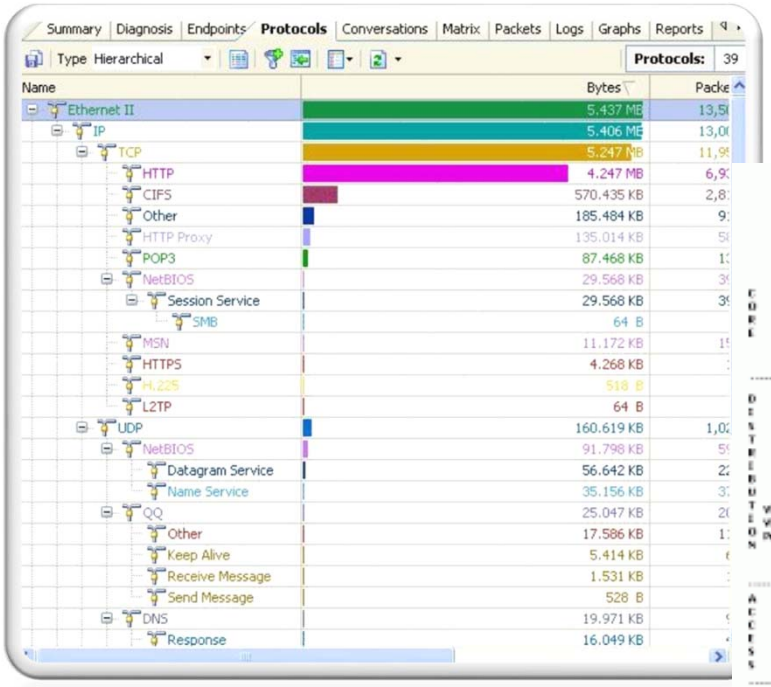
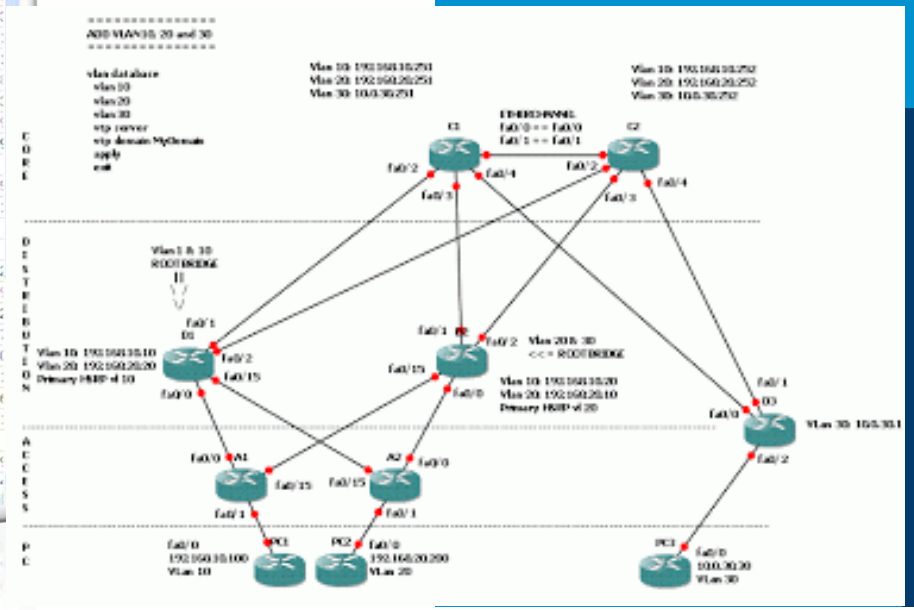
VNiVERSiDAD D SALAMANCA

http://informatica.usal.es/gii

Analizadores





VNiVERSiDAD D SALAMANCA

VNiVERSiDAD D SALAMANCA

Contenido

- Analizadores
  - El modo promiscuo
  - tcpdump
  - wireshark
  - Encapsulación de protocolos en Internet
  - Referencias

VNiVERSiDAD D SALAMANCA

Analizadores

- Antecedentes
  - Todo administrador de redes ha tenido que enfrentarse alguna vez a una pérdida del rendimiento de la red que gestiona.
  - No siempre es sencillo, por falta de tiempo y recursos o por desconocimiento de las herramientas apropiadas, tener claros los motivos por los que esto ha sucedido.
  - En ocasiones, incluso se ha podido llegar a perder la conectividad o bien ciertos equipos han podido desconectarse sin motivo aparente.
- Los analizadores de red (también conocidos como *sniffers*) capturan los paquetes que circulan por la red y permiten
  - Detectar y analizar fallos de conectividad
  - Medir el tráfico y la congestión (o saturación) de la red
  - Auditar la seguridad, detectar intrusos, virus, …
  - Ejemplos: tcpdump (modo consola) y wireshark (modo gráfico)

