



udp UNIVERSIDAD
DIEGO PORTALES

Facultad de Ingeniería
Escuela de Informática y Telecomunicaciones

Informe de laboratorio

Alumnos: Thomas Gonzáles
Michiru Nakamura
Flavio Pallini
Fernando Peña

Profesor: Jaime Álvarez
Ayudante: Maximiliano Vega
fernando.Penas@mail.udp.cl
thomas.gonzalez@mail.udp.cl
michiru.nakamura@mail.udp.cl
Flavio.pallini@mail.udp.cl

24 de Marzo de 2016

Índice general

1. Introducción	2
1.1. Preguntas del laboratorio	2
2. Paquete de datos	4
2.1. Componentes	4
3. Los destinos de un paquete	6
3.1. Envío de paquete a todos los equipos de la red	6
3.2. Envío de paquete a un equipo específico	7
3.3. Envío de paquete a una dirección que no está en la red	8
4. Conclusión	9

1. Introducción

Scapy es un programa que permite al usuario manipular paquetes, con el se pueden crear, configurar, enviar, capturar y analizar paquetes libremente de manera interactiva utilizando código escrito en Python. También es posible estructurar un paquete en base a un protocolo por capas. En este laboratorio se usa esta última función para crear paquetes por niveles basándose en el modelo OSI, los niveles fueron configurados de manera que contuvieran la dirección MAC e IP de tanto la fuente como el destino del paquete, y con la ayuda de Wireshark, programa que permite analizar paquetes recibidos con una interfaz grafica, se verificó como distintos valores de las direcciones afectan el envío del paquete.

1.1. Preguntas del laboratorio

1. ¿Qué pasa cuando envío un paquete a la dirección FF:FF:FF:FF:FF:FF?
¿Quienes lo reciben? ¿Por qué?

r: Envía el paquete por medio broadcast a todos los usuarios conectados a la red. Esto sucede porque al no tener la dirección mac de algún hardware en específico, esta pregunta en la red por las direcciones ip relacionadas con las MAC de cada ordenador.

2. ¿Qué pasa cuando envío un paquete a una MAC de otro equipo?
¿Quienes lo pueden recibir? ¿Por qué?

r:El paquete llega al equipo con la MAC señalada y a ningún otro equipo más que el señalado. Esto se debe a que cada equipo tiene una

dirección MAC única.

3. ¿Qué sucede si envía un paquete a una MAC que no corresponda a ningún equipo de la red? ¿Quiénes lo pueden recepcionar? ¿Por qué?

r: No llega a nadie que está dentro de la red, se lo puede enviar a algún equipo que se pueda comunicar por fuera de la red, equipos que puedan ser capaces de comunicarse de esta manera son los módems y los routers. Alternativamente, al no poder accederse a módems y routers, el paquete va a ser enviado por broadcast y recibido por todos los equipos en la red.

2. Paquete de datos

2.1. Componentes

Un paquete típico va a ser armado siguiendo el modelo OSI, por lo que termina obligatoriamente en la capa física, hacia atrás. Luego, dependiendo de la finalidad del paquete, va a contener más o menos elementos de diferentes capas de OSI.

