

INFORME PRÁCTICA 2

Alejandro Santorum Varela - alejandro.santorum@estudiante.uam.es

David Cabornero Pascual - david.cabornero@estudiante.uam.es

Redes de Comunicaciones I - Práctica 2 - Pareja 4

Universidad Autónoma de Madrid

10-11-2018

Contents

1	Introducción	2
2	Estructura del programa	2
3	Ejemplos de salidas	3
4	Decisiones de diseño	10
5	Conclusión	11

1 Introducción

Este documento consiste en el informe de la práctica 2 de Redes de Comunicaciones I. Se recoge el diseño, estructura y salidas del código pedido para esta práctica, que consiste en introducirnos en el *parseo* de las cabeceras de nivel 2, 3 y 4, además de aplicar filtros sobre ciertos campos de las cabeceras.

2 Estructura del programa

En primer lugar vamos a comentar un poco el programa implementado.

Primero, en el ejemplo colgado en Moodle se incluía una muestra de como *parsear* argumentos de entrada en C, pero debido a que no comprendíamos al 100% el funcionamiento hemos decidido desarrollar nuestra propia función de comprobación de parámetros de entrada: **void input_parameter_checking(int, char**)**. En esta función nos aseguramos que los parámetros de entrada se han introducido correctamente: que interfaz y fichero .pcap sean excluyentes, que no haya indicadores (-ipo, -ipd, -po, ...) desconocidos, que los filtros introducidos sean correctos (números enteros, quizá separados por "." en el caso de las direcciones IP), etc. Además nos ayudamos de unas variables globales para saber en todo momento que filtros han sido activados.

Continuando con el programa, abrimos una captura en vivo si ha sido introducida una interfaz o, por el contrario, abrimos una captura *offline* en el caso que haya sido aportado el nombre de un fichero .pcap.

Es ahora cuando llamamos a pcap_loop(...) para analizar cada paquete. En la función **void package_treat(...)** es donde examinamos las cabeceras de los distintos niveles de cada paquete.

Primero, imprimimos por pantalla el número de paquete que se está analizando y el momento de llegada. A continuación analizamos el nivel de enlace (nivel 2), imprimiendo en hexadecimal la dirección Ethernet destino, la dirección Ethernet origen y el tipo Ethernet, donde nos aseguramos que el siguiente protocolo sea IPv4, de lo contrario no continuaríamos analizando el paquete en los siguientes niveles.

En el caso de que efectivamente el siguiente protocolo sea IPv4, analizamos el nivel de red (nivel 3), mostrando por pantalla en decimal la versión (si está bien hecho el programa debería ser el código en decimal de la versión 4 del protocolo IP), la longitud de la cabecera, la longitud total, el desplazamiento, el tiempo de vida, el protocolo que encapsula, la dirección IP origen y la dirección IP destino. Por último en este nivel, comprobamos que el desplazamiento sea igual a cero, de lo contrario no se analizaría el nivel 4 porque el paquete está fragmentado y no posee la cabecera de nivel 4; y también comprobaríamos que el protocolo que encapsula es UDP o TCP. En el caso de que el protocolo no sea ninguno de los anteriores, tampoco se analizaría el nivel 4 porque no sería uno de los protocolos esperados.

En el caso de que el protocolo de nivel de transporte (nivel 4) sea UDP o TCP, se procederá a su análisis, imprimiendo por pantalla en decimal el puerto origen, el puerto destino, el campo longitud en el caso de UDP y las banderas SYN y FIN en el caso de TCP.

Finaliza el programa con control+C o porque se ha finalizado de leer todo el fichero .pcap,

mostrando el número de paquete analizados.

3 Ejemplos de salidas

Empezaremos mostrando salidas para un ejemplo de lectura de traza (en este caso del fichero fragipv4udp.pcap aportado). Como la lectura de los diferentes campos de las cabeceras es igual para lectura desde fichero y para interfaz, podemos ejemplificar su correcto funcionamiento con estos pantallazos.

Número de paquete

▼ **Frame 1** 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits)

Encapsulation type: Ethernet (1)
Arrival Time: Sep 12, 2013 03:47:28.000000000 UTC
[Time shift for this packet: 0.000000000 seconds]
Epoch Time: 1378957648.000000000 seconds
[Time delta from previous captured frame: 0.000000000 seconds]
[Time delta from previous displayed frame: 0.000000000 seconds]
[Time since reference or first frame: 0.000000000 seconds]
Frame Number: 1
Frame Length: 1514 bytes (12112 bits)
Capture Length: 1514 bytes (12112 bits)
[Frame is marked: False]
[Frame is ignored: False]
[Protocols in frame: eth:ip:data]

▼ **Ethernet II**, Src: 14:dd:a9:d2:ef:57 (14:dd:a9:d2:ef:57), Dst: 00:00:00:00:00:00 (00:00:00:00:00:00)

► Destination: 00:00:00:00:00:00 (00:00:00:00:00:00) **Dirección Ethernet Destino**
► Source: 14:dd:a9:d2:ef:57 (14:dd:a9:d2:ef:57) **Dirección Ethernet Origen**
Type: IP (0x0800)

▼ **Internet Protocol Version 4**, Src: 150.244.58.114 (150.244.58.114), Dst: 8.8.8.8 (8.8.8.8)

Version: 4 **Versión**
Header length: 20 bytes **Longitud de cabecera**
► Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
Total Length: 1500 **Longitud total**
Identification: 0x0067 (103)
► Flags: 0x01 (More Fragments)
Fragment offset: 0 **Desplazamiento**
Time to live: 128 **Tiempo de vida**
Protocol: UDP (17) **Protocolo**
► Header checksum: 0x3334 [validation disabled]
Source: 150.244.58.114 (150.244.58.114) **Dirección IP de origen**
Destination: 8.8.8.8 (8.8.8.8) **Dirección IP de destino**
[Source GeoIP: Unknown]
[Destination GeoIP: Unknown]
Reassembled IPv4 in frame: 3