VNiVERSiDAD D SALAMANCA

El modo promiscuo

- El modo promiscuo es aquel en el que un elemento de una red (con medio compartido) captura todo el tráfico que circula por ella
  - En el modo normal los equipos sólo recogen las tramas que tienen como dirección física de destino la suya
  - En el modo promiscuo un elemento intermedio de la red capturaría todas las tramas, las destinadas a él mismo y al resto de máquinas (que habitualmente desecharía).
- En Linux
  - Se activa: `ifconfig <interfaz> promisc`
  - Se desactiva: `ifconfig <interfaz> -promisc`
- En Windows
  - Se activa/desactiva a través de controladores

VNiVERSiDAD D SALAMANCA

Tcpdump: Introducción

- *Tcpdump* es una herramienta en línea de comandos para sistemas operativos UNIX cuya funcionalidad principal es analizar el tráfico que circula por la red
  - Sitio web oficial http://www.tcpdump.org
  - Páginas de manual http://www.tcpdump.org/tcpdump_man.html
  - Se basa en la biblioteca de captura *pcap*
- La versión análoga para sistemas Windows es *Windump*
  - Sitio web oficial http://www.winpcap.org/windump/
  - La implementación de la biblioteca *pcap* es *winpcap*

VNiVERSiDAD D SALAMANCA

Tcpdump: Uso básico (I)

- Entrar como superusuario (root)
- Instalación en debian:

  apt-get install tcpdump
- Sintaxis

```
Usage: tcpdump [-aAbdDefIKlLnNOpqRStuUvxX] [ -B size ] [ -c count ]
        [ -C file_size ] [ -E algo:secret ] [ -F file ] [ -G seconds ]
        [ -i interface ] [ -M secret ] [ -r file ]
        [ -s snaplen ] [ -T type ] [ -w file ] [ -W filecount ]
        [ -y datalinktype ] [ -z command ] [ -Z user ]
        [ expression ]
```

- Si no usamos la opción “-c count” u otra similar para programar el final de la captura, podemos terminarla pulsando Ctrl+C

VNiVERSiDAD D SALAMANCA

Tcpdump: Uso básico (II)

- **-D**: Muestra las interfaces de red por las que escuchar (también /sbin/ifconfig -a)
- **-n**: No resolver direcciones de red
- **-w fichero**: Volcar una captura hacia fichero (trabajar en *offline*). Este tipo de ficheros puede ser leído por otros analizadores (p. ej.: *Wireshark*)
- **-r fichero**: Muestra una captura previa desde fichero (trabajar en *offline*)
- **-s longitud**: Establecer la longitud de los datos que captura (por defecto 68 bytes). Es habitual usar la MTU del medio (en ethernet –s 1500)
- **-v, -vv**: Cantidad de información a interpretar
- **-x**: imprime el contenido de los paquetes (-X imprime en ASCII)

VNiVERSiDAD D SALAMANCA

Tcpdump: Uso básico (y III)

- Ejemplos:
  - Mostrar por pantalla una captura de paquetes que circulan por la interfaz `eth0`

    ```
    tcpdump -i eth0
    ```
  - Grabar en el fichero `eth0.cap` una captura de paquetes visibles desde la interfaz eth0 con un nivel de detalle alto

    ```
    tcpdump -i eth0 -vv -w eth0.cap
    ```
  - Mostar por pantalla una captura previa desde el fichero `eth0.cap`

    ```
    tcpdump -r eth0.cap
    ```
  - Realizar una captura sin límite de tamaño de los paquetes por cualquier interfaz

    ```
    tcpdump -s 0
    ```

VNiVERSiDAD D SALAMANCA

Tcpdump: Interpretar una captura (I)

- Peticiones ARP/RARP sobre ethernet
  - RFC 826 y RFC 1293 respectivamente
  - Aparecen de la siguiente manera:

```
18:33:49.908612 arp who-has 192.168.1.2 tell 192.168.1.1
18:33:49.908691 arp reply 192.168.1.2 is-at 0:2:a5:ee:ec:10
```

  - En el ejemplo anterior la máquina con IP 192.168.1.1 pregunta por la MAC de la 192.168.1.2 (asumimos que están en la misma subred). Muy probablemente se ha usado la opción -n ya que aparecen direcciones en lugar de nombres

VNiVERSiDAD D SALAMANCA

Tcpdump: Interpretar una captura (II)

- Peticiones TCP (I)
  - Los paquetes TCP aparecen de la siguiente manera
    - src > dst: flags [dataseq ack window urgent options
  - El significado de dichos parámetros es el siguiente
    - src: Dirección y puerto origen (si no se especifica -n la IP se resuelve vía DNS y el puerto en el /etc/services)
    - dst: Dirección y puerto destino, como en el caso anterior
    - flags: Flags de la cabecera TCP. Un "." significa que no hay flags o bien una combinación de S (SYN), F (FIN), P (PUSH), W (*Congestion Window Reduced* (CWR) - reducción de la ventana de congestión), E (ECN (*Explicit Congestion Notification*) eco).
    - dataseq: Número de secuencia del primer byte de datos en este segmento TCP. El formato es primero:ultimo(n), que significa que desde primero a ultimo (sin incluir ultimo) hay un total de n bytes de datos.
    - ack: Número de asentimiento. Indica el número siguiente de secuencia que se espera recibir (los SYN también se asienten)
    - win: Tamaño de la ventana de recepción
    - urgent: Existen datos urgentes
    - options: Indica la existencia de opciones (aparecen entre < y >)

