

Introducció al Wireshark



Què és Wireshark?

- Wireshark és un analitzador de paquets i/o protocols de xarxa.
- Un analitzador de paquets de xarxa tractarà de capturar paquets de xarxa i tracta de mostrar-los de la forma més detallada possible.
- Wireshark és potser un dels millors analitzadors de paquets de còdi obert disponibles avui dia per UNIX i Windows.

Analitzador de protocol de xarxa. (Network Protocol Analyzer)

- Intercepta i registra el trànsit que passa per la xarxa, provinents de maquinari o programari (Hardware o software).
- Captura paquets, descodifica i analitza el contingut.
- Un analitzador de xarxa que s'utilitza per:
 - Resolució de problemes a la xarxa.
 - Analitzar el rendiment d'una xarxa per descobrir colls d'ampolla.
 - Xarxa de detecció d'intrusions.
 - Analitzar les operacions d'aplicacions.

Usuaris corrents de Wireshark

- ▶ **Administradors de xarxes:**
 - ▶ Solucionar problemes de xarxa
 - ▶ Examinar problemes de seguretat
- ▶ **Desenvolupadors:**
 - ▶ Debuging de implementacions de los protocols
 - ▶ Aprendre com funcionen els protocols internament

“Wireshark no és un sistema de detecció d'intrusos”

“Wireshark no manipula dispositius dins la xarxa, només
“mesura” les coses de la xarxa”

Més coses de Wireshark

- Es tracta d'una aplicació informàtica, analitzador de paquets de trànsit de xarxa.
- La funcionalitat és molt similar a tcpdump.
- Té una informació GUI front-end i moltes més opcions de filtre (Interfície gràfica).
- "EWeek Labs" l'ha anomenat com una de les aplicacions de Open Source de tots els temps a partir del 2007.

Origens

- Iniciat per Gerald Combs sota el nom de Ethernal.
- La primera versió va ser llançada el 1998.
- El nom de Wireshark es va adoptar el juny de 2006

Característiques

- "Entén" l'estructura dels diferents protocols de xarxa.
- Mostra camps d'encapsulació i interpreta el seu significat.
- Només es pot capturar en les xarxes recolzades per PCAP.
- És multi-plataforma que s'executa en diferents sistemes operatius (Linux, Mac OS X, Microsoft Windows).

WinPcap

- Eina estàndard per a l'accés a la xarxa de capa d'enllaç a l'entorn de finestres; l'aplicació et permet capturar i transmetre paquets de xarxa que passa per la pila de protocols.
- Consisteix en un sistema operatiu conductor dirigit a facilitar l'accés de xarxa de baix nivell.
- Consta de biblioteques per facilitar l'accés a les capes baixes del nivell de xarxa.
- També conté una versió per a Windows de libpcap Unix API

EINES DE MONITORATGE DE XARXES:



20070824-1200.pcap - Wireshark

File Edit View Go Capture Analyze Statistics Help

Filter: Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
76	31.498460	192.168.1.8	67.68.217.48	UDP	Source port: 19946 Destination port: 39832
77	31.756858	67.68.217.48	192.168.1.8	UDP	Source port: 39832 Destination port: 19946
78	37.762916	192.168.1.8	89.16.68.77	UDP	Source port: 19946 Destination port: 39580
79	37.913062	205.85.40.22	192.168.1.8	TCP	https > 49517 [FIN, ACK] Seq=34894 Ack=1012 win=34000 Len=0
80	37.913194	192.168.1.8	205.85.40.22	TCP	49517 > https [ACK] Seq=1012 Ack=34895 win=17680 Len=0
81	38.096530	89.16.68.77	192.168.1.8	UDP	Source port: 39580 Destination port: 19946
82	39.147519	192.168.1.8	76.87.145.159	TCP	49164 > 33629 [PSH, ACK] Seq=0 Ack=0 win=63 Len=2
83	39.940990	76.87.145.159	192.168.1.8	TCP	33629 > 49164 [PSH, ACK] Seq=0 Ack=2 win=65316 Len=2
84	40.135446	192.168.1.8	76.87.145.159	TCP	49164 > 33629 [ACK] Seq=2 Ack=2 win=63 Len=0
85	46.570025	209.85.201.189	192.168.1.8	HTTP	Continuation or non-HTTP traffic
86	46.570322	192.168.1.8	205.85.40.22	TCP	49517 > https [RST, ACK] Seq=1012 Ack=34895 win=0 Len=0
87	46.570598	192.168.1.8	202.171.135.212	TCP	49511 > http [RST, ACK] Seq=0 Ack=1 win=0 Len=0
88	46.765829	192.168.1.8	209.85.201.189	TCP	49505 > http [ACK] Seq=0 Ack=154 win=17424 Len=0
89	54.072635	192.168.1.8	210.65.0.71	TCP	49518 > http [SYN] Seq=0 Len=0 MSS=1460 WS=2
90	54.109770	210.65.0.71	192.168.1.8	TCP	http > 49518 [SYN, ACK] Seq=0 Ack=1 win=24684 Len=0 WS=0 MSS=145
91	54.109996	192.168.1.8	210.65.0.71	TCP	49518 > http [ACK] Seq=1 Ack=1 win=17424 Len=0
92	54.115065	192.168.1.8	210.65.0.71	HTTP	GET /V5/forecast/taiwan/w01.htm HTTP/1.1
93	54.180177	210.65.0.71	192.168.1.8	TCP	http > 49518 [ACK] Seq=1 Ack=606 win=24684 Len=0

Frame 85 (131 bytes on wire, 131 bytes captured)

Ethernet II, Src: 3comEuro_9a:d4:c8 (00:0d:54:9a:d4:c8), Dst: IntelCor_of:d0:8b (00:1b:77:0f:d0:8b)

Internet Protocol, Src: 209.85.201.189 (209.85.201.189), Dst: 192.168.1.8 (192.168.1.8)

Transmission Control Protocol, Src Port: http (80), Dst Port: 49505 (49505), Seq: 77, Ack: 0, Len: 77

Source port: http (80)
Destination port: 49505 (49505)
Sequence number: 77 (relative sequence number)
[Next sequence number: 154 (relative sequence number)]
Acknowledgement number: 0 (relative ack number)
Header length: 20 bytes

```
0000  00 1b 77 0f d0 8b 00 0d 54 9a d4 c8 08 00 45 50  ..w.... T....EP
0010  00 75 63 6c 00 00 2f 06 cb 03 d1 55 c9 bd c0 a8  .ucl../. ...U...
0020  01 08 00 50 c1 61 b5 b5 11 c1 48 b6 c7 09 50 18  ..P.a...H...P.
0030  7f ff 50 c0 00 00 34 37 0d 0a 3c 73 63 72 69 70  .P...47 ..<scrip
0040  74 3e 74 72 79 20 7b 70 61 72 65 6e 74 2e 6d 28  t>try {parent.m(
0050  22 5b 5b 38 30 2c 5b 5c 22 6e 6f 6f 70 5c 22 5d  "[[80,[ \"noop\"
0060  5c 6e 5d 5c 6e 5d 5c 6e 22 29 7d 20 63 61 74 63  \n]\n\" )}] catc
0070  68 28 65 29 20 7b 7d 3c 2f 73 63 72 69 70 74 3e  h(e) {}< /script>
0080  0a 0d 0a                                     ...
```

Transmission Control Protocol (tcp), 20 bytes

P: 15360 D: 15360 M: 0

Wireshark (i WinPcap)

