Protocolo ICMP y ARP java

David Steven López Tovar, Edwin Alejandro Turizo Prieto, Gabriel Alejandro Terán Guerrero
Departamento de Ingeniería de Sistemas, Pontificia Universidad Javeriana, Bogotá, Colombia.

dalopez@javeriana.edu.co
edwin.turizo@javeriana.edu.co
teran.g@javeriana.edu.co

I. INTRODUCCIÓN

El presente informe, pretende dar documentación acerca de una aplicación desarrollada en el ámbito de las redes y comunicación de los computadores. Dicha aplicación, consiste en permitir el envío de tramas ethernet que incluyan el uso de protocolos ICMP (*Internet Control Message Protocol) y* ARP *(Address Resolution Protocol)* en el caso de Direccionamiento IPV4.

II. CONTENIDO TEÓRICO

A. ARP

Es el Protocolo de Resolución de Direcciones, que trabaja en la capa de Enlace del modelo OSI (Open System Interconnection). Su principal objetivo, es conocer la dirección física (MAC) del receptor de una tarjeta de interfaz de red correspondiente a una dirección IP (Internet Protocol).

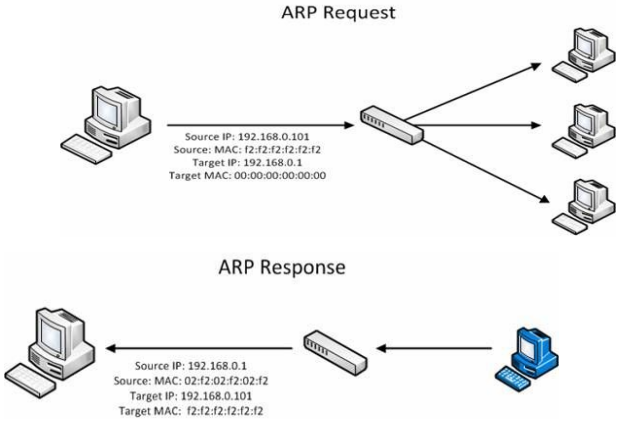

Figura 1:Esquema ARP[3]

Para que las direcciones físicas se puedan conectar con las direcciones lógicas, el protocolo ARP interroga a los equipos de la red para averiguar sus direcciones físicas y luego crea una tabla de búsqueda entre las direcciones lógicas y físicas en una memoria caché.

Dirección IP	Dir. de red
202.2.3.4	ee.ee.ee.ee.ee.ee
202.2.3.3	cc.cc.cc.cc.cc.cc
202.2.3.1	xx.xx.xx.xx.xx.xx

Figura 2: Tabla ARP [4]

La cabecera de ARP, comienza con la información de dos bytes de longitud sobre el tipo de dirección de hardware. A continuación, se indica el tipo protocolo (16 bits también), que en las direcciones IPv4 se distinguen por el valor *0x0800.* Los dos campos siguientes, informan sobre la longitud de las las direcciones MAC con un tamaño de seis bytes y en cambio, las direcciones IP con con un tamaño de cuatro bytes. Los siguientes dos bytes, son del tipo de operación: El *valor 1* se utiliza para una solicitud ARP y el dos revela que se trata de una respuesta ARP. Por último, los paquetes reciben las cuatro direcciones relevantes y previamente declaradas: La dirección MAC del transmisor, la dirección IP del transmisor, la dirección MAC de receptor, y la dirección IP del receptor.

Tipo Hardw	Tipo Protoc.	Tam. Hard.	Tam. Protoc.	Tipo Operac	MAC Origen	IP Origen	MAC Dest.	IP Dest.
2	2	1	1	2	6	4	6	4

Figura 3: Trama ARP[5]

B. ICMP

Es el Protocolo de Control de Mensajes de Internet. ICMP, proporciona así una comunicación entre el software IP de una máquina y el mismo software en otra. Se utiliza para manejar mensajes de error y de control necesarios para los sistemas

de la red, informando con ellos a la fuente original para que evite o corrija el problema detectado.

A continuación, se muestra una imagen donde se explica el procedimiento de envio y recibimiento de un ping, en el cual el cliente ejecuta el protocolo ICMP echo request obteniendo respuesta.

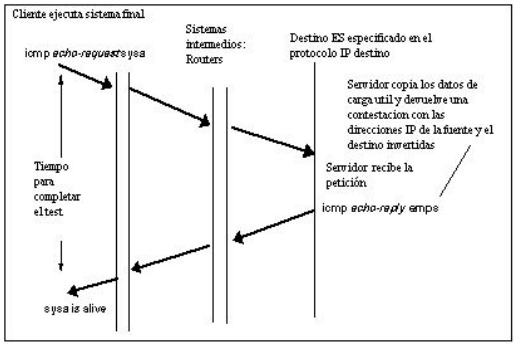

Figura 4: Procedimiento de envio y recibimiento de un ping.[6]

El mensaje ICMP, comienza con un tipo de ocho bits el cual nos especifica que tipo de ICMP es. Luego, está el código de ocho bits que nos referencia la situación de la trama, un ejemplo de esta situación puede ser que ha logrado llegar de manera correcta, la dirección de host está prohibida por el administrador, que la ruta fuente ha fallado, entre otros. Posteriormente, esta el checksum de 16 bits el cual se usa como mecanismo para detectar errores de transmisión de datos. Por último, se tiene el mensaje que es de tamaño variable.

Título	Mensaje ICMP			
	Tipo (8 bits)	Código (8 bits)	Checksum (16 bits)	Mensaje (tamaño variable)

Figura 5: Un mensaje ICMP encapsulado en un datagrama IP [8]

III. DESARROLLO DEL PROYECTO

Para el desarrollo y la ejecución del proyecto, se utilizó la librería Pcap4J la cual a partir de una serie de funciones pertenecientes a la librería, se logra crear y enviar tramas ethernet, en nuestro caso, de un computador a otro.

Para mayor información sobre la descripción en detalle de la librería y su implementación. *ver [9] y [10]*

Para la creación del protocolo ICMP, se utilizó de manera completa la librería.

Para establecer el protocolo ARP, se utilizó la librería mencionada, pero tan solo para el envio de la trama ya que la creación de la trama se hizo de forma manual y sin ninguna libreria.

Clases del proyecto

A. ARP:

Atributos:

Dentro de los atributos de la clase ARP ,encontramos que todos son de tipo byte, la única diferencia es que algunas variables son contenedores y otros no. Los atributos son privados.

- tipoH: Contenedor. Guarda el tipo de hardware.
- tipoP: Contenedor. Guarda tipo de protocolo.
- longitudH: longitud de la dirección de hardware (MAC).
- longitudP: longitud de las direcciones utilizadas en el protocolo de capa superior, como es ipv4.
- operacion: Contenedor. Código de operación, determina si se va a hacer un request o replay
- macO: contenedor. Dirección mac origen.
- ipO: contenedor. Dirección ip origen.
- macD: contenedor. Dirección mac destino.
- ipD: contenedor. Dirección ip destino.
- tramaARP: contenedor. Trama Ethernet cuyo paylouder es una trama ARP.

Métodos:

- ARP. Parámetros de entrada: la dirección ip de origen, la dirección MAC de origen y la dirección ip de destino. Creador de la clase ARP.
- creadorTrama. Parámetros de entrada: ninguno. Salida: la trama ARP completa.

B. ICMP:

Atributos:

En esta clase se ve como el ICMP se conforma y como se crea, para ello se usó las cuatro primeras variables que ayudan al momento de abrir la interfaz.

- READ_TIMEOUT_KEY: Cadena de caracteres.
- READ_TIMEOUT: Número natural.
- SNAPLEN_KEY: Cadena de caracteres.
- SNAPLEN: Número natural.
- ip0: Cadena de caracteres. Dirección ip de origen.
- mac0: Cadena de caracteres. Dirección MAC de origen.
- ipD: Cadena de caracteres. Dirección ip de destino.
- macD: Cadena de caracteres. Dirección de destino MAC destino.
- TipoIcmp: Entero. Indica el Tipo de ICMP.
- Codigo: Entero. Indica el código del ICMP.
- TTL: Entero. Indica el Time to Live de la trama Ethernet.
- Datos: Cadena de Caracteres. Datos del ICMP.

Métodos:

- ICMP: Parámetros de entrada: Dirección ip de origen, dirección ip de destino, Dirección MAC de origen, dirección de destino MAC destino, tipo de ICMP, código del ICMP, TTL y Datos del ICMP. Creador de la clase ICMP.
- creadorICMP: Parámetros de entrada: ninguno. Salida: el paquete con la trama ethernet.

C. Proyecto_Vista1

Atributos:

En esta clase, se generan las distintas ventanas de la interfaz gráfica, a su vez, posee un conjunto de variables especiales para la ayuda de iniciar la interfaz.

- READ_TIMEOUT_KEY: Cadena de caracteres.
- READ_TIMEOUT: Número natural.
- SNAPLEN_KEY: Cadena de caracteres.
- SNAPLEN: Número natural.
- interfaz: Atributo público de tipo PcapNetworkInterface perteneciente a la clase Pcap4J.
- IPo: Atributo público de tipo InetAddress perteneciente a la clase Pcap4J.
- i: Atributo público de tipo interfaces.

Métodos:

Los métodos pertenecientes a esta clase, hacen parte de la interfaz gráfica, como los son los botones y los cuadros de texto. También, se encuentra el main en el cual el programa comienza a correr.

D. Interfaces:

Atributos:

- p: contenedor de PcapNetworkInterface. usado para guardarlas y luego mostrarlas en la interfaz.

Métodos:

- Interfaces: Parámetros de entrada: ninguno. Cuenta con tan solo un único método y es su constructor.

Diagrama de clases:

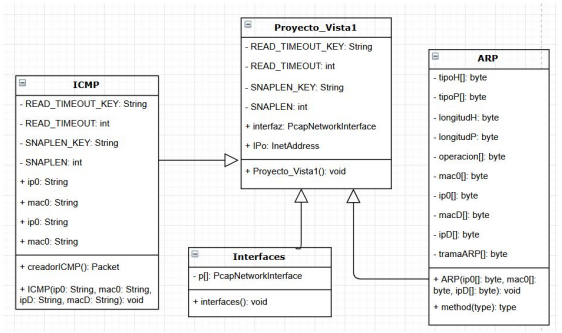

Figura 6: Diagrama de clases proyecto uno de redes

IV. IMPLEMENTACIÓN DEL PROYECTO

A continuación se explicará el funcionamiento del programa teniendo como ayuda toma de pantalla a los computadores tanto receptores como transmisores.

Antes de comenzar el programa, verificamos las direcciones ip y direcciones MAC de ambas computadoras.

Figura 7: dirección ip y MAc de computador receptor.

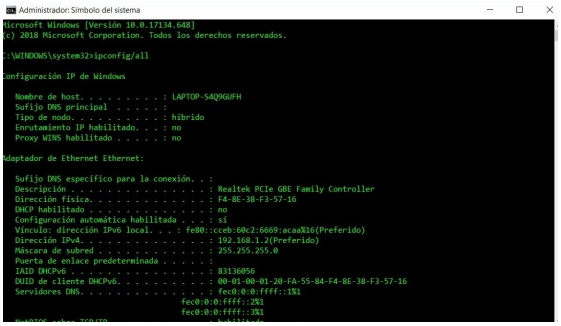

Figura 8: dirección ip y MAc de computador transmisor.

A continuación, se pone en ejecución el programa hecho en el lenguaje Java como primera interacción de interfaz obtenemos lo siguiente.

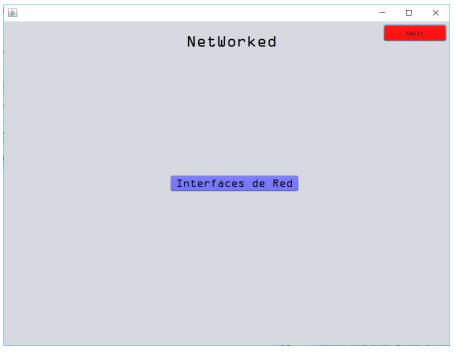

Figura 9: primera interacción de interfaz.

En el próximo paso se escoge la interfaz de red.

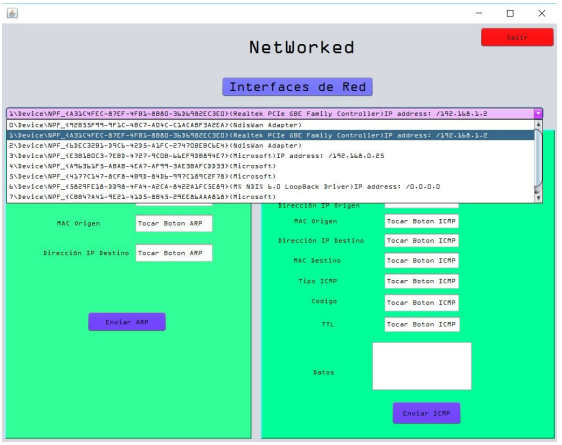

Figura 10: selección de la interfaz de red.

Luego de haber seleccionado la interfaz de red deseada, entraremos a la segunda interacción con la interfaz gráfica.

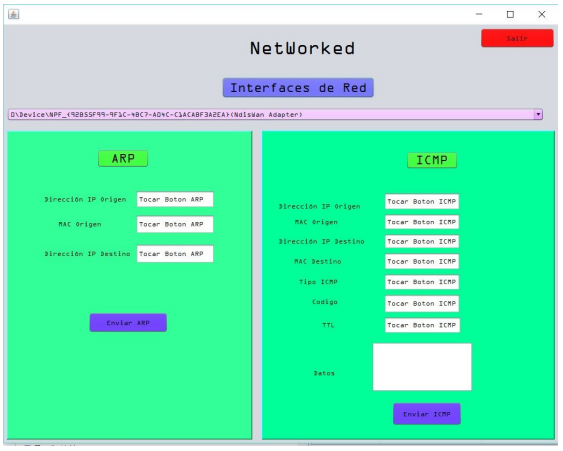

Figura 11: segunda interacción con interfaz.

En este momento, se selecciona el botón que dice ARP, para así crear la trama ARP.

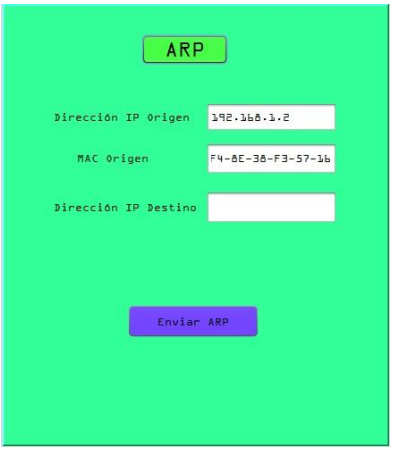

Figura 12: dirección ip y MAC listas para envio.

Ahora, manualmente se agrega la dirección IP de destino que en este caso es de el computador receptor 192.168.1.1.

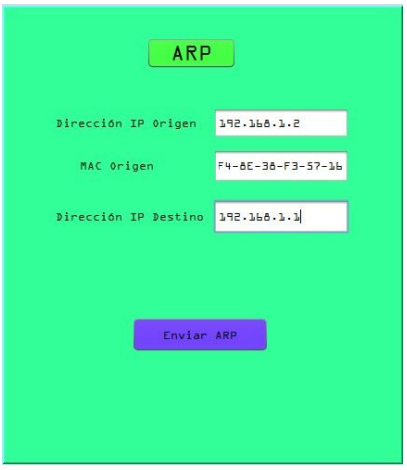

Figura 13: ingresa dirección ip receptor.

A continuación, se abre wireshark *ver[11]* en el computador receptor, seleccionando la conexión de red deseada, en este caso la conexión ethernet, ya que ambos computadores se encuentran conectados mediante un cable UTP con conectores RJ45.

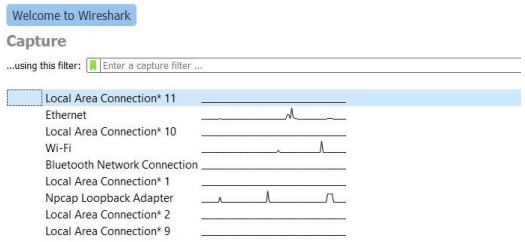

Figura 14: interfaz Wireshark.

Luego de haber tenido la interacción anterior, se acciona el botón de "Enviar ARP". En la siguiente imagen, se ve la llegada y respuesta por parte del receptor respecto a esta trama.

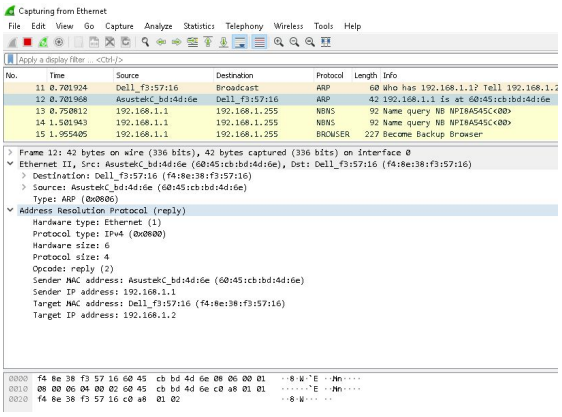

Figura 15: recepción de ARP.

Luego de haber confirmado que la trama anterior haya llegado de manera correcta, proseguimos a ejecutar el siguiente protocolo ICMP, seleccionando el botón indicado con la palabra "ICMP", en el que se obtiene la dirección ip y MAC de origen.

Figura 16: dirección y MAC lista para envio.

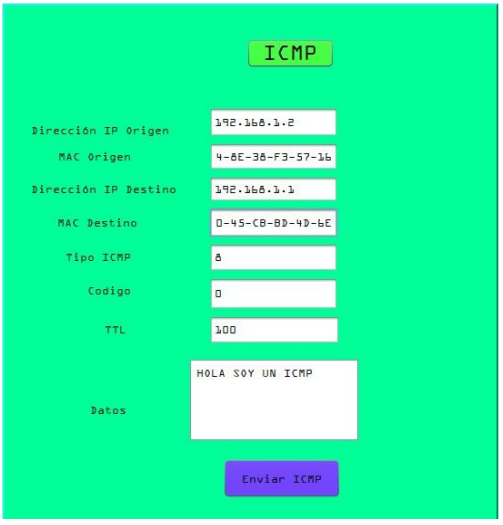

Figura 17: campos listos para envio.

Ahora, se llenan los campos faltantes, en este caso la dirección ip y MAC de destino. Luego de haber llenado estos campos, se prosigue a la ejecución del programa enviando la trama con el botón de "Enviar ICMP". Dentro del campo "Datos", podemos seleccionar el mensaje que deseamos enviar a la computadora receptora, en este caso se envía "HOLA SOY UN ICMP".

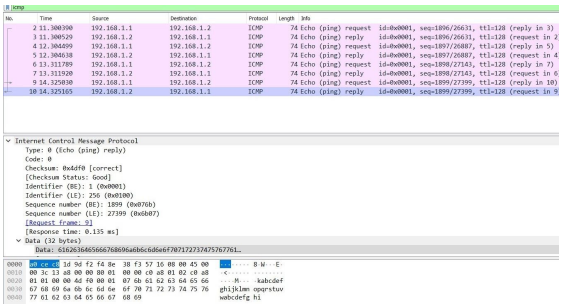

FIgura 18: recepción ICMP.

Como se puede ver en la figura 18, se ve la trama ICMP y más abajo se puede conocer que el mensaje que está en la trama, es efectivamente el enviado por el computador transmisor.

V. FUNCIONAMIENTO PROTOCOLOS ARP E ICMP EN EL PROYECTO

A. Protocolo ARP

EL protocolo ARP en el desarrollo del proyecto, se genera desde su constructor, donde cada atributo adquiere un valor, a su vez, desde su segundo método de nombre creadorTrama, se concatena la trama ethernet con la ARP, para su envío.

Reanudando desde la clase Proyecto_Vista1, se puede ver que existe un método llamado "jButton4ActionPerformed" donde se tiene la dirección ip y la dirección MAC de la computadora transmisora. Dentro de este método, existe una seria de try-catch en los cuales algunos atrapan el valor que existe en los cuadros de texto y los convierte en una ip. En el otro caso, agarra el valor de texto que existe en esta casilla y lo convierte en una MAC.

Luego de haber definido los siguientes datos mediante los try-catch; dirección ip origen, dirección MAC origen y dirección ip destino en ese debido orden, se puede instanciar el ARP llamando el constructor con los parámetros mencionados. Luego, se crea una variable de tipo PcapHandle, pero para ello, antes se necesita llamar la función de la interfaz "openlife", en donde a dentro de este método, van los parámetros que se mencionan al comienzo del documento, en donde se describen los atributos los cuales van sujetos a la librería.

Al final, lo que el sistema realiza, es enviar el paquete ARP.creadorTrama que es donde recibimos la trama completa en bytes y al final se cierra la interfaz.

B. Protocolo ICMP

Al comienzo, tenemos los atributos del ICMP, donde en su constructor, se les da determinados valores obtenidos a partir del cuadro de texto que se tiene en la interfaz.

En la función creadora ICMP, se crea el paquete ICMP, luego se crea el paquete IPv4 para dentro de ese introducir el paquete ICMP. Por último, se crea el paquete ethernet, donde el paquete IPv4 se inserta en el previamente mencionada. Al final, se le envía al Packet llamado con el nombre de "p" en el script.

Lo siguiente, es que a partir de la función "jButton5ActionPerformed" ubicada en la clase Proyecto_Vista1, inicia de igual modo que lo hace con la ARP, es decir, obtiene los valores de los cuadros de texto, crea el ICMP, lo envía y luego cierra la interfaz.

REFERENCIAS

[1]***Address Resolution Protocol (ARP).*** microchip [online]. Tomado de: http://microchipdeveloper.com/tcpip:arp#arp_request

[2]***Capturar paquetes ipv4/ipv6 en java.*** programacion.net [online]. Tomado de: https://programacion.net/articulo/capturar_paquetes_ipv4_ipv6_en_java_1233

[3]*Definición ARP.* Sistemas master [online]. Tomado de: https://sistemas.com/arp.php ,.....

[4]*Comunicación de datos I.* Slide player. [online]. Tomado de: https://slideplayer.es/slide/4159445/

[5]*ARP, RARP,* slideshare [online]. Tomado de: https://es.slideshare.net/visualnet_20_735/arprarp

[6]*Internet control message protocol.* [online]. Tomado de: https://erg.abdn.ac.uk/users/gorry/eg3567/inet-pages/icmp.html

[7] *Java library for capturing, crafting, and sending packets.* Pcap4J [online] Tomado de: http://www.javadoc.io/doc/org.pcap4j/pcap4j/1.7.3

[8] *Protocolo ICMP.* CCM. [online]. Tomado de: https://es.ccm.net/contents/265-el-protocolo-icmp

[9]*A Java library for capturing, crafting, and sending packets.* Pcap4J [online]. Tomado de: https://www.pcap4j.org/

[10] *Pcap4J.* Github [online]. Tomado de: https://github.com/kaitoy/pcap4j

[11]*About wireshark.* Wireshar [online]. Tomado de: https://www.wireshark.org/