VNiVERSiDAD D SALAMANCA

Tcpdump: Interpretar una captura (III)

- Peticiones TCP (y II)
  - El siguiente ejemplo simula una conexión originada por la máquina rtsg con destino a csam, con el servicio rlogin

```
1.rtsg.1023 > csam.login: S 768512:768512(0) win 4096 <mss 1024>
2.csam.login > rtsg.1023: S 947648:947648(0) ack 768513 win 4096 <mss 1024>
3.rtsg.1023 > csam.login: . ack 1 win 4096
4.rtsg.1023 > csam.login: P 1:2(1) ack 1 win 4096
5.csam.login > rtsg.1023: . ack 2 win 4096
6.rtsg.1023 > csam.login: P 2:21(19) ack 1 win 4096
7.csam.login > rtsg.1023: P 1:2(1) ack 21 win 4077
8.csam.login > rtsg.1023: P 2:3(1) ack 21 win 4077 urg 1
9.csam.login > rtsg.1023: P 3:4(1) ack 21 win 4077 urg 1
```

VNiVERSiDAD D SALAMANCA

Tcpdump: Interpretar una captura (IV)

- Peticiones UDP (I)
  - Los paquetes UDP aparecen de la siguiente manera
    - origen.srcport > destino.dsrpot: udp len
  - El significado de dichos parámetros es el siguiente
    - origen: Nombre o dirección origen
      - srcport: Puerto origen
    - destino: Nombre o dirección destino
      - dstport: Puerto destino
    - len: Longitud de los datos de usuario
  - Algunos servicios UDP son reconocidos (a través del puerto de origen/destino) y se muestra la información del protocolo de nivel superior (con más o menos detalle usando las opciones -v,-vv,-vvv). En particular DNS y NFS.

VNiVERSiDAD D SALAMANCA

Tcpdump: Interpretar una captura (V)

- Peticiones UDP (y II)
  - El siguiente ejemplo simula el envío de un datagrama UDP de 84 bytes por el puerto del servicio **rwho** desde la máquina actinide hacia la dirección broadcast (todas las máquinas de la subred)

```
actinide.who > broadcast.who: udp 84
```

  - En las páginas de manual de la utilidad tcpdump se encuentran muchos más ejemplos para identificar las salidas de protocolos reconocidos como el DNS (peticiones y respuestas) entre otros.

```
h2opolo.1538 > helios.domain: 3+ A? ucbvax.berkeley.edu. (37)
helios.domain > h2opolo.1538: 3 3/3/7 A 128.32.137.3 (273)
```

Tcpdump: Interpretar una captura (VI)

- Fragmentos de datagramas IP
  - Los fragmentos de datagramas IP se indican con una expresión entre paréntesis en un lado de la siguiente manera:

    ```
    (frag id:size@offset+) (frag id:size@offset)
    ```

  - Donde:
    - id: es el identificador del fragmento
    - size: tamaño del fragmento
    - Offset: posición del fragmento en el datagrama original. Si existe el + al final de offset significa que aún quedan más fragmentos. En caso de ausencia, que es el último
  - Los datos del protocolo del nivel superior sólo se muestran en el primer fragmento

VNiVERSiDAD D SALAMANCA

Tcpdump: Filtrar una captura (I)

- Filtro
  - Un filtro es una expresión que va detrás de las opciones y que selecciona sólo los paquetes que cumplen el filtro
- Modificadores
  - type [host|net|port]
    - Máquina [host], red [net] o puerto concreto [port]
  - dir [src|dst|src or dst|src and dst]
    - Especifica desde [src] o hacia dónde [dst] se dirige la información
    - Se pueden combinar con operadores lógicos (or y and) y paréntesis
  - proto [tcp|udp|ip|ether]
    - Protocolo que queremos capturar
    - En el caso de *ether* captura tramas a nivel de enlace (arp, rarp, fddi)

VNiVERSiDAD D SALAMANCA

Tcpdump: Filtrar una captura (II)

- Combinaciones de modificadores permitidas (I)
  - [dst|src] host maquina
    - Este filtro se cumple si la dirección de destino (dst) u origen (src) coincide con maquina.
    - Ejemplo: `tcpdump src host 192.168.1.1`
  - [dst|src] net red
    - Este filtro se cumple si la dirección de red destino u origen coinciden con red
    - Ejemplo: `tcpdump net 10.0.0.0/24` ó `tcpdump net 10.0.0.0 netmask 255.255.255.0`
  - [dst|src] port puerto
    - Este filtro se cumple si el puerto (ya sea udp o tcp) coincide con puerto
    - Ejemplo: `tcpdump dst port 53`