▼ **Data (1480 bytes)**

Data: a22f12f00bf4000031323334353637383930313233343536...
[Length: 1480]

lubuntu@lubuntu: ~/Desktop/codigo

Archivo Edición Pestañas Ayuda

lubuntu@lubuntu:~/Desktop/codigo\$./practica2 -f fragipv4udp.pcap

---> NUEVO PAQUETE (nº 1) capturado el Thu Sep 12 03:47:28 2013

ANALISIS DE NIVEL DE ENLACE (NIVEL 2):

Dirección Ethernet Destino = 00-00-00-00-00-00

Dirección Ethernet Origen = 14-dd-a9-d2-ef-57

Tipo Ethernet = 0800

ANALISIS DE NIVEL DE RED IP (NIVEL 3):

Version = 4

Longitud Cabecera = 20

Logitud Total = 1500

Desplazamiento = 0

Tiempo de vida = 128

Protocolo = 17

Dirección IP Origen = 150.244.58.114

Dirección IP Destino = 8.8.8.8

Desplazamiento distinto de cero, este paquete no contiene la cabecera de nivel 4

Finalizamos analisis de este paquete

---> NUEVO PAQUETE (nº 2) capturado el Thu Sep 12 03:47:28 2013

ANALISIS DE NIVEL DE ENLACE (NIVEL 2):

Dirección Ethernet Destino = 00-00-00-00-00-00

Dirección Ethernet Origen = 14-dd-a9-d2-ef-57

Tipo Ethernet = 0800

ANALISIS DE NIVEL DE RED IP (NIVEL 3):

Version = 4

Longitud Cabecera = 20

Logitud Total = 1500

Desplazamiento = 1480

Tiempo de vida = 128

Protocolo = 17

Dirección IP Origen = 150.244.58.114

Dirección IP Destino = 8.8.8.8

Desplazamiento distinto de cero, este paquete no contiene la cabecera de nivel 4

Finalizamos analisis de este paquete

---> NUEVO PAQUETE (nº 3) capturado el Thu Sep 12 03:47:28 2013

ANALISIS DE NIVEL DE ENLACE (NIVEL 2):

Dirección Ethernet Destino = 00-00-00-00-00-00

Dirección Ethernet Origen = 14-dd-a9-d2-ef-57

Tipo Ethernet = 0800

ANALISIS DE NIVEL DE RED IP (NIVEL 3):

Version = 4

Longitud Cabecera = 20

Logitud Total = 120

Desplazamiento = 2960

Tiempo de vida = 128

Protocolo = 17

Dirección IP Origen = 150.244.58.114

Dirección IP Destino = 8.8.8.8

Desplazamiento distinto de cero, este paquete no contiene la cabecera de nivel 4

Finalizamos analisis de este paquete

Traza leida con exito

NUMERO DE PAQUETES PROCESADOS: 3

lubuntu@lubuntu:~/Desktop/codigo\$

Podemos ver que coinciden todos los campos. Una posible interrogación que se nos plantea es por qué Wireshark no nos muestra la información sobre los puerto origen, puerto destino y campo longitud de nivel 4, cosa que sí hace nuestro programa. La respuesta puede ser que Wireshark decodifica la información de nivel 4 cuando le llega el último fragmento, que en este ejemplo es el paquete número 3. Lo podemos ver a continuación.

Filter:
Expression...
Clear
Apply
Guardar

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	150.244.58.114	8.8.8.8	IPv4	1514	Fragmented IP protocol (pr
2	0.000000	150.244.58.114	8.8.8.8	IPv4	1514	Fragmented IP protocol (pr
3	0.000000	150.244.58.114	8.8.8.8	UDP	134	Source port: 41519 Desti

Frame 3: 134 bytes on wire (1072 bits), 134 bytes captured (1072 bits)

Encapsulation type: Ethernet (1)

Arrival Time: Sep 12, 2013 03:47:28.000000000 UTC

[Time shift for this packet: 0.000000000 seconds]

Epoch Time: 1378957648.000000000 seconds

[Time delta from previous captured frame: 0.000000000 seconds]

[Time delta from previous displayed frame: 0.000000000 seconds]

[Time since reference or first frame: 0.000000000 seconds]

Frame Number: 3

Frame Length: 134 bytes (1072 bits)

Capture Length: 134 bytes (1072 bits)

[Frame is marked: False]

[Frame is ignored: False]

[Protocols in frame: eth:ip:udp:data]

[Coloring Rule Name: UDP]

[Coloring Rule String: udp]

Ethernet II, Src: 14:dd:a9:d2:ef:57 (14:dd:a9:d2:ef:57), Dst: 00:00:00:00:00:00 (00:00:00:00:00:00)

Destination: 00:00:00:00:00:00 (00:00:00:00:00:00)

Source: 14:dd:a9:d2:ef:57 (14:dd:a9:d2:ef:57)

Type: IP (0x0800)

Internet Protocol Version 4, Src: 150.244.58.114 (150.244.58.114), Dst: 8.8.8.8 (8.8.8.8)

Version: 4

Header length: 20 bytes

Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))

Total Length: 120

Identification: 0x0067 (103)

Flags: 0x00

Fragment offset: 2960

Time to live: 128

Protocol: UDP (17)

Header checksum: 0x5726 [validation disabled]

Source: 150.244.58.114 (150.244.58.114)

Destination: 8.8.8.8 (8.8.8.8)

[Source GeoIP: Unknown]

[Destination GeoIP: Unknown]

[3 IPv4 Fragments (3060 bytes): #1(1480), #2(1480), #3(100)]

User Datagram Protocol, Src Port: 41519 (41519), Dst Port: appserv-http (4848)

Source port: 41519 (41519)

Destination port: appserv-http (4848)

Length: 3060

Checksum: 0x0000 (none)

Data (3052 bytes)

Data: 313233343536373839303132333435363738393031323334...