Wireshark

WinPcap – Lib Open Source per la captura de paquets

Sistema Operatiu – Windows & Unix/Linux

Driver de la placa de Xarxa– Ethernet/WiFi Card

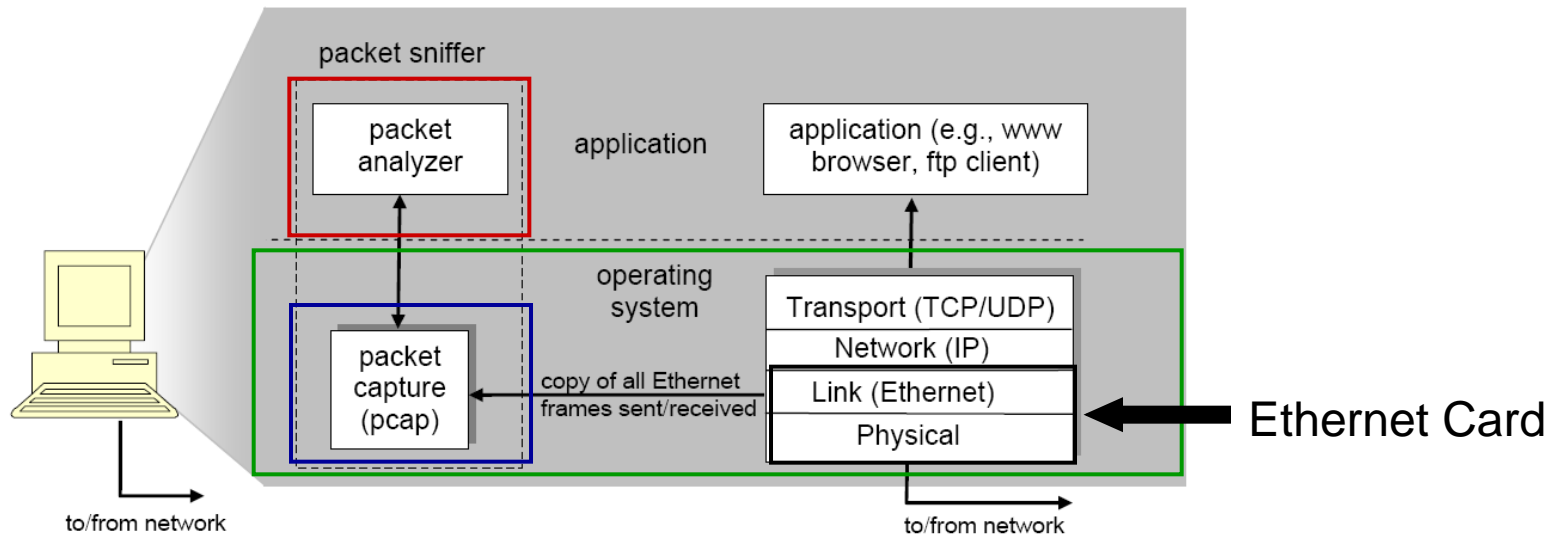


Figure 1: Packet sniffer structure

Instal·lació

- Descarregar des de
 - www.wireshark.org/download.htm
!
- Requerirà instal·lar diferents drivers de captura
 - Windows: winpcap
 - (www.winpcap.org)
 - Linux: libpcap



Esquema de la pantalla de Wireshark

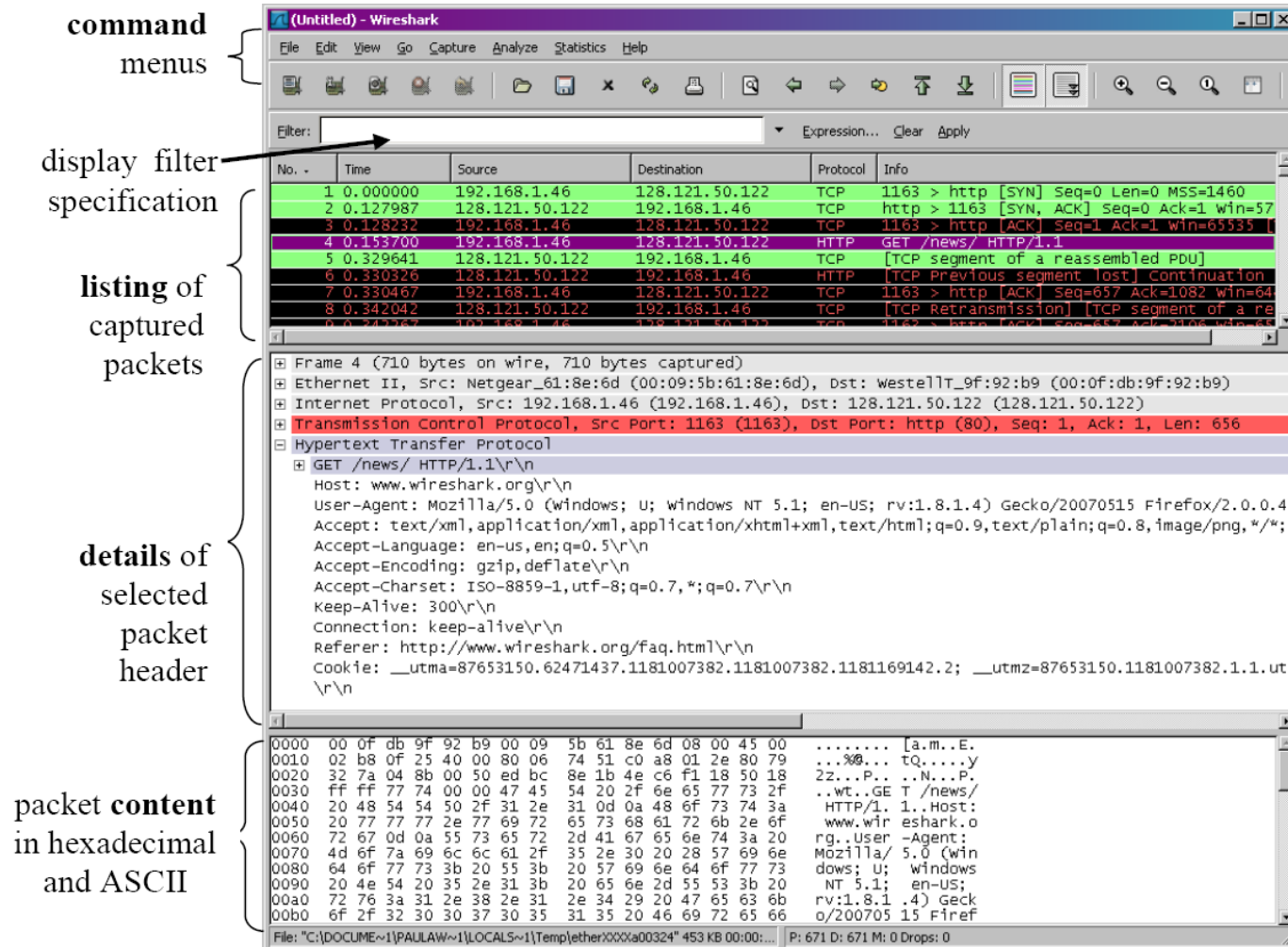
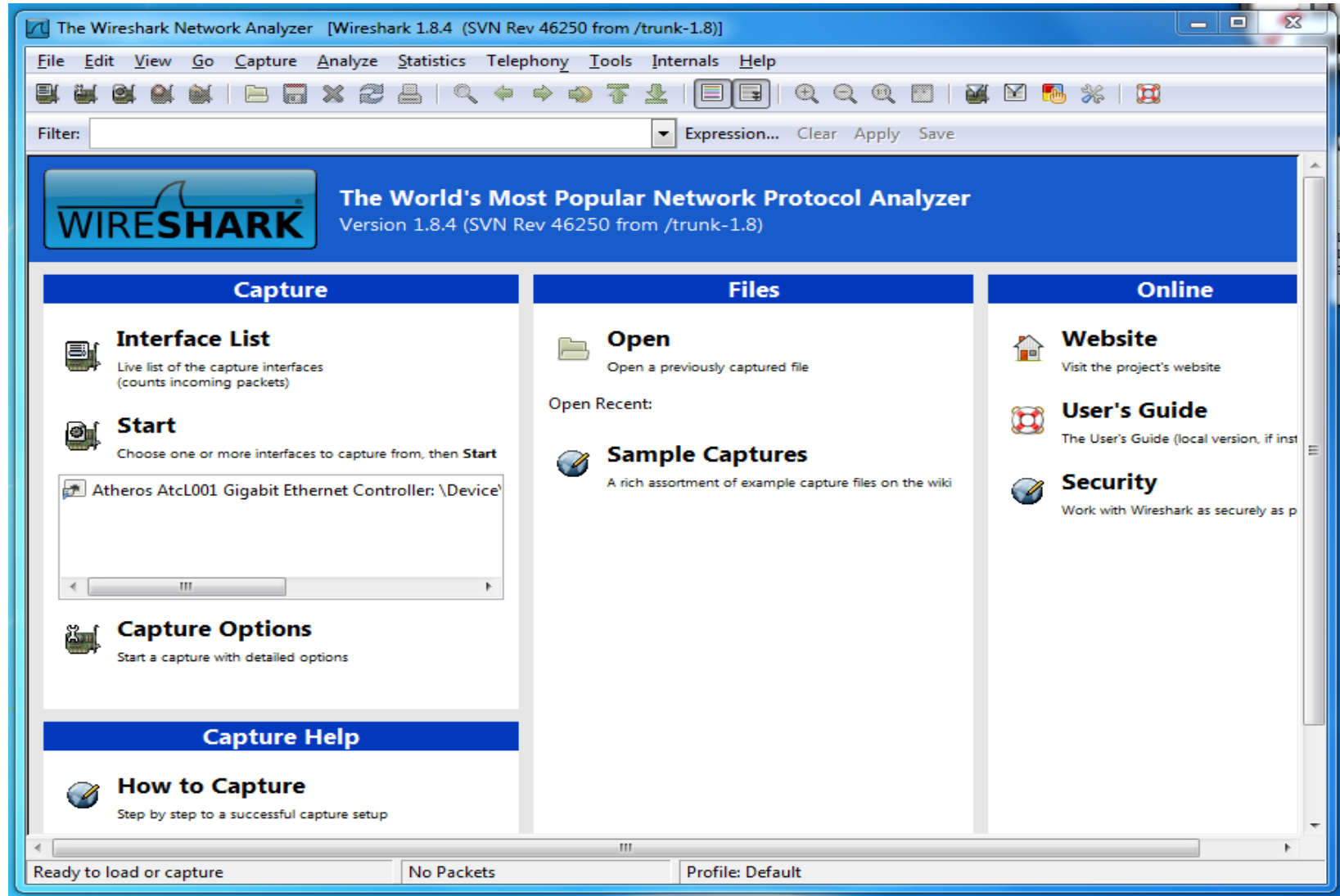


Figure 2: Wireshark Graphical User Interface

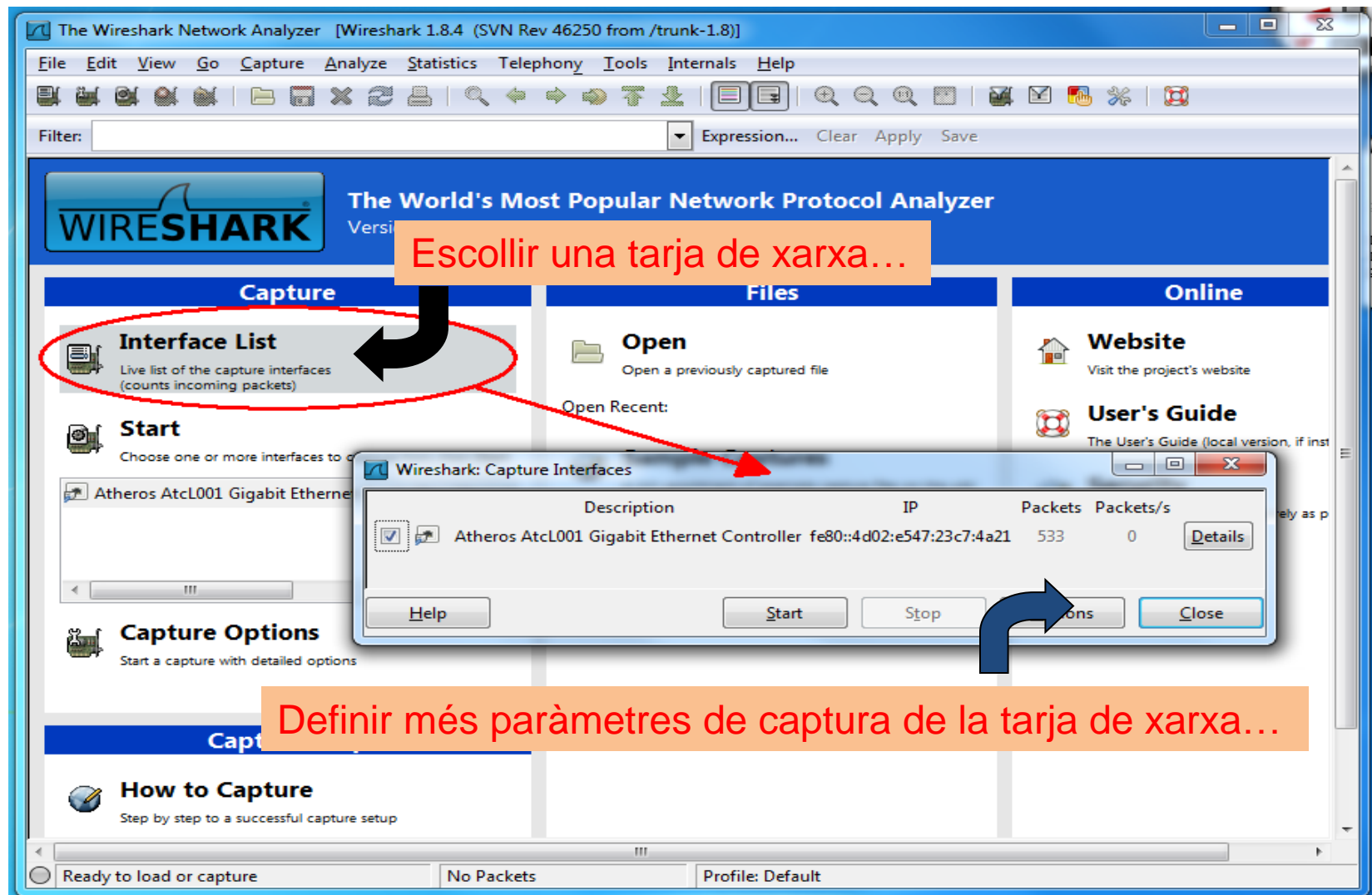
EINES DE MONITORATGE DE XARXES:



Executant Wireshark...



Executant Wireshark...



EINES DE MONITORATGE DE XARXES:

Atheros Atcl001 Gigabit Ethernet Controller: \Device\NPF_{4202B7CC-4678-4D1A-AB8C-35C0AD8C061A} [Wireshark 1.8.4 (SVN Rev 46250 from /trunk-1.8)]

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
308	29.68578500	77.234.44.90	192.168.1.35	TCP	60	80 > 52582 [ACK] Seq=1 Ack=527 win=3752 Len=0
309	29.68753200	77.234.44.90	192.168.1.35	HTTP/DL	210	unknown (0x0a)
310	29.68789600	77.234.44.90	192.168.1.35	TCP	60	80 > 52582 [FIN, ACK] Seq=157 Ack=527 win=3752 Len=0
311	29.68793600	192.168.1.35	77.234.44.90	TCP	54	52582 > 80 [ACK] Seq=527 Ack=158 win=65184 Len=0
312	29.69032500	192.168.1.35	77.234.44.90	TCP	54	52582 > 80 [FIN, ACK] Seq=527 Ack=158 win=65184 Len=0
313	29.82722700	77.234.44.90	192.168.1.35	TCP	60	80 > 52582 [RST] Seq=158 win=0 Len=0
314	30.03472500	fe80::4d02:e547:23c7:ff02::c	ff02::c	SSDP	208	M-SEARCH * HTTP/1.1
315	32.59328300	192.168.1.35	173.194.45.38	TLSv1	91	Application Data
316	32.65547600	173.194.45.38	192.168.1.35	TLSv1	91	Application Data
317	32.85782300	192.168.1.35	173.194.45.38	TCP	54	52490 > 443 [ACK] Seq=527 Ack=158 win=65184 Len=0
318	33.04534900	fe80::4d02:e547:23c7:ff02::c	ff02::c	SSDP	208	M-SEARCH * HTTP/1.1
319	36.04167500	10.187.106.169	224.0.0.1	IGMP	60	Membership Query
320	36.05592000	fe80::4d02:e547:23c7:ff02::c	ff02::c	SSDP	208	M-SEARCH * HTTP/1.1
321	36.47707100	192.168.1.35	224.0.0.252	IGMP	46	Membership Report

Paquete #309: HTTP

Detalls del paquet seleccionat (#309)

Frame 309: 210 bytes on wire (1680 bits), 210 bytes captured (1680 bits) on interface 0

Ethernet II, Src: ZyxeCom_7c:36:0a (50:67:f0:7c:36:0a), Dst: AsustekC_43:16:18 (00:1f:c6:43:16:18)

Internet Protocol Version 4, Src: 77.234.44.90 (77.234.44.90), Dst: 192.168.1.35 (192.168.1.35)

Transmission Control Protocol, Src Port: 80 (80), Dst Port: 52582 (52582), Seq: 1, Ack: 527, Len: 156

Hypertext Transfer Protocol

SIP/NOE Protocol, unknown (0x0a)

0000 00 1f c6 43 16 18 50 67 f0 7c 36 0a 08 00 45 00 ...C..Pg .|6...E.
0010 00 c4 60 bc 40 00 33 06 aa 68 4d ea 2c 5a c0 a8 ...@.3. .hm.,Z..
0020 01 23 00 50 cd 66 39 8d 2b f9 6b 3f 57 da 50 18 ...#.P.f9. +.k?W.P.
0030 0e a8 1c 9f 00 00 48 54 54 50 2f 31 2e 31 20 32 ...HT TP/1.1 2
0040 30 30 20 4f 4b 0d 0a 43 6f 6e 74 65 6e 74 2d 54 00 OK..C ontent-T
0050 79 70 65 3a 20 61 70 70 6c 69 63 61 74 69 6f 6e ype: app lication
0060 2f 6f 63 74 65 74 2d 73 74 72 65 61 6d 0d 0a 43 /octet-s tream..C
0070 6f 6e 74 65 6e 74 2d 4c 65 6e 67 74 68 3a 20 32 ontent-L ength: 2
0080 34 0d 0a 53 65 74 2d 43 6f 6f 6b 69 65 3a 20 75 4..Set-C ookie: u
0090 73 65 72 69 64 3d 61 35 36 66 30 66 62 61 65 31 serid=a5 6f0fbae1
00a0 62 62 39 33 34 34 63 36 66 65 36 31 63 32 32 37 bb9344c6 fe61c227
00b0 65 36 33 35 66 36 0d 0a 0d 0a 0a 16 0a 0b 08 ba e635f6..
00c0 01 10 c8 01 18 a0 38 20 00 12 07 08 02 10 02 188
00d0 d8 04 ..

File: "C:\Users\xavier\AppData\Local\Temp\... Packets: 321 Displayed: 321 Marked: 0 Dropped: 0 Profile: Default

EINES DE MONITORATGE DE XARXES:

Wireshark 1.8.4 (SVN Rev 46250 from /trunk-1.8)]

Filter: `http.request.method` Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
19	3.010591000	fe80::4d02:e547:23c7::ff02::c	fe80::4d02:e547:23c7::ff02::c	SSDP	208	M-SEARCH * HTTP/1