VNiVERSiDAD D SALAMANCA

Tcpdump: Filtrar una captura (III)

- Combinaciones de modificadores permitidas (II)
  - ip proto procolo (también ip6 proto protocolo)
    - Escucha el protocolo que se indique en protocolo (icmp, icmp6, igmp, igrp, ah, esp, udp ó tcp). En caso de usar icmp, udp o tcp hay que anteponer un \ (en unix \\).
    - Ejemplo: `tcpdump ip proto \\udp`
  - ip broadcast (no existe en ip6)
    - Escucha los paquetes que van dirigidos a la dirección de difusión de IP
    - Ejemplo: `tcpdump ip broadcast`
  - ip multicast (también ip6 multicast)
    - Escucha los paquetes que van dirigidos a la dirección de multicast IP
    - Ejemplo: `tcpdump ip multicast`

Tcpdump: Filtrar una captura (IV)

- Combinaciones de modificadores permitidas (y III)
  - ether src|dst|host edir
    - Este filtro se cumple si la dirección de origen (src), destino (dst) o cualquiera de las dos (host) coincide con la dirección ethernet (edir).
    - Ejemplo: `tcpdump ether dst 0:2:a5:ee:ec:10`
  - ether proto procolo
    - Escucha el protocolo que se indique en protocolo (ip, ip6, arp, rarp, etc.)
    - Ejemplo: `tcpdump ether proto \\arp`
  - ether broadcast
    - Escucha los paquetes que van dirigidos a la dirección de difusión ethernet
    - Ejemplo: `tcpdump ether broadcast`
  - ether multicast
    - Escucha los paquetes que van dirigidos a la dirección de difusión ethernet
    - Ejemplo: `tcpdump ether multicast`

Tcpdump: Filtrar una captura (V)

- Combinaciones de filtros
  - Los filtros anteriores se pueden combinar por medio de operadores lógicos (not, and y or) para formar expresiones más complejas.
  - También se pueden utilizar los paréntesis (anteponiendo '\')
  - Ejemplos:
    - Capturar el tráfico web

      ```
      tcpdump tcp and port 80
      ```
    - Capturar todas las peticiones DNS

      ```
      tcpdump udp and dst port 53
      ```
    - Capturar el tráfico por telnet o ssh

      ```
      tcpdump tcp and \(port 22 or port 23\)
      ```
    - Capturar todo el tráfico tcp excepto el web

      ```
      tcpdump tcp and not port 80
      ```

VNiVERSiDAD D SALAMANCA

Wireshark: Introducción

- Wireshark es un analizador de protocolos *open-source* diseñado por Gerald Combs y que actualmente está disponible para plataformas Windows y Unix
  - Conocido originalmente como Ethereal, su principal objetivo es el análisis de tráfico además de ser una excelente aplicación didáctica para el estudio de las comunicaciones y para la resolución de problemas de red
  - También se apoya en la biblioteca pcap para la captura de paquetes
  - Sitio web oficial: http://www.wireshark.org
  - Documentación oficial: http://www.wireshark.org/docs/
  - Existen certificaciones oficiales para un perfil profesional de auditor

VNiVERSiDAD D SALAMANCA

Wireshark: características

- Características
  - Implementa una amplia gama de filtros que facilitan la definición de criterios de búsqueda para los más de 1100 protocolos soportados actualmente (versión 1.4.3)
  - Dispone de una interfaz sencilla e intuitiva que permite desglosar por capas cada uno de los paquetes capturados
  - Wireshark también dispone de una versión en línea de comandos similar a tcpdump, denominada tshark
  - Al igual que tcpdump es posible que wireshark no sea capaz de interpretar ciertos protocolos de nivel de aplicación (esta situación se reproducirá con las prácticas ad-hoc de la asignatura)
  - Wireshark reconoce capturas realizadas con tcpdump

