Teoría de las Comunicaciones
TP2

31 de mayo de 2016

Integrante	LU	Correo electrónico
Martín Baigorria	575/14	martinbaigorria@gmail.com
Federico Beuter	827/13	federicobeuter@gmail.com
Mauro Cherubini	835/13	cheru.mf@gmail.com

Reservado para la cátedra

Instancia	Docente	Nota
Primera entrega		
Segunda entrega		

Índice

1. Introducción

Cuando un usuario accede a un sitio web o intenta transmitir información de un punto a otro por medio de la web, el mismo en general se abstrae de todos los mecanismos necesarios requeridos para esta transmisión. Dada la gran cantidad y heterogeneidad del hardware subyacente, en lo comienzos de la web se empezaron a diseñar distintos tipos de protocolos de comunicación para lograr unificar todos estos dispositivos en una red capaz de transmitir información desde cualquier punto a otro de forma relativamente confiable.

Hoy en día, el protocolo dominante para la transmisión de datos por la red es el IP (Internet Protocol). Creado por Vint Cerf y Bob Kahn en 1974, el protocolo IP esta diseñado para ser utilizado en redes de conmutación de paquetes y no esta orientado a conexión. Esto significa que al enviar un archivo, el mismo es fragmentado en paquetes que no necesariamente siguen la misma ruta en la red. Además, al no ser un protocolo orientado a conexión, no hay garantías de que los paquetes lleguen a destino dado que no hay ningún tipo de protocolo de handshake entre origen y destino.

En relación al modelo OSI, el protocolo IP pertenece a la capa de internet. Existen muchos otros protocolos que han sido construidos sobre el protocolo IP con el objetivo de proveer otro tipo de garantías y funcionalidades, entre los mas conocidos se encuentran el protocolo TCP (Transmission Control Protocol) y el UDP (User Datagram Protocol). Estos protocolos pertenecen a la capa de interconexión.

En el presente trabajo utilizaremos el protocolo ICMP (Internet Control Message Protocol), el cual esta especificado en el RFC 792 [3], con el objetivo de analizar las diferentes trazas y el RTT (round trip time) al momento de conectarse a un sitio web. Una traza se define como la sucesión de dispositivos de red que son recorridos, ya sean puentes, routers o gateways, al momento de transmitir información en la red. El protocolo ICMP esta implementado sobre IP, aunque se considera que el mismo no pertenece a la capa de interconexión a diferencia de TCP y UDP. Esto se debe a que su principal propósito no es intercambiar datos entre sistemas, si no que en general se utilizan para enviar mensajes de error entre dispositivos de red, con la excepción de su uso en herramientas como ping y traceroute.

1.1. Traceroute

La herramienta traceroute, desarrollada inicialmente por Van Jacobson en 1988, es una herramienta sumamente útil de diagnostico de red para buscar una aproximación de la traza de conexión y encontrar el RTT a cada hop (o salto) en la traza. Como ya hemos mencionado, esta herramienta utiliza el protocolo ICMP. A continuación discutiremos la estructura de los paquetes IP/ICMP para luego explicar como se hace efectivamente para identificar los diferentes hops. Luego discutiremos que potenciales problemas puede tener esta herramienta.

Como ya hemos mencionado, el protocolo ICMP esta implementado sobre IP. A continuación mostramos la estructura del header de un paquete IPv4. Todo lo que mencionaremos sigue siendo valido para IPv6.

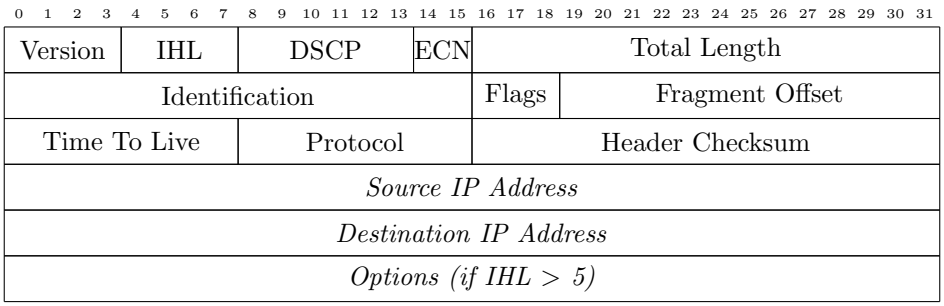


Figura 1: Paquete IPv4

Como podemos observador, el header de un paquete IPv4 tiene 24 bytes. En este momento, el campo que mas nos interesa es Time To Live (TTL). Como lo dice su nombre, este campo fue pensado para imponer un limite de tiempo a la vida del paquete en la red. Si no llegaba el paquete antes de ese tiempo, el mismo era descartado por el correspondiente hop. Sin embargo, en la practica esto se implemento como un limite a la cantidad de hops que un paquete podía recorrer. Esto se ve en los headers de los paquetes IPv6, donde el campo fue renombrado como Hop Limit.

