienta, y en la experimentación.

II. Desarrollo

Como explicamos anteriormente, tendremos que, primero, implementar una herramienta de traceroute, y luego agregarle la funcionalidad de detección de enlaces submarinos.

A. Primera consigna

La primera consigna atañe al desarrollo de la herramienta de traceroute. Esta herramienta está basada, como dijimos antes, en paquetes ICMP echo request. Lo que hacemos es crear paquetes ICMP echo request con un TTL fijo usando scapy [1]. Por ejemplo, si creamos un paquete ICMP con TTL igual a 1, el paquete será siempre descartado por nuestro router, pues cuando llegue a él, el router decrementará el TTL y como es 0 lo descartará. Esto provocará que nuestro router nos mande un paquete ICMP de tipo time exceeded.

De esta manera, iremos incrementando el TTL, así los diversos nodos de la ruta irán devolviendonos paquetes ICMP de tipo time exceeded y nosotros podremos construir la ruta. Una vez que recibamos un paquete de tipo ICMP echo reply, nos detendremos. Además, si iteramos más de 30 TTLs, también nos detendremos, porque supondremos que el host al cual le estamos enviando el paquete ICMP está configurado para no responder a paquetes de echo request.

Como las rutas no son necesariamente la misma cada vez, para cada TTL enviaremos varios paquetes (30 por defecto), y nos quedaremos con el gateway del cual nos llegaron más paquetes ICMP echo reply.

Sorprendentemente, las rutas son bastante estables, y siempre recibimos los replies de los mismos gateways. Las mayores diferencias que notamos son internas a la red de nuestra ISP, en particular Fibertel.

B. Segunda consigna

Para la detección de enlaces submarinos (es decir, de outliers de nuestra variable aleatoria $ZRTT$), utilizamos exactamente las técnicas descriptas anterioremente en la introducción teórica.

Cabe destacar que en el cálculo de los ZRTTs, sólo tomamos valores no-negativos[1] de los RTTs entre saltos. Sin embargo, en los gráficos de RTTs que presentamos, los valores pueden ser menores a 0. De forma contraria las figuras proveen información confusa y difícil de interpretar.

Además, la segunda consigna concierne a la geolocalización de las IPs de la ruta. Para lograr esto, utilizamos una base de datos pública muy importante, conocida como MaxMind [**?**], que tiene una granularidad bastante alta (de ciudades). Esta base de datos, como todas las disponibles libremente tiene errores, que ya veremos en los experimentos.

C. Conceptos generales

???

D. Ōsaka daigaku

Este experimento corresponde a la Universidad de Osaka, localizada en Osaka, Japón. Más especificamente, nuestro traceroute es al departamento de Ciencias de la Computación de la Universidad de Osaka, que se encuentra en la dirección `www.es.osaka-u.ac.jp`.

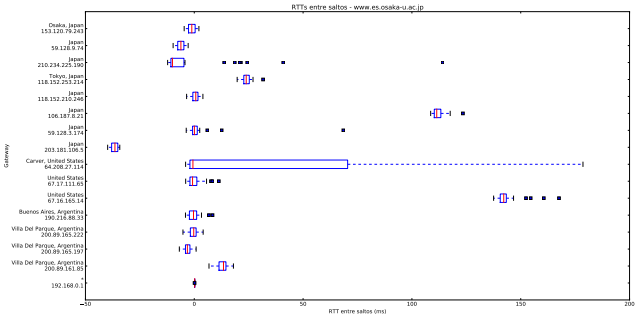

Figura 1
RTTs entre saltos. El valor asignado al i-ésimo nodo corresponde al salto entre el i-ésimo y el $i-1$-ésimo nodo. Para el primer nodo se utiliza simplemente su RTT.

[1] Consideramos como 0 a los valores negativos.

Se observan mediciones normales que explicaremos a continuación. En el gateway de Craver, Estados Unidos observamos algunas mediciones con un promedio muy alto. Tomamos las mediciones varias veces y siempre daba así. Se lo podemos atribuir a que quizás el gateway estaba muy sobrecargado en ese momento, dado que probamos varios días después y ya no sucedía.

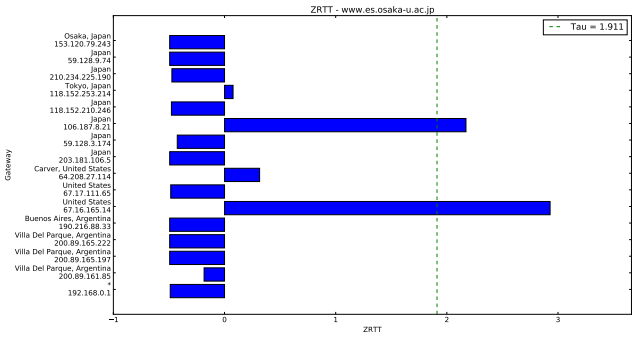

Figura 2
ZRTTs entre saltos.