VNiVERSiDAD D SALAMANCA

Wireshark: entorno

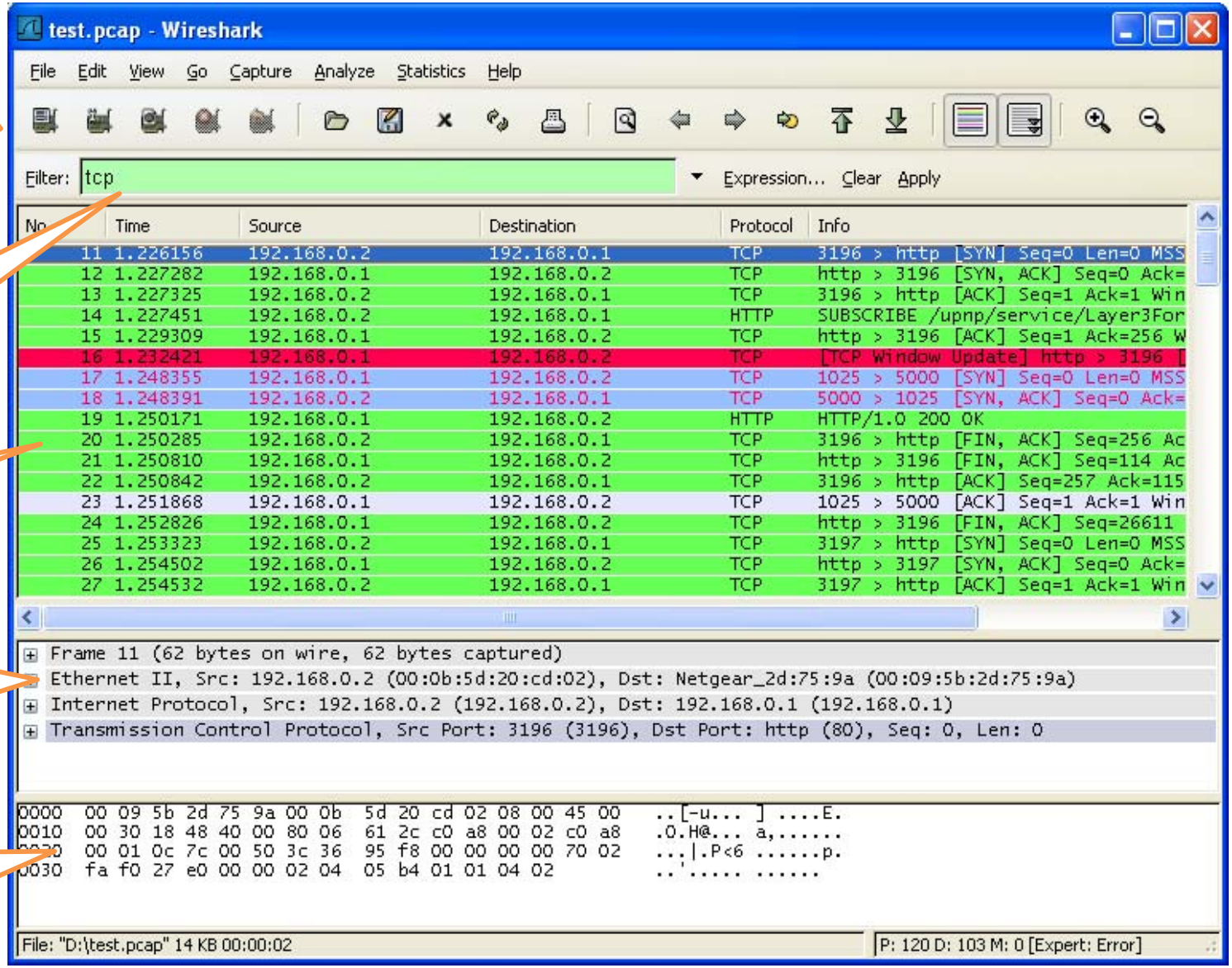
Barra de menú y controles para seleccionar interfaz, iniciar, parar, abrir y guardar una captura

Filtro de captura

Paquetes capturados

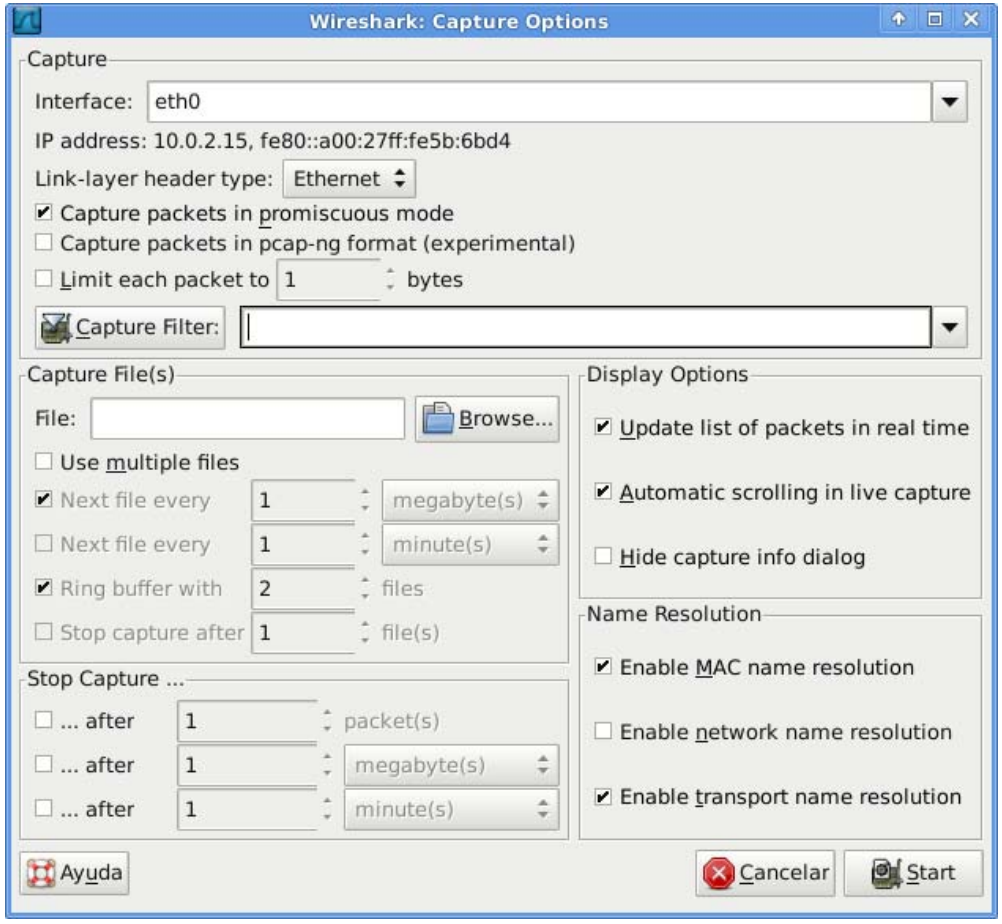
Desglose por capas del paquete seleccionado