En los paquetes IP/ICMP, al header de IP se le suma el header de ICMP. El mismo tiene la siguiente estructura:

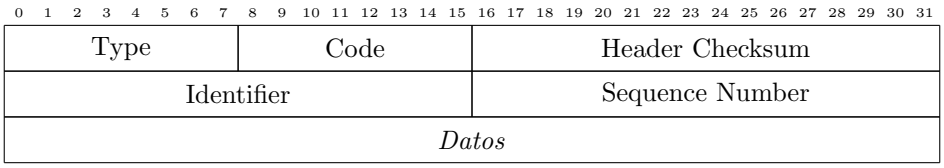


Figura 2: Paquete ICMP

El protocolo ICMP es parte del Internet Protocol Suite, especificado en RFC 792. Este header de 12 bytes se ubica luego del header de IP, teniendo un tamaño de header total de 36 bytes. La especificación explica en detalle como se utiliza el protocolo normalmente. Lo bueno de este protocolo es que si en algún momento header de IP llega a un $ttl = 0$, el hop correspondiente devolverá un mensaje de error al cliente de origen. En unos momentos veremos porque esto es sumamente útil.

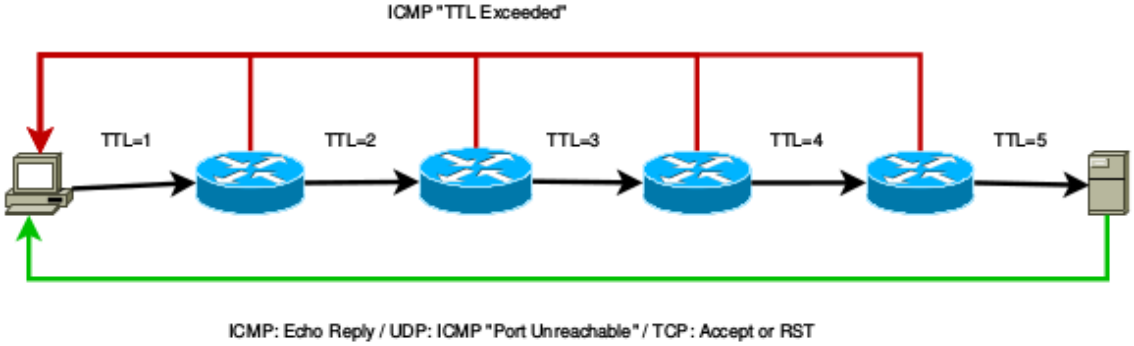
En nuestro caso, los siguientes casos de type seran los mas relevantes:

- 0 – Echo Reply
- 3 - Destination Unreachable
- 8 – Echo Request
- 11 – Time Exceeded

En principio, implementaremos nuestra propia herramienta traceroute utilizando paquetes ICMP. Para ello, enviaremos un Echo Request al URL (Uniform Resource Locator) al que queremos acceder y encontrar la traza utilizando el campo TTL del header IP.

Si el paquete llega al destino, el servidor nos enviara un paquete ICMP con el type Echo Reply. Para poder encontrar los diferentes hops en la traza, utilizaremos el parámetro Time To Live del header IP, inicializándolo en 1 y luego aumentándolo hasta que nos respondan con un Echo Reply. Es decir, si la red hasta el destino tiene 10 hops y queremos identificar el hop i, tendremos que settear el parametro $TTL = i$.

Notar que no necesariamente el servidor estara disponible o aceptara paquetes ICMP, por lo que tendremos que poner un limite a la cantidad de hops que buscaremos. Caso contrario iteraríamos hasta llegar al limite dado por los 8 bytes del campo TTL, lo cual no tiene sentido practico. Este procedimiento se puede ver un poco mejor en la siguiente imagen:



Para implementar esta herramienta utilizaremos Python, con la libreria scapy. Esta librería nos permite formar paquetes ICMP y luego hacer los respectivos requests.