[Length: 3052]

lubuntu@lubuntu: ~/Desktop/codigo

Archivo Edición Pestañas Ayuda

lubuntu@lubuntu:~/Desktop/codigo\$./practica2 -f fragipv4udp.pcap

---> NUEVO PAQUETE (nº 1) capturado el Thu Sep 12 03:47:28 2013

ANALISIS DE NIVEL DE ENLACE (NIVEL 2):

Direccion Ethernet Destino = 00-00-00-00-00-00

Direccion Ethernet Origen = 14-dd-a9-d2-ef-57

Tipo Ethernet = 0800

ANALISIS DE NIVEL DE RED IP (NIVEL 3):

Version = 4

Longitud Cabecera = 20

Longitud Total = 1500

Desplazamiento = 0

Tiempo de vida = 128

Protocolo = 17

Direccion IP Origen = 150.244.58.114

Direccion IP Destino = 8.8.8.8

ANALISIS DE NIVEL DE TRANSPORTE UDP(NIVEL 4):

Puerto Origen = 41519

Puerto Destino = 4848

Campo Longitud = 3060

---> NUEVO PAQUETE (nº 2) capturado el Thu Sep 12 03:47:28 2013

ANALISIS DE NIVEL DE ENLACE (NIVEL 2):

Direccion Ethernet Destino = 00-00-00-00-00-00

Direccion Ethernet Origen = 14-dd-a9-d2-ef-57

Tipo Ethernet = 0800

ANALISIS DE NIVEL DE RED IP (NIVEL 3):

Version = 4

Longitud Cabecera = 20

Longitud Total = 1500

Desplazamiento = 1480

Tiempo de vida = 128

Protocolo = 17

Direccion IP Origen = 150.244.58.114

Direccion IP Destino = 8.8.8.8

Desplazamiento distinto de cero, este paquete no contiene la cabecera de nivel 4

Finalizamos analisis de este paquete

---> NUEVO PAQUETE (nº 3) capturado el Thu Sep 12 03:47:28 2013

ANALISIS DE NIVEL DE ENLACE (NIVEL 2):

Direccion Ethernet Destino = 00-00-00-00-00-00

Direccion Ethernet Origen = 14-dd-a9-d2-ef-57

Tipo Ethernet = 0800

ANALISIS DE NIVEL DE RED IP (NIVEL 3):

Version = 4

Longitud Cabecera = 20

Longitud Total = 120

Desplazamiento = 2960

Tiempo de vida = 128

Protocolo = 17

Direccion IP Origen = 150.244.58.114

Direccion IP Destino = 8.8.8.8

Desplazamiento distinto de cero, este paquete no contiene la cabecera de nivel 4

Finalizamos analisis de este paquete

Traza leida con exito

NUMERO DE PAQUETES PROCESADOS: 3

lubuntu@lubuntu:~/Desktop/codigo\$

lubuntu@lubuntu:~/Desktop/codigo\$

lubuntu@lubuntu:~/Desktop/codigo\$

lubuntu@lubuntu:~/Desktop/codigo\$

El análisis del paquete número es análogo a estos dos. El número dos tiene un desplazamiento distinto de cero, por lo que no contiene la cabecera de nivel 4 y no se imprimirá nada de este nivel. Además, tampoco es el último fragmento, por lo que no tendrá ninguna información de los puertos origen, destino, etc. que sí tenía el número 3 en Wireshark.

Ahora vamos a mostrar un ejemplo de captura en vivo, utilizando para ello el comando hping3 en la terminal.

En primer lugar, mostramos un paquete que posee cabecera de nivel 4 de tipo UDP.

Filter: Expression... Clear Apply Guardar

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.59.131	192.168.59.2	DNS	70	Standard query 0x1f99 A
2	0.030955	192.168.59.2	192.168.59.131	DNS	86	Standard query response 0x1f99
3	0.071185	192.168.59.131	150.244.214.237	TCP	54	2715 → 80 [SYN] Seq=0 Win=0 Len=0
4	0.091556	150.244.214.237	192.168.59.131	TCP	60	80 → 2715 [SYN, ACK] Seq=0 Win=0 Len=0
5	0.091642	192.168.59.131	150.244.214.237	TCP	54	2715 → 80 [RST] Seq=1 Win=0 Len=0
6	1.074399	192.168.59.131	150.244.214.237	TCP	54	2716 → 80 [SYN] Seq=0 Win=0 Len=0
7	1.121589	150.244.214.237	192.168.59.131	TCP	60	80 → 2716 [SYN, ACK] Seq=0 Win=0 Len=0

▼ Frame 3: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface eth0

Encapsulation type: Ethernet (1)

Arrival Time: Nov 9, 2018 21:43:12.282236000 UTC

[Time shift for this packet: 0.000000000 seconds]

Epoch Time: 1541799792.282236000 seconds

[Time delta from previous captured frame: 0.040230000 seconds]

[Time delta from previous displayed frame: 0.040230000 seconds]

[Time since reference or first frame: 0.071185000 seconds]

Frame Number: 3

Frame Length: 54 bytes (432 bits)

Capture Length: 54 bytes (432 bits)

[Frame is marked: False]

[Frame is ignored: False]

[Protocols in frame: eth:ip:tcp]

[Coloring Rule Name: HTTP]

[Coloring Rule String: http || tcp.port == 80]

▼ Ethernet II, Src: 00:0c:29:43:ec:bb (00:0c:29:43:ec:bb), Dst: 00:50:56:fe:12:a6 (00:50:56:fe:12:a6)

Destination: 00:50:56:fe:12:a6 (00:50:56:fe:12:a6)