Sin embargo, en el gráfico de $ZRTT$ podemos observar problemas con la geolocalización IP. Hay 2 IPs en la ruta que nos indican que el gateway está en Japón, pero recién el tercero está indicado como un enlace intercontinental. Esto nos permite afirmar, con bastante seguridad, que las dos primeras IP que según el servicio de geolocalización IP están en Japón (`203.181.106.5` y `59.128.3.174`), están en realidad en Estados Unidos.

El porcentaje de saltos que no responden los time exceeded es 30.43 % (exactamente 7 de 23). En términos de saltos que sí responden, la ruta tiene 16 nodos, incluyendo nuestro router.

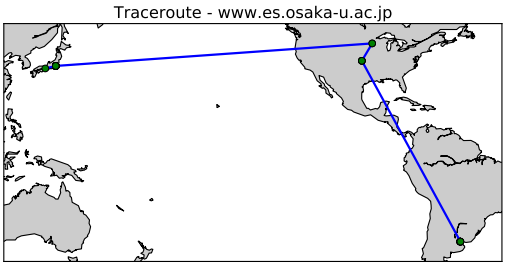

Figura 3
Ubicación geográfica estimada de la ruta tomada.

Como se ve, la ruta es aproximadamente la esperada. La ruta como bien se observa en el mapa, tiene 2 saltos intercontinentales, y como bien predijo nuestro modelo, tiene 2 enlaces submarinos, uno entre Argentina y Estados Unidos, y otro entre Estados Unidos y Japón.

En este caso el modelo fue extremadamente preciso: hay 2 enlaces submarinos y nuestro modelo los detectó, y no detectó ninguno más (es decir, no hubo falsos negativos).

E. Université de Laval

A continuación presentamos los gráficos resultantes de realizar una *traceroute* a la página de la Université de Laval[2].

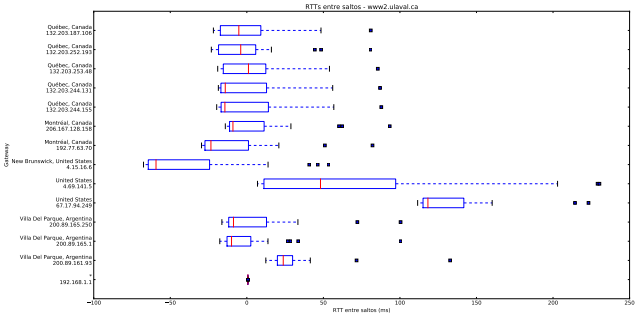

Figura 4
RTTs entre saltos. El valor asignado al i-ésimo nodo corresponde al salto entre el i-ésimo y el $i-1$-ésimo nodo. Para el primer nodo se utiliza simplemente su RTT.

En la realización de la *traceroute*, 6 de los 22 ($27.\overline{27}\%$) nodos no respondieron los *Time exceeded*.

Observamos una cantidad de *outliers* y una desviación consistente entre los distintos saltos en la figura 4. Esto posiblemente se debe a una congestión momentárea en la red durante ciertas repticiones de la *traceroute*, resultanto en que todos los RTTs de la repetición sean considerados *outliers*.

Si bien la mayoría de los nodos presentan desviaciones similares, en el de dirección 4.69.141.5 se observa una varianza excepcionalmente alta. Esto se puede deber a algunos de los factores mencionados en Jobst 2012[4]: métodos de balanceo de carga variando las rutas de los paquetes de diversas mediciones, o caminos de ida/vuelta asimétricos resultando en RTTs inconsistentes.

[2] www2.ulaval.ca.

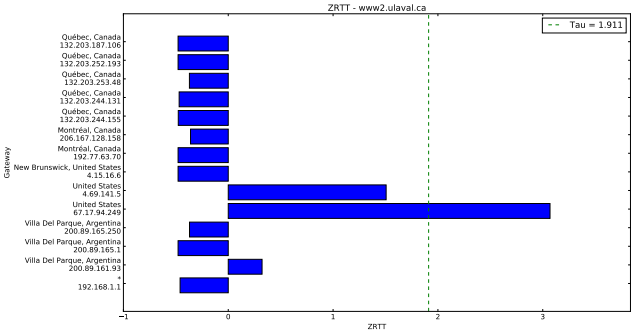

Figura 5
ZRTTs entre saltos.

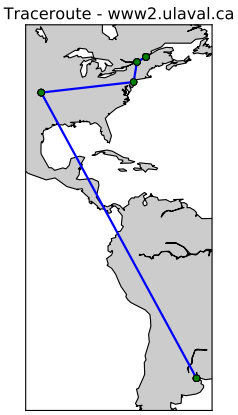

Figura 6
Ubicación geográfica estimada de la ruta tomada.