Contenido del paquete en hexadecimal y ASCII



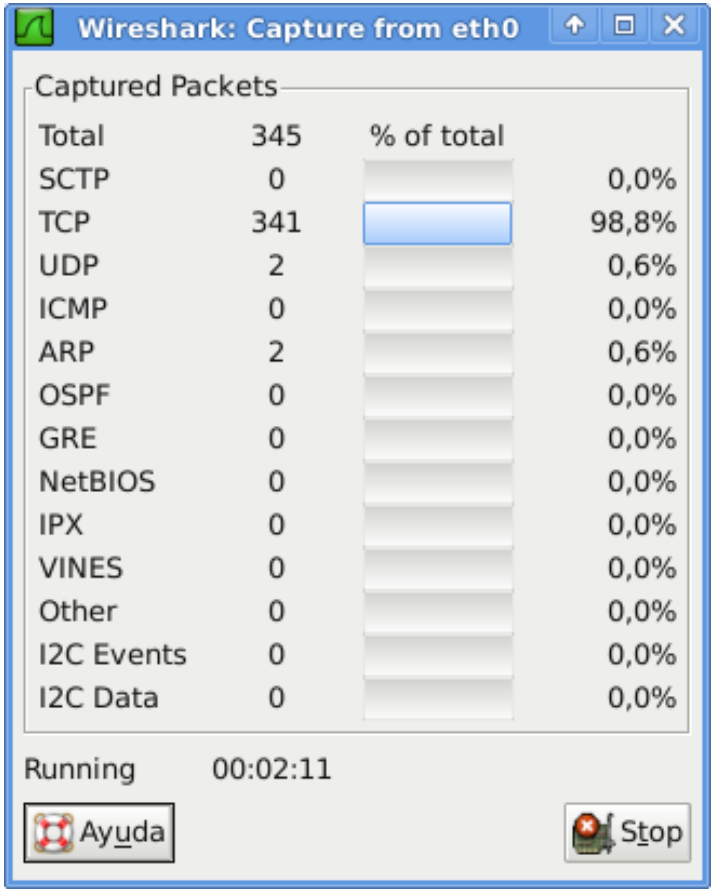
Wireshark: mi primera captura (I)

- Inicie el navegador web
- Inicie Wireshark y seleccione la opción de menú “Capture > Options…” [Ctrl+K]
  - En caso de varias interfaces seleccionar la que se desee “escuchar” o *any* para todas
  - Desmarcar “*Hide capture info dialog*” para ver el resumen de captura al vuelo
  - Pulse Start para iniciar la captura