Source: 00:0c:29:43:ec:bb (00:0c:29:43:ec:bb)

Type: IP (0x0800)

▼ Internet Protocol Version 4, Src: 192.168.59.131 (192.168.59.131), Dst: 150.244.214.237 (150.244.214.237)

Version: 4

Header length: 20 bytes

Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))

Total Length: 40

Identification: 0x906e (36974)

Flags: 0x00

Fragment offset: 0

Time to live: 64

Protocol: (TCP) (6)

Header checksum: 0x8054 [validation disabled]

Source: 192.168.59.131 (192.168.59.131)

Destination: 150.244.214.237 (150.244.214.237)

[Source GeoIP: Unknown]

[Destination GeoIP: Unknown]

▼ Transmission Control Protocol, Src Port: 2715 (2715), Dst Port: 80 (80), Seq: 0, Len: 0

Source port: 2715 (2715)

Destination port: 80 (80)

[Stream index: 0]

Sequence number: 0 (relative sequence number)

Acknowledgment Number: 0x76507964 [should be 0x00000000 because ACK flag is not set]

Header length: 20 bytes

▼ Flags: 0x002 (SYN)

000. = Reserved: Not set

...0. = Nonce: Not set

...0. = Congestion Window Reduced (CWR): Not set

...0. = ECN-Echo: Not set

...0. = Urgent: Not set

...0. = Acknowledgment: Not set

...0. = Push: Not set

...0. = Reset: Not set

...1. = SYN: Set SYN = 1

...0. = FIN: Not set FIN = 0

Window size value: 512

Archivo Edición Pestañas Ayuda

```

lubuntu@lubuntu:~/Desktop/codigo$ sudo ./practica2 -i eth0
---> NUEVO PAQUETE (nº 1) capturado el Fri Nov 9 21:43:12 2018
ANALISIS DE NIVEL DE ENLACE (NIVEL 2):
Direccion Ethernet Destino = 00:50:56-fe-12-a6
Direccion Ethernet Origen = 00:0c-29-43-ec-bb
Tipo Ethernet = 0800
ANALISIS DE NIVEL DE RED IP (NIVEL 3):
Version = 4
Longitud Cabecera = 20
Longitud Total = 56
Desplazamiento = 0
Tiempo de vida = 64
Protocolo = 17
Direccion IP Origen = 192.168.59.131
Direccion IP Destino = 192.168.59.2
ANALISIS DE NIVEL DE TRANSPORTE UDP(NIVEL 4):
Puerto Origen = 2571
Puerto Destino = 53
Campo Longitud = 36

---> NUEVO PAQUETE (nº 2) capturado el Fri Nov 9 21:43:12 2018
ANALISIS DE NIVEL DE ENLACE (NIVEL 2):
Direccion Ethernet Destino = 00:0c-29-43-ec-bb
Direccion Ethernet Origen = 00:50:56-fe-12-a6
Tipo Ethernet = 0800
ANALISIS DE NIVEL DE RED IP (NIVEL 3):
Version = 4
Longitud Cabecera = 20
Longitud Total = 72
Desplazamiento = 0
Tiempo de vida = 128
Protocolo = 17
Direccion IP Origen = 192.168.59.2
Direccion IP Destino = 192.168.59.131
ANALISIS DE NIVEL DE TRANSPORTE UDP(NIVEL 4):
Puerto Origen = 53
Puerto Destino = 2571
Campo Longitud = 52

---> NUEVO PAQUETE (nº 3) capturado el Fri Nov 9 21:43:12 2018
ANALISIS DE NIVEL DE ENLACE (NIVEL 2):
Direccion Ethernet Destino = 00:50:56-fe-12-a6
Direccion Ethernet Origen = 00:0c-29-43-ec-bb
Tipo Ethernet = 0800
ANALISIS DE NIVEL DE RED IP (NIVEL 3):
Version = 4
Longitud Cabecera = 20
Longitud Total = 40
Desplazamiento = 0
Tiempo de vida = 64
Protocolo = 6
Direccion IP Origen = 192.168.59.131
Direccion IP Destino = 150.244.214.237
ANALISIS DE NIVEL DE TRANSPORTE TCP (NIVEL 4):
Puerto Origen = 2715
Puerto Destino = 80
Bandera SYN = 1
Bandera FIN = 0

---> NUEVO PAQUETE (nº 4) capturado el Fri Nov 9 21:43:12 2018
ANALISIS DE NIVEL DE ENLACE (NIVEL 2):
Direccion Ethernet Destino = 00:0c-29-43-ec-bb
Direccion Ethernet Origen = 00:50:56-fe-12-a6
Tipo Ethernet = 0800
ANALISIS DE NIVEL DE RED IP (NIVEL 3):
Version = 4
Longitud Cabecera = 20
Longitud Total = 44
Desplazamiento = 0
Tiempo de vida = 128
Protocolo = 6

```

Como se puede ver coinciden todos los campos diseccionados, incluídas las *flags* SYN y FIN del nivel 4.

Ahora mostraremos un paquete que su nivel 3 no se corresponde ni con IPv4, por que el programa debería detener su análisis.

The image shows a Wireshark packet capture analysis. The packet list on the left shows several packets, with packet 19 selected. The packet details pane on the right shows the structure of the selected packet, which is an ARP request. The packet is 60 bytes long and contains the following information:

- Encapsulation type: Ethernet (I)
- Arrival Time: Nov 9, 2018 21:43:17.215723000 UTC
- Epoch Time: 1541799797.215723000 seconds
- Frame Number: 19
- Frame Length: 60 bytes (480 bits)
- Capture Length: 60 bytes (480 bits)
- Protocols in frame: eth:arp
- Coloring Rule Name: ARP
- Coloring Rule String: arp
- Ethernet II, Src: 00:50:56:fe:12:a6 (00:50:56:fe:12:a6), Dst: 00:0c:29:43:ec:bb (00:0c:29:43:ec:bb)
- Destination: 00:0c:29:43:ec:bb (00:0c:29:43:ec:bb)
- Source: 00:50:56:fe:12:a6 (00:50:56:fe:12:a6)
- Type: ARP (0x0806) **ARP is IPV4**
- Padding: 00000000000000000000000000000000
- Address Resolution Protocol (reply)

The packet bytes pane on the right shows the raw data of the packet, which is an ARP request. The analysis shows that the packet is an ARP request for the IP address 192.168.59.2. The packet is captured on the interface eth0.