Notar que el request lo mandaremos a un URL, por lo que el mismo se debe resolver a un IP mediante un request a un DNS (Domain Name System). Correr dos veces la herramienta no nos garantizara que hagamos el request a un mismo IP, dado que un sitio puede tener varios IPs asignados. Esto pasa normalmente con google.com. A su vez, en el ejemplo ilustrativo consideramos una topologia de red sumamente simple. Dado que las topologias tienden a ser sumamente complejas, esto lleva a que al hacer el traceroute se puedan presentar una serie de problemas que deben ser tenidos en cuenta.

1.2. Anomalidades en traceroutes

A continuación veremos las potenciales problemáticas de hacer un traceroute utilizando paquetes ICMP. En general las mismas surgen debido a la complejidad innata de las topologías de red. Las mismas en general se pueden agrupar en los siguientes tipos:

1. Missing hops
2. Missing destination
3. False RRTs
4. Missing links
5. Loops and Circles
6. Diamonds

2. Traceroute a Universidades

A continuacion mostramos diferentes traceroutes a universidades. El RTT promedio se calculo a partir de 5 requests. El host name lo conseguimos a partir de la funcion socket.gethostbyaddr de Python, que lo que hace es simplemente un DNS lookup. La ubicación la conseguimos a partir de una base de datos publica de GeoIP.

2.1. dc.ubar.ar

Cuadro 1: traceroute: dc.uba.ar

Hop	Avg. RTT	IP Address	Host name	Location
1	9.3842 ms	181.169.12.1	1-12-169-181.fibertel.com.ar	AR, SA
2	* * * * *			
3	* * * * *			
4	* * * * *			
5	14.025 ms	200.89.164.53	53-164-89-200.fibertel.com.ar	AR, SA
6	14.7514 ms	200.89.165.2	2-165-89-200.fibertel.com.ar	AR, SA
7	22.5916 ms	200.89.165.86	86-165-89-200.fibertel.com.ar	AR, SA
8	16.5408 ms	200.49.69.161	VPN-corp.metrored.net.ar	AR, SA
9	* * * * *			
10	* * * * *			
11	* * * * *			
12	12.7052 ms	157.92.47.53	157.92.47.53	AR, SA
13	13.067 ms	192.168.121.2	192.168.121.2	
14	* * * * *			
...	* * * * *			

Como es de esperar, al hacer un request a dc.uba.ar desde Argentina no hay saltos intercontinentales. Sin embargo, notemos que este traceroute cae en el problema de missing destination. Esto no es porque el servidor no exista, si no porque probablemente esta configurado para no devolver ICMP requests.

Intentamos acceder a metrored.net.ar, ya sea por URL o por IP y no lo logramos. Sin embargo, al hacer un IP Whois encontramos:

Cuadro 2: Whois lookup: metrored.net.ar

owner:	Techtel LMDS Comunicaciones Interactivas S.A.
ownerid:	AR-TLCI-LACNIC
responsible:	Administrador de Direcciones IP - CLARO
address:	Garay, 34
address:	C1063AB - Buenos Aires
country:	AR
phone:	+54 11 4000-3000 [3270]
nserver:	DNSMR1.METRORED.NET.AR

Por lo que podemos ver, el hop pertenece a Claro.

2.2. mit.edu

Cuadro 3: traceroute: mit.edu

Hop	Avg. RTT	IP Address	Host name	Location
1	12.6968 ms	181.169.12.1	1-12-169-181.fibertel.com.ar	AR, SA
2	* * * * *			
3	* * * * *			
4	* * * * *			
5	20.0602 ms	200.89.160.9	9-160-89-200.fibertel.com.ar	AR, SA
6	18.026 ms	200.89.165.198	198-165-89-200.fibertel.com.ar	AR, SA
7	13.8548 ms	200.89.165.86	86-165-89-200.fibertel.com.ar	AR, SA
8	13.0754 ms	195.22.220.154	xe-1-2-0.baires3.bai.seabone.net	IT, EU
9	251.8128 ms	149.3.183.73	149.3.183.73	IT, EU
10	254.8316 ms	89.221.43.107	akamai-row.londra32.lon.seabone.net	IT, EU
11	253.6456 ms	104.65.21.108	a104-65-21-108.deploy.static.akamaitechnologies.com	NL, EU

En este gráfico hay un enlace transatlántico claramente identificable entre el hop 7 y el hop 8. Notar que el host name ya indica que es transatlántico. Buscando a quien pertenece seabone.net, averiguamos que pertenece a la empresa Sparkle que provee servicios de enlaces transatlánticos.

> Sparkle is a leading global telecommunication service provider, offering a complete range of IP, Data, Cloud, Data Center, Mobile and Voice solutions designed to meet the ever changing needs of Fixed and Mobile Operators, ISPs, OTTs, Media & Content Players, Application Service Providers and Multinational Corporations (MNCs)

Curiosamente, al final el request termino en Holanda en nodo de Akamai y no en Estados Unidos. Haciendo un Whois al URL confirmamos que esto es correcto.

2.3. ox.ac.uk

Cuadro 4: traceroute: ox.ac.uk (oxford)

Hop	Avg. RTT	IP Address	Host name	Location
1	10.9412	181.169.12.1 ms	1-12-169-181.fibertel.com.ar	AR, SA
2	* * * * *			
3	* * * * *			
4	* * * * *			
5	16.9558	200.89.160.13 ms	13-160-89-200.fibertel.com.ar	AR, SA
6	15.4314	200.89.165.250 ms	250-165-89-200.fibertel.com.ar	AR, SA
7	9.7228	190.216.88.33 ms	190.216.88.33	AR, SA
8	138.7252	67.17.99.233 ms	ae0-300G.ar5.MIA1.gblx.net	US, NA
9	* * * * *			
10	* * *			
10	224.1195	4.69.143.190 ms	ae-1-3104.ear2.London2.Level3.net	US, NA
11	224.8286	212.187.139.166 ms	unknown.Level3.net	GB, EU
12	236.9458	146.97.33.2 ms	ae29.londpg-sbr2.ja.net	GB, EU
13	240.9694	146.97.37.194 ms	ae19.readdy-rbr1.ja.net	GB, EU
14	227.1278	193.63.108.94 ms	ae2.oxfoii-rbr1.ja.net	GB, EU
15	227.3266	193.63.108.98 ms	ae3.oxforq-rbr1.ja.net	GB, EU
16	228.0936	193.63.109.90 ms	193.63.109.90	GB, EU
17	* * * * *			
18	* * * * *			
19	239.6874	192.76.32.62 ms	boucs-lompi1.sdc.ox.ac.uk	GB, EU
20	225.6974	129.67.242.154 ms	aurochs-web-154.nsms.ox.ac.uk	GB, EU

Aquí podemos identificar claramente enlaces transatlánticos a partir del host name y la ubicación. Level3 es una empresa conocida proveedora de enlaces. Un dato de color, sus acciones cotizan en Nasdaq.

> Level 3 Communications, Inc. is a premier provider of global communication services, creating solutions that strengthen the growth, efficiency and security of businesses around the world. Our business started as part of a subsidiary of a construction company that created one of the first competitive local exchange carriers, MFS Communications.

2.4. u-tokyo.ac.jp

Cuadro 5: traceroute: u-tokyo.ac.jp

Hop	Avg. RTT	IP Address	Host name	Location
1	9.9508 ms	181.169.12.1	1-12-169-181.fibertel.com.ar	AR, SA
2	* * * * *			
3	* * * * *			
4	* * * * *			
5	16.979 ms	200.89.160.21	21-160-89-200.fibertel.com.ar	AR, SA
6	15.2796 ms	200.89.165.222	222-165-89-200.fibertel.com.ar	AR, SA
7	10.541 ms	195.22.220.102	xe-1-0-3.baires5.bai.seabone.net	IT, EU
8	39.8348 ms	195.22.219.17	ae7.sanpaolo8.spa.seabone.net	IT, EU
9	36.1798 ms	195.22.219.17	ae7.sanpaolo8.spa.seabone.net	IT, EU
10	42.7854 ms	149.3.181.65	149.3.181.65	IT, EU
11	159.2136 ms	129.250.2.227	ae-4.r24.nycmny01.us.bb.gin.ntt.net	US, NA
12	237.3446 ms	129.250.4.13	ae-2.r20.sttlwa01.us.bb.gin.ntt.net	US, NA
13	225.4494 ms	129.250.2.54	ae-0.r21.sttlwa01.us.bb.gin.ntt.net	US, NA
14	426.808 ms	129.250.3.86	ae-2.r20.osakjp02.jp.bb.gin.ntt.net	US, NA
15	429.0596 ms	129.250.6.188	ae-4.r22.osakjp02.jp.bb.gin.ntt.net	US, NA
16	421.2708 ms	129.250.2.255	ae-1.r01.osakjp02.jp.bb.gin.ntt.net	US, NA
17	417.919 ms	61.200.80.218	xe-0-4-0-7.r01.osakjp02.jp.ce.gin.ntt.net	JP, AS
18	425.9262 ms	158.205.192.173	ae0.ostcr01.idc.jp	JP, AS
19	426.6464 ms	158.205.192.86	158.205.192.86	JP, AS
20	534.723 ms	158.205.121.250	po2.l321.fk1.eg.idc.jp	JP, AS
21	436.512 ms	154.34.240.254	154.34.240.254	JP, AS
22	424.7352 ms	210.152.135.178	210.152.135.178	JP, AS

Como era de esperar, este termino siendo el traceroute mas largo, pasando por un camino sumamente raro. De Argentina a Italia, luego a EE.UU. y finalmente a Japon. Encontramos nuevamente los enlaces transatlánticos de Sparkle. Entrando a ntt.net nos encontramos con:

Esto nos hace inferir que es una empresa Japonesa de telecomunicaciones.

3. Experimentos

3.1. Caching

Hop	Avg. RTT	IP Address	Host name	Location
1	10.6688	181.169.12.1 ms	1-12-169-181.fibertel.com.ar	AR, SA
2	* * * * *			
3	* * * * *			
4	* * * * *			
5	20.2096	200.89.160.21 ms	21-160-89-200.fibertel.com.ar	AR, SA
6	14.3278	200.89.165.129 ms	129-165-89-200.fibertel.com.ar	AR, SA
7	12.5566	200.89.165.150 ms	150-165-89-200.fibertel.com.ar	AR, SA
8	* * * * *			
9	10.9052	209.85.251.86 ms	209.85.251.86	US, NA
10	40.759	209.85.252.42 ms	209.85.252.42	US, NA
11	38.5816	216.239.58.221 ms	216.239.58.221	US, NA
12	38.1802	216.58.202.4 ms	gru06s26-in-f4.1e100.net	US, NA

Cuadro 6: traceroute: google.com sin caching

Hop	Avg. RTT	IP Address	Host name	Location
1	11.1854	181.169.12.1 ms	1-12-169-181.fibertel.com.ar	AR, SA
2	* * * * *			
3	* * * * *			
4	* * * * *			
5	21.9184	200.89.165.33 ms	33-165-89-200.fibertel.com.ar	AR, SA
6	15.066	200.89.164.26 ms	26-164-89-200.fibertel.com.ar	AR, SA
7	* * * * *			
8	11.6574	181.30.241.187 ms	187-241-30-181.fibertel.com.ar	AR, SA

Cuadro 7: traceroute: google.com con caching

3.2. Detección de links intercontinentales

1. Falsos Positivos / Falsos Negativos
Intercontinental Local Test Intercontinental Test Local
Muestra: 100 sitios de alexa?
Hacer funcion que detecte enlaces intercontinentales con libreria de Python.

3.3. Traceroute anomalities

4. Conclusión

Discutir alternativas, onda hacer esto por IP.
Charlar sobre el uso de embebidos para network topology (discutir challenges de topology)
cerrar con ideas, estadisticas e imagenes de aca? http://internetcensus2012.bitbucket.org/paper.html

Referencias

[1] Carna Botnet. Internet census 2012: Port scanning using insecure embedded devices. http://internetcensus2012.bitbucket.org/paper.html, 2013.

[2] John M Cimbala. Outliers. http://www.mne.psu.edu/cimbala/me345/Lectures/Outliers.pdf, 2011.

[3] RFC 792 (ICMP). http://www.ietf.org/rfc/rfc792.txt.

[4] Martin Erich Jobst. Traceroute anomalies. http://www.net.in.tum.de/fileadmin/TUM/NET/NET-2012-08-1/NET-2012-08-1_02.pdf, 2012.

[5] Traceroute (Wikipedia). http://en.wikipedia.org/wiki/Traceroute.