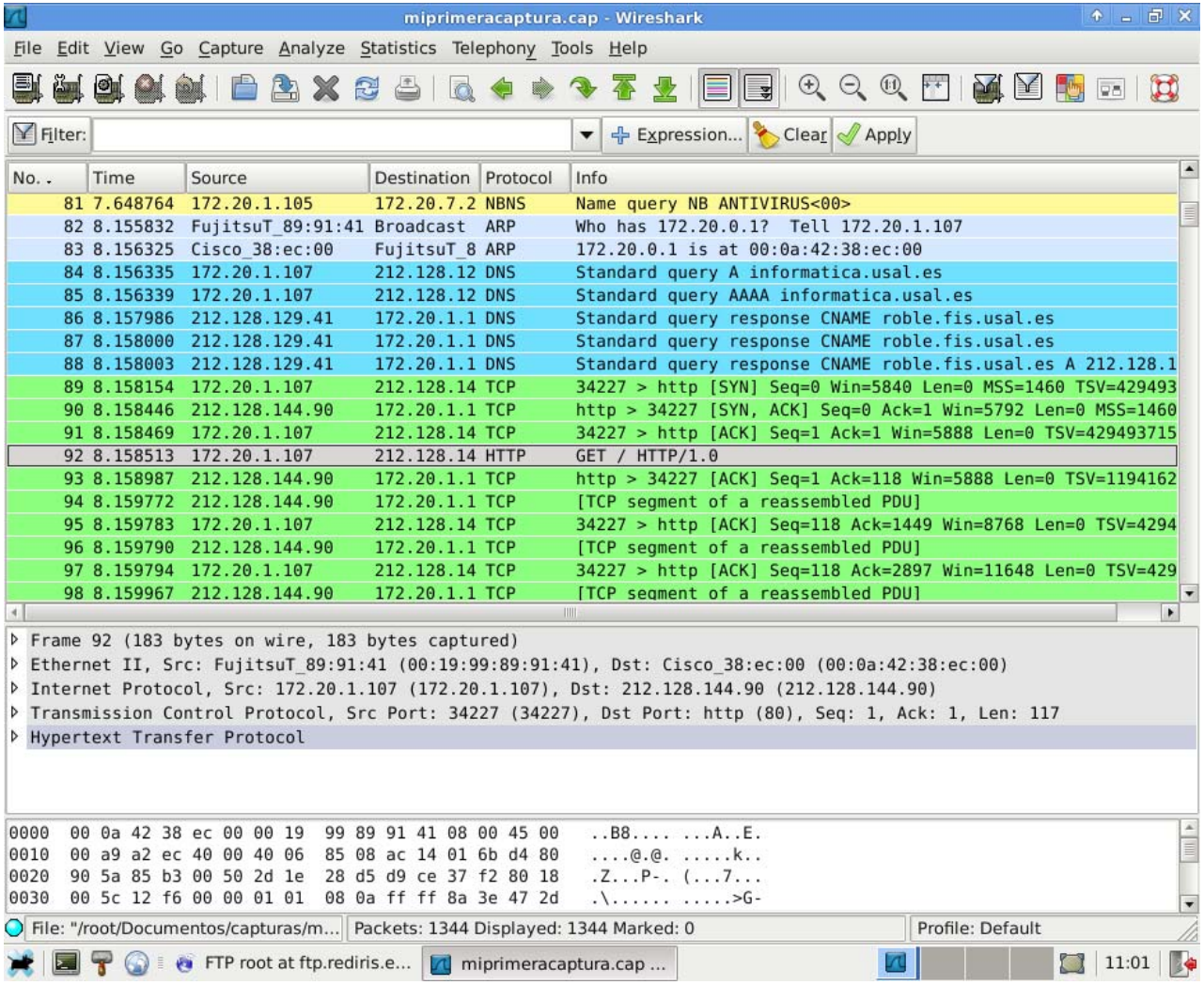
Wireshark: mi primera captura (II)

- Una vez iniciada la captura se muestra una ventana con el resumen de paquetes capturados agrupados por protocolos
- Mientras se está capturando utilice el navegador para visitar las siguientes direcciones:
  - http://informatica.usal.es
  - http://bddoc.csic.es:8080
  - https://moodle.usal.es
  - ftp://ftp.rediris.es
  - gopher://gopher.floodgap.com



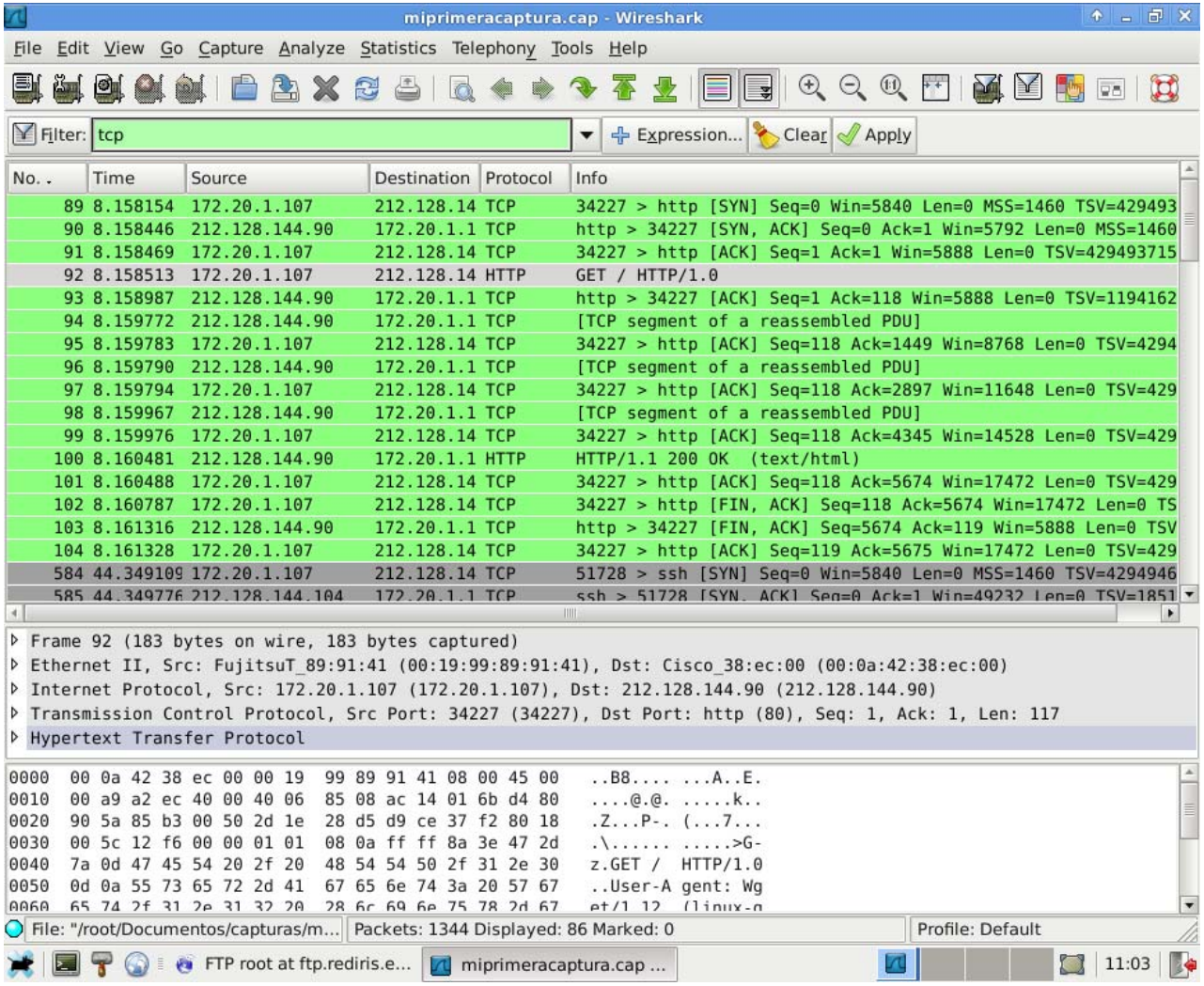
Wireshark: mi primera captura (III)

- Pulse Stop para terminar la captura
  - La apariencia de Wireshark es similar al de la transparencia donde se presentaba la interfaz de usuario
- Guarde la captura desde la opción de menú “File > Save” [Ctrl+S]



Wireshark: mi primera captura (IV)

- Desde la barra de filtros escoja diversos protocolos/filtros predeterminados para seleccionar paquetes de la ventana de captura
  - También se pueden escribir expresiones más complejas
  - El campo de texto del filtro es sensible a lo que se escribe (asesora)
- Intenta identificar el sentido de cada uno de los paquetes filtrados



VNiVERSiDAD D SALAMANCA

Wireshark: Filtros

- Ejemplos de filtros que capturan tráfico determinado
  - Tráfico asociado a una IP
    - ip.addr eq 172.20.1.107
  - Suprimir todo el tráfico del de la red de Windows
    - not nbns
  - Tráfico TCP y UDP pero suprimir el NBNS
    - not nbns and (tcp or udp)
  - Tráfico IPv4
    - ip.version eq 4

VNiVERSiDAD D SALAMANCA

Encapsulación de protocolos de Internet

HTTP	FTP	TELNET	SMTP	POP	NFS (*)	DNS	SNMP
T C P (**)					U D P (**)		
I P							
Ethernet	Token- Ring	FDDI	ATM	Punto a punto	Frame Relay	…	

(*) NFS originalmente diseñado para trabajar con UDP pero últimas versiones soportan TCP

(**) Revisar el fichero /etc/services para una lista completa de protocolos/puertos bien conocidos así como el protocolo de transporte utilizado.

VNiVERSiDAD D SALAMANCA

Referencias

1. Tcpdump: Manual pages. http://www.tcpdump.org/tcpdump_man.html
2. Wireshark: Documentación oficial. http://www.wireshark.org/docs/
3. Wireshark: Wireshark filter syntax and reference .http://www.wireshark.org/docs/man-pages/wireshark-filter.html
4. Wireshark: Tshark, Dump and analyze network traffic. http://www.wireshark.org/docs/man-pages/tshark.html
5. Wireshark: Ejemplos prácticos de capturas de tráfico. http://wiki.wireshark.org/SampleCaptures
6. INTECO: Análisis de tráfico con Wireshark .http://cert.inteco.es/extfrontinteco/img/File/intecocert/EstudiosInformes/cert_inf_seguridad_analisis_trafico_wireshark.pdf
7. Chappell, Laura (2010). *Wireshark Network Analysis (Second Edition): The Official Wireshark Certified Network Analyst Study Guide*. Chappell University.
8. Vyncke, Eric (2008). *LAN Switch Security: What Hackers Know About Your Switches.* Cisco press.