Por último vamos a mostrar 4 salidas utilizando cada uno de los filtros. En estos casos se han utilizado los filtros individualmente, pero su funcionamiento con varios filtros es análogo ya que el programa detiene el análisis de un paquete en el momento que un campo filtrado no coincida con el filtro introducido.

Se ha guardado una captura de tráfico de la interfaz anterior en un fichero llamado **ejemplo.pcap** que se incluye en la carpeta de la entrega por si desea comprobar cualquier detalle.

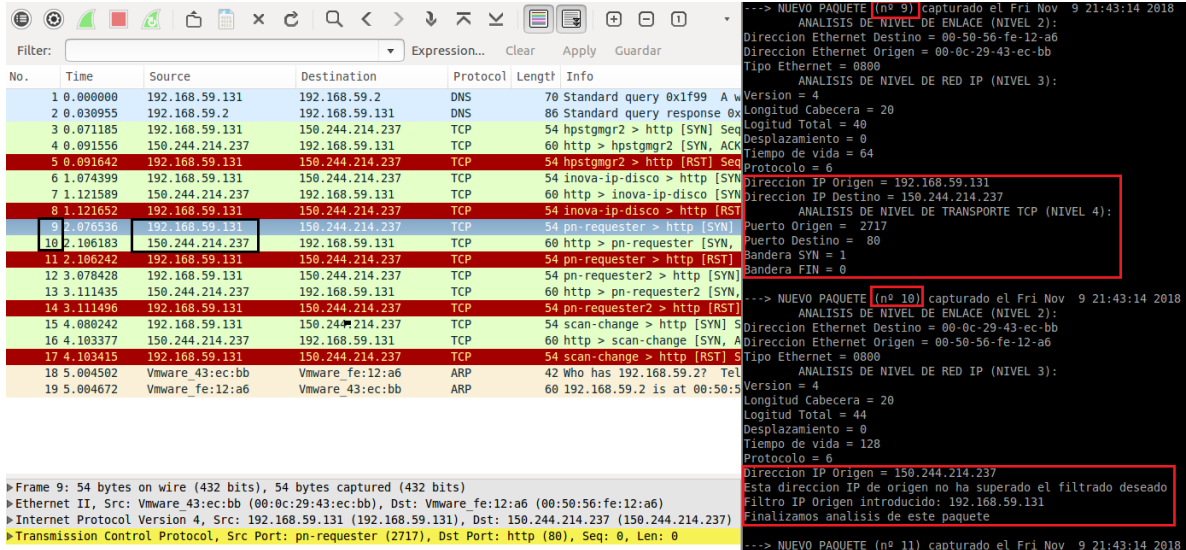
Programa ejecutado como: `./practica2 -f ejemplo.pcap -ipd 150.244.214.237`

The image shows a Wireshark packet capture analysis of a file named 'ejemplo.pcap'. The packet list on the left shows a list of packets, with packet 9 selected. The packet details pane on the right shows the structure of the selected packet, which is a TCP reset (RST). The packet is 54 bytes long and contains the following information:

- Version: 4
- Longitud Cabecera: 20
- Logitud Total: 40
- Desplazamiento: 0
- Tiempo de vida: 64
- Protocolo: 6
- Direccion IP Origen = 192.168.59.131
- Direccion IP Destino = 150.244.214.237
- Puerto Origen = 2717
- Puerto Destino = 80
- Bandera SYN = 1
- Bandera FIN = 0
- Analisis de Nivel de Transporte TCP (Nivel 4):
- Analisis de Nivel de Enlace (Nivel 2):
- Direccion Ethernet Destino = 00:0c:29:43:ec:bb
- Direccion Ethernet Origen = 00:50:56:fe:12:a6
- Tipo Ethernet = 0800
- El siguiente protocolo no se corresponde con un protocolo IP.
- Finalizamos analisis de este paquete

The packet bytes pane on the right shows the raw data of the packet, which is a TCP reset. The analysis shows that the packet is a TCP reset for the IP address 150.244.214.237. The packet is captured on the interface eth0.