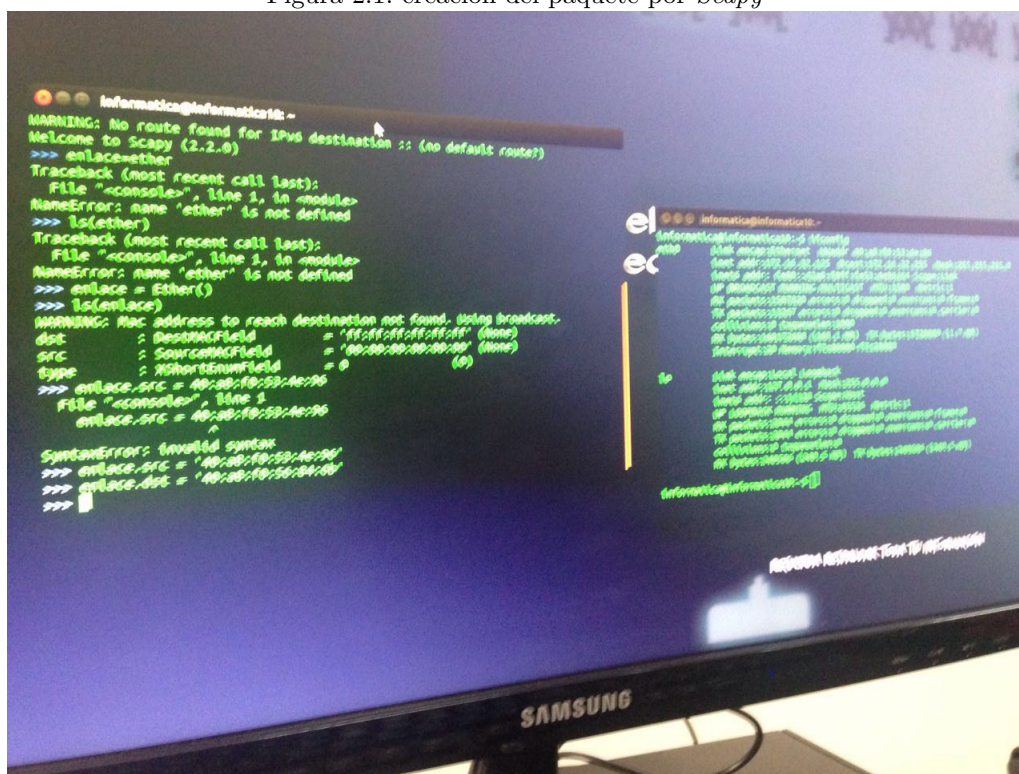
- Capas de aplicación, presentación y sesión: Se refieren a la parte más orientada al usuario de cualquier proceso de comunicación entre dispositivos. Entre estas capas, se genera un flujo (o *stream*) de datos a ser enviado al otro equipo, usualmente segmentado. En estos niveles, al *stream* no se le ha hecho ninguna modificación más allá de la fragmentación.
- Capa de transporte: En esta capa, inicia el proceso de encapsulamiento del *stream* a un paquete de datos, proceso que verifica que estos se envíen perfectamente al equipo correcto. Se añaden bytes que refieran a la verificación de envío correcto, el tiempo de demora del envío, si es que fue necesario retransmitir, etc.; en general, asegurando que el envío haya sido perfecto.
- Capa de red: En este nivel, continúa el encapsulamiento, insertando el paquete en una trama. Se añaden bytes con las direcciones IP tanto del emisor como del receptor (o, en caso de envío por *Broadcast*, la dirección IP saturada: 255.255.255.255)
- Capa de enlace: En este nivel, termina el proceso de encapsula-

miento de datos. Se agregan un número de bytes con las direcciones MAC tanto del emisor como del receptor (o, en caso de envío por *Broadcast*, la dirección MAC saturada: FF:FF:FF:FF:FF:FF)

- Capa física: En este nivel, se convierte la trama primero en información en bits para ser enviados por el medio de transporte.

Una de las ventajas que tiene *Scapy* al momento de estudiarse el envío de paquetes de datos es que permite rellenar sólo la información más relevante al momento de determinar a qué equipo enviar información. Así, de poseer una dirección MAC o IP de algún equipo en particular, basta con una de ellas para enviar el *stream* de datos a éste, prescindiendo del resto u rellenando a medida que se envía.

Figura 2.1: creación del paquete por *Scapy*



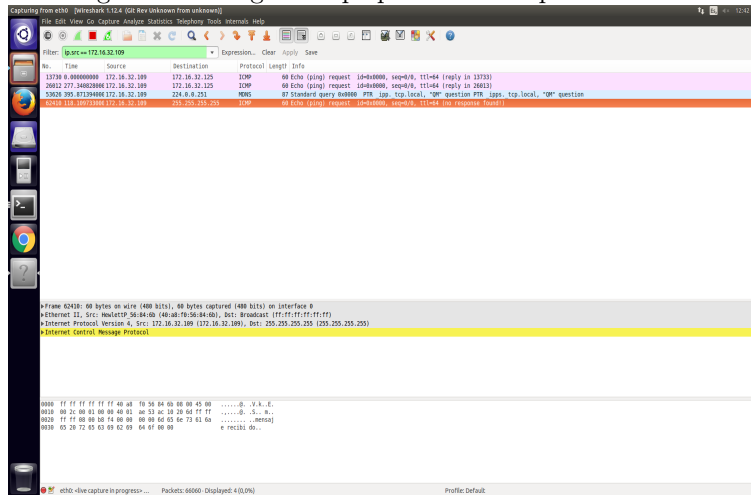
3. Los destinos de un paquete

El destino de un paquete a enviar viene definido en el mismo. Gracias al modelo OSI, es posible crear paquetes similares pero con distinto destino modificando las capas de transporte y/o red. A continuación se describirá que ocurre en los distintos casos posibles para el envío de un paquete.