Sólo un salto (de 200.89.165.250 a 67.17.94.249) se presentó como *outlier* según el método planteado[2]. De acuerdo a la estimación geográfica, este salto se da entre Buenos Aires y Estados Unidos.

No hay ningún cable submarino que una directamente a Argentina con Estados Unidos. Sin embargo se pueden trazar rutas submarinas componiendo distintos cables submarinos entre ambos extremos[3].

Si bien estos dos hechos parecen contradecirse, Jobst 2012[4] provee una explicación: routers de *MPLS*[4] que no honran el campo *TTL*, y por ende no aparecen en la *traceroute*. Los cables submarinos manejan un flujo muy alto de paquetes, por lo que es esperable que busquen optimizar la performance; ignorar el campo *TTL* es una forma

[3] Por ejemplo, el South American Crossing (SAC) une a Argentina con Panamá y el Pan-American Crossing (PAC) a Panamá con Estados Unidos.

[4] *Multiprotocol Label Switching (MPLS)* es un protocolo empleado para simplificar el *forwarding*.

de logarlo. Esto resulta en que toda la ruta entre Buenos Aires y Estados Unidos se vea condensada en un único salto.

Cabe destacar que la herramienta geógrafica utilizada no pudo proveer la latitud y longitud de la dirección IP 67.17.94.249, por lo que el mapa emplea la próxima dirección ubicable, la 4.69.141.5.

F. Universidad de Alejandría

Presentaremos ahora los gráficos correspondientes al traceroute hacia la Universidad de Alejandría, ubicada en Alejandría, Egipto, África.

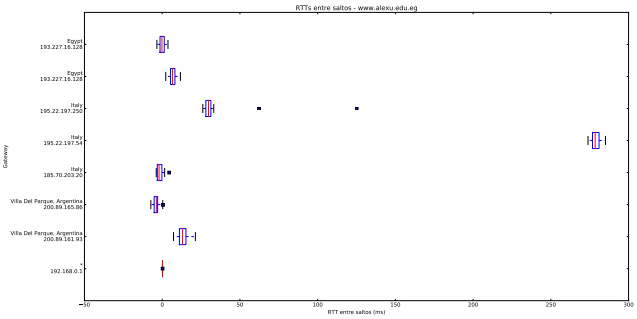

Figura 7

RTTs entre saltos. El valor asignado al i-ésimo nodo corresponde al salto entre el i-ésimo y el $i-1$-ésimo nodo. Para el primer nodo se utiliza simplemente su RTT.

Todas las mediciones son normales en todos los gateways, y no se observan outliers.

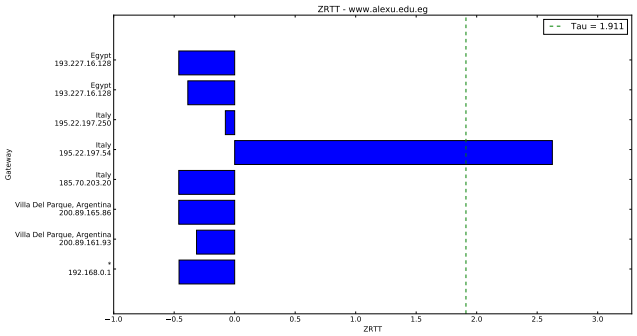

Figura 8

ZRTTs entre saltos.

En la figuta de la ruta hacia Alejandría podemos ver como tenemos problemas de geolocalización. Recién en la segunda IP que corresponde a Italia nos está marcando el enlace intercontinental, por lo cuál suponemos que esta primer IP italiana (185.70.203.20) en realidad se encuentra en Argentina. En el traceroute, 7 de los 22 nodos no respondieron el time exceeded.

Figura 9

Ubicación geográfica estimada de la ruta tomada.

En este caso tenemos dos enlaces intercontinentales según muestra el mapa, uno de Argentina hacia Italia, y el segundo de Italia hacia Egipto, pero solo detectamos 1 solo enlace, el de Argentina hacia Italia, teniendo así un falso positivo. Esto creemos que es debido a la cercanía de Italia con Egipto, a pesar de encontrarse en países distintos.

III. Conclusión

Referencias

[1] Scapy
http://www.secdev.org/projects/scapy
[2] Outliers - Cimbala, J.
http://www.mne.psu.edu/cimbala/me345/Lectures/Outliers.pdf
[3] MaxMind IP geolocalization
https://dev.maxmind.com/geoip/legacy/geolite/
[4] Traceroute Anomalies
https://www.net.in.tum.de/fileadmin/TUM/NET/NET-2012-08-1/NET-2012-08-1_02.pdf