Programa ejecutado como: `./practica2 -f ejemplo.pcap -ipo 192.168.59.131`



Filter: Expression... Clear Apply Guardar

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.59.131	192.168.59.2	DNS	70	Standard query 0x1f99 A w
2	0.030955	192.168.59.2	192.168.59.131	DNS	86	Standard query response 0x
3	0.071185	192.168.59.131	150.244.214.237	TCP	54	hpstgmgr2 > http [SYN] Seq
4	0.091556	150.244.214.237	192.168.59.131	TCP	60	http > hpstgmgr2 [SYN, ACK
5	0.091642	192.168.59.131	150.244.214.237	TCP	54	hpstgmgr2 > http [RST] Seq
6	1.074399	192.168.59.131	150.244.214.237	TCP	54	inova-ip-disco > http [SYN]
7	1.121589	150.244.214.237	192.168.59.131	TCP	60	http > inova-ip-disco [SYN]
8	1.121652	192.168.59.131	150.244.214.237	TCP	54	inova-ip-disco > http [RST]
9	2.076536	192.168.59.131	150.244.214.237	TCP	54	pn-requester > http [SYN]
10	2.106183	150.244.214.237	192.168.59.131	TCP	60	http > pn-requester [SYN,
11	2.106242	192.168.59.131	150.244.214.237	TCP	54	pn-requester > http [RST]
12	3.078428	192.168.59.131	150.244.214.237	TCP	54	pn-requester2 > http [SYN]
13	3.111435	150.244.214.237	192.168.59.131	TCP	60	http > pn-requester2 [SYN,
14	3.111496	192.168.59.131	150.244.214.237	TCP	54	pn-requester2 > http [RST]
15	4.080242	192.168.59.131	150.244.214.237	TCP	54	scan-change > http [SYN] S
16	4.103377	150.244.214.237	192.168.59.131	TCP	60	http > scan-change [SYN, A
17	4.103415	192.168.59.131	150.244.214.237	TCP	54	scan-change > http [RST] S
18	5.004502	Vmware_43:ec:bb	Vmware_fe:12:a6	ARP	42	Who has 192.168.59.2? Tel
19	5.004672	Vmware_fe:12:a6	Vmware_43:ec:bb	ARP	60	192.168.59.2 is at 00:50:5

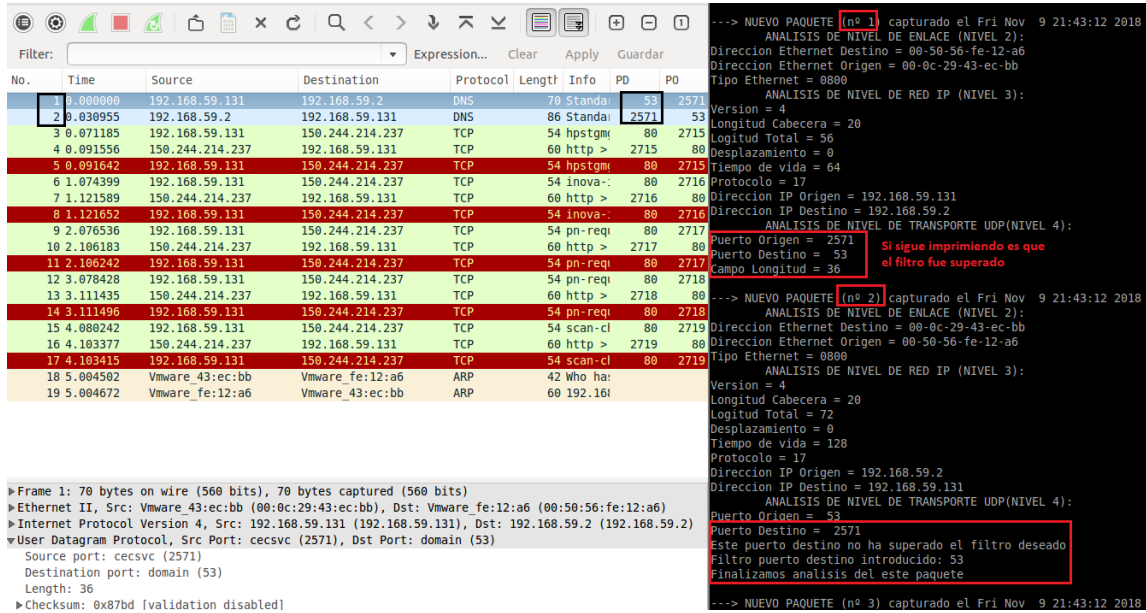
Frame 9: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
 Ethernet II, Src: Vmware_43:ec:bb (00:0c:29:43:ec:bb), Dst: Vmware_fe:12:a6 (00:50:56:fe:12:a6)
 Internet Protocol Version 4, Src: 192.168.59.131 (192.168.59.131), Dst: 150.244.214.237 (150.244.214.237)
 Transmission Control Protocol, Src Port: pn-requester (2717), Dst Port: http (80), Seq: 0, Len: 0

---> NUEVO PAQUETE (nº 9) capturado el Fri Nov 9 21:43:14 2018
 ANALISIS DE NIVEL DE ENLACE (NIVEL 2):
 Direccion Ethernet Destino = 00-50-56-fe-12-a6
 Direccion Ethernet Origen = 00-0c-29-43-ec-bb
 Tipo Ethernet = 0800
 ANALISIS DE NIVEL DE RED IP (NIVEL 3):
 Version = 4
 Longitud Cabecera = 20
 Logitud Total = 40
 Desplazamiento = 0
 Tiempo de vida = 64
 Protocolo = 6
 Direccion IP Origen = 192.168.59.131
 Direccion IP Destino = 150.244.214.237
 ANALISIS DE NIVEL DE TRANSPORTE TCP (NIVEL 4):
 Puerto Origen = 2717
 Puerto Destino = 80
 Bandera SYN = 1
 Bandera FIN = 0

---> NUEVO PAQUETE (nº 10) capturado el Fri Nov 9 21:43:14 2018
 ANALISIS DE NIVEL DE ENLACE (NIVEL 2):
 Direccion Ethernet Destino = 00-50-56-fe-12-a6
 Direccion Ethernet Origen = 00-50-56-fe-12-a6
 Tipo Ethernet = 0800
 ANALISIS DE NIVEL DE RED IP (NIVEL 3):
 Version = 4
 Longitud Cabecera = 20
 Logitud Total = 44
 Desplazamiento = 0
 Tiempo de vida = 128
 Protocolo = 6
 Direccion IP Origen = 150.244.214.237
 Esta direccion IP de origen no ha superado el filtrado deseado
 Filtro IP Origen introducido: 192.168.59.131
 Finalizamos analisis de este paquete

---> NUEVO PAQUETE (nº 11) capturado el Fri Nov 9 21:43:14 2018

Programa ejecutado como: `./practica2 -f ejemplo.pcap -pd 53`



Filter: Expression... Clear Apply Guardar

No.	Time	Source	Destination	Protocol	Length	Info	PD	PD
1	0.000000	192.168.59.131	192.168.59.2	DNS	70	Standard query 0x1f99 A w	53	2571
2	0.030955	192.168.59.2	192.168.59.131	DNS	86	Standard query response 0x	2571	53
3	0.071185	192.168.59.131	150.244.214.237	TCP	54	hpstgmgr2 > http [SYN] Seq	80	2715
4	0.091556	150.244.214.237	192.168.59.131	TCP	60	http > hpstgmgr2 [SYN, ACK	2715	80
5	0.091642	192.168.59.131	150.244.214.237	TCP	54	hpstgmgr2 > http [RST] Seq	80	2715
6	1.074399	192.168.59.131	150.244.214.237	TCP	54	inova-ip-disco > http [SYN]	80	2716
7	1.121589	150.244.214.237	192.168.59.131	TCP	60	http > inova-ip-disco [SYN]	2716	80
8	1.121652	192.168.59.131	150.244.214.237	TCP	54	inova-ip-disco > http [RST]	80	2716
9	2.076536	192.168.59.131	150.244.214.237	TCP	54	pn-requester > http [SYN]	80	2717
10	2.106183	150.244.214.237	192.168.59.131	TCP	60	http > pn-requester [SYN,	2717	80
11	2.106242	192.168.59.131	150.244.214.237	TCP	54	pn-requester > http [RST]	80	2717
12	3.078428	192.168.59.131	150.244.214.237	TCP	54	pn-requester2 > http [SYN]	80	2718
13	3.111435	150.244.214.237	192.168.59.131	TCP	60	http > pn-requester2 [SYN,	2718	80
14	3.111496	192.168.59.131	150.244.214.237	TCP	54	pn-requester2 > http [RST]	80	2718
15	4.080242	192.168.59.131	150.244.214.237	TCP	54	scan-cl > http [SYN] S	80	2719
16	4.103377	150.244.214.237	192.168.59.131	TCP	60	http > scan-cl [SYN, A	2719	80
17	4.103415	192.168.59.131	150.244.214.237	TCP	54	scan-cl > http [RST] S	80	2719
18	5.004502	Vmware_43:ec:bb	Vmware_fe:12:a6	ARP	42	Who has 192.168.59.2? Tel		
19	5.004672	Vmware_fe:12:a6	Vmware_43:ec:bb	ARP	60	192.168.59.2 is at 00:50:5		

Frame 1: 70 bytes on wire (560 bits), 70 bytes captured (560 bits)
 Ethernet II, Src: Vmware_43:ec:bb (00:0c:29:43:ec:bb), Dst: Vmware_fe:12:a6 (00:50:56:fe:12:a6)
 Internet Protocol Version 4, Src: 192.168.59.131 (192.168.59.131), Dst: 192.168.59.2 (192.168.59.2)
 User Datagram Protocol, Src Port: cecsvc (2571), Dst Port: domain (53)
 Source port: cecsvc (2571)
 Destination port: domain (53)
 Length: 36
 Checksum: 0x87bd [validation disabled]

---> NUEVO PAQUETE (nº 1) capturado el Fri Nov 9 21:43:12 2018
 ANALISIS DE NIVEL DE ENLACE (NIVEL 2):
 Direccion Ethernet Destino = 00-50-56-fe-12-a6
 Direccion Ethernet Origen = 00-0c-29-43-ec-bb
 Tipo Ethernet = 0800
 ANALISIS DE NIVEL DE RED IP (NIVEL 3):
 Version = 4
 Longitud Cabecera = 20
 Logitud Total = 56
 Desplazamiento = 0
 Tiempo de vida = 64
 Protocolo = 17
 Direccion IP Origen = 192.168.59.131
 Direccion IP Destino = 192.168.59.2
 ANALISIS DE NIVEL DE TRANSPORTE UDP (NIVEL 4):
 Puerto Origen = 2571
 Puerto Destino = 53
 Campo Longitud = 36
 Si sigue imprimiendo es que el filtro fue superado

---> NUEVO PAQUETE (nº 2) capturado el Fri Nov 9 21:43:12 2018
 ANALISIS DE NIVEL DE ENLACE (NIVEL 2):
 Direccion Ethernet Destino = 00-0c-29-43-ec-bb
 Direccion Ethernet Origen = 00-50-56-fe-12-a6
 Tipo Ethernet = 0800
 ANALISIS DE NIVEL DE RED IP (NIVEL 3):
 Version = 4
 Longitud Cabecera = 20
 Logitud Total = 72
 Desplazamiento = 0
 Tiempo de vida = 128
 Protocolo = 17
 Direccion IP Origen = 192.168.59.2
 Direccion IP Destino = 192.168.59.131
 ANALISIS DE NIVEL DE TRANSPORTE UDP (NIVEL 4):
 Puerto Origen = 53
 Puerto Destino = 2571
 Este puerto destino no ha superado el filtro deseado
 Filtro puerto destino introducido: 53
 Finalizamos analisis del este paquete

---> NUEVO PAQUETE (nº 3) capturado el Fri Nov 9 21:43:12 2018

Programa ejecutado como: `./practica2 -f ejemplo.pcap -po 2571`

The screenshot shows the Wireshark interface with a packet capture of 'ejemplo.pcap'. The packet list on the left shows 19 packets. The details pane on the right shows the selected packet (No. 1) with the following information:

- Version = 4
- Longitud Cabecera = 20
- Logitud Total = 56
- Desplazamiento = 0
- Tiempo de vida = 64
- Protocolo = 17
- Direccion IP Origen = 192.168.59.131
- Direccion IP Destino = 192.168.59.2
- Analisis de Nivel de Transporte UDP (Nivel 4):
- Puerto Origen = 2571
- Puerto Destino = 53
- Campo Longitud = 36

The packet details pane also shows the packet's structure and the filter applied: `Source port: cecsvc (2571)`.

Finalmente, como último ejemplo de salida entregamos un informe de memoria de Valgrind:

```
Traza leida con exito
NUMERO DE PAQUETES PROCESADOS: 19
==6873==
==6873==  HEAP SUMMARY:
==6873==    in use at exit: 0 bytes in 0 blocks
==6873==   total heap usage: 27 allocs, 27 frees, 67,068 bytes allocated
==6873==
==6873== All heap blocks were freed -- no leaks are possible
==6873==
==6873== For counts of detected and suppressed errors, rerun with: -v
==6873== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
lubuntu@lubuntu:~/Desktop/codigo$
```

En este ejemplo se ejecutó el programa leyendo una traza guardada, pero el uso de memoria es análogo para el caso de una captura en vivo.

NOTA: Como actualización de última hora hemos decidido mostrar el campo Longitud de cabecera y Desplazamiento de una forma un poco diferente a lo mostrado en las capturas anteriores.

Debido a que el campo Logitud de cabecera expresa el número de palabras de 32 bits que tiene la cabecera, hay que multiplicar por cuatro este valor para obtener la longitud en bytes. En la documentación no se especificaba que valor poner, por lo que simplemente nos hemos curado en salud y hemos indicado los dos.

Algo parecido pasa con el campo Desplazamiento. Como para este campo solo hay reservados 13 bits, se ha decidido que el desplazamiento siempre sea múltiplo de ocho, por lo que el valor recogido en ese campo debe ser multiplicado por ocho para obtener el valor real de desplazamiento. Por el mismo razonamiento que el anterior, hemos incluido los dos.

A continuación mostramos una salida del fichero **fragipv4udp.pcap** utilizando este formato de salida, muy similar al mostrado en las capturas de arriba.

```

AlejandroSantorum@DESKTOP-GC6HCIA: /mnt/c/users/Alejandro_Santorum/Desktop/practica2_1302_P0$ ./practica2 -f fragipv4udp.pcap
---> NUEVO PAQUETE (nº 1) capturado el Thu Sep 12 05:47:28 2013
      ANALISIS DE NIVEL DE ENLACE (NIVEL 2):
Direccion Ethernet Destino = 00-00-00-00-00-00
Direccion Ethernet Origen = 14-dd-a9-d2-ef-57
Tipo Ethernet = 0800
      ANALISIS DE NIVEL DE RED IP (NIVEL 3):
Version = 4
Valor del campo Longitud Cabecera = 5
Longitud de Cabecera real (interpretando el valor del campo) = 20
Longitud Total = 1500
Valor campo desplazamiento en decimal = 0
Desplazamiento real (interpretando el valor del campo) = 0
Tiempo de vida = 128
Protocolo = 17
Direccion IP Origen = 150.244.58.114
Direccion IP Destino = 8.8.8.8
      ANALISIS DE NIVEL DE TRANSPORTE UDP(NIVEL 4):
Puerto Origen = 41519
Puerto Destino = 4848
Campo Longitud = 3060

---> NUEVO PAQUETE (nº 2) capturado el Thu Sep 12 05:47:28 2013
      ANALISIS DE NIVEL DE ENLACE (NIVEL 2):
Direccion Ethernet Destino = 00-00-00-00-00-00
Direccion Ethernet Origen = 14-dd-a9-d2-ef-57
Tipo Ethernet = 0800
      ANALISIS DE NIVEL DE RED IP (NIVEL 3):
Version = 4
Valor del campo Longitud Cabecera = 5
Longitud de Cabecera real (interpretando el valor del campo) = 20
Longitud Total = 1500
Valor campo desplazamiento en decimal = 185
Desplazamiento real (interpretando el valor del campo) = 1480
Tiempo de vida = 128
Protocolo = 17
Direccion IP Origen = 150.244.58.114
Direccion IP Destino = 8.8.8.8
Desplazamiento distinto de cero, este paquete no contiene la cabecera de nivel 4
Finalizamos analisis de este paquete

```

4 Decisiones de diseño

En primer lugar ya ha sido comentado que no hemos utilizado el ejemplo de *parseo* de parámetros de entrada aportado en Moodle, sino que hemos implementado nuestra propia función.

Por otro lado, a la hora de avanzar el puntero `uint8_t *pack` no se ha comprobado que no acababa en una zona de memoria no permitida (*segmentation fault*) asegurándonos que el avance era menor que el caplen y/o que la longitud de la cabecera, pero esto creemos que es innecesario para el objetivo de este ejercicio ya que todo paquete tiene una cabecera de nivel 2 idéntica y, en este nivel, comprobamos que el siguiente protocolo es IPv4, por lo que ya tendría una cabecera de nivel 3 con una estructura determinada; e igual para el caso de avanzar hasta la cabecera de nivel 4, ya que hacemos la comprobación en el campo desplazamiento y en el campo protocolo en la cabecera de nivel 3.

Por último, comentar que el análisis de las cabeceras de los 3 niveles se realiza en la función `void package_treat(...)`. Somos conscientes de que de separar el análisis de cada nivel en una función distinta aumentaría el estilo del código, pero aumentarían las comprobaciones (pasos de argumentos entre funciones, retornos, manejo de punteros, etc.) que tendríamos que realizar, por lo que se ha optado por separar con unos comentarios bien visibles el análisis de cada nivel, en lugar de separarlo en diferentes funciones.

5 Conclusión

Esta práctica ha resultado ser muy útil para comprender las cabeceras de los niveles de enlace, red y transporte; aprendiendo como seleccionar cualquier campo de interés y mostrarlo/guardarlo para su posterior uso.

Esperamos poder afianzar dichos conocimientos en las siguientes prácticas.