3.1. Envío de paquete a todos los equipos de la red

Una dirección MAC definida como FF:FF:FF:FF:FF:FF se ve un poco extraña, ya que claramente esta dirección **no** pertenece a ningún equipo en específico, más bien pertenece a **todos los equipos conectados a la red**. Cuando se envía un paquete con ésta MAC de destino, el paquete llega a todos los equipos conectados a la red, pues cada uno de ellos sabe que la dirección FF:FF:FF:FF:FF:FF corresponde a todos los equipos conectados (dicho de otro modo, se realiza una transferencia de tipo *broadcast*). También, es posible lograr el mismo resultado utilizando la dirección IP asociada a un envío broadcast; en el caso de nuestra red: 168.192.1.**255**. Así, es posible visualizar un paquete enviado con destino a ésta MAC desde cualquier equipo que sea parte de la red.

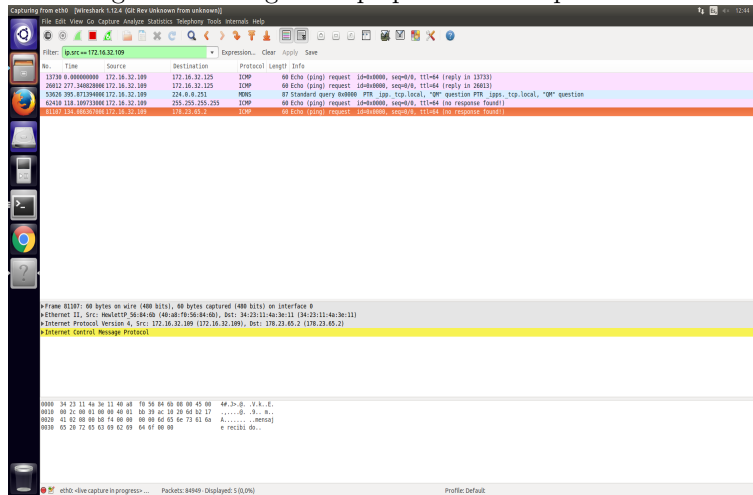
Figura 3.1: imagen del paquete enviado por broadcast



3.2. Envío de paquete a un equipo específico

Para enviar un paquete a un equipo en particular, es necesario conocer la dirección MAC, o bien la dirección IP de éste. Cuando se envía un paquete de este tipo, llegará solo al equipo que contenga la dirección MAC y/o IP de destino escrita en el paquete. Ningun otro equipo debería ser capaz de visualizar el paquete, puesto que solo debe haber un equipo con la dirección MAC buscada. Cabe mencionar que en el caso de una red que utiliza los dispositivos descontinuados 'hub' era posible visualizar cualquier paquete transmitido a través de la red, puesto que el hub direcciona los paquetes a todos los equipos de la red sin importar su destino. No así con los switch, sin embargo éste ultimo se puede configurar para que actúe de forma broadcast, haciendo posible que todos los equipos de la red puedan visualizar paquetes no correspondidos (éste es el caso de nuestra red).

Figura 3.2: imagen del paquete enviado por unicast



3.3. Envío de paquete a una dirección que no está en la red

Quando se envía un paquete cuya dirección de destino no corresponde a ningún equipo de la red, éste no llegará a ningún equipo puesto que no existe un equipo que reciba el mensaje. Cabe destacar que en nuestra red si es posible visualizar paquetes con destino inexistente, pues nuestra red es del tipo broadcast como se menciona en la sección 3.1.

4. Conclusión

La experiencia de este experimento nos permite entender como los equipos se comunican entre si, que la creacion de paquetes de red vienen con una estructura definida por capas. Se vio que los paquetes enviados por broadcast y unicast necesitan de una estructura similar, pero dentro de sus capas algunos de los datos que poseen son los que varían (como la dirección MAC de destino). Además, se logró visualizar con mayor detalle acerca de cómo está contruido un paquete de datos, cómo estos se van armando hasta que se envían a su destino. También se lograron resultados positivos en cuanto a la transferencia de paquetes dentro de nuestra red, la operación de enviar un paquete desde un ordenador y poder recibirlo desde otro ordenador fue exitosa. Se concluye el modelo OSI utilizado es un modelo óptimo y claro para poder definir la estructura de un paquete y hacer que sus componentes sean de fácil manipulación.

Índice de figuras

2.1. creación del paquete por *Scapy* 5

3.1. imagen del paquete enviado por broadcast 7

3.2. imagen del paquete enviado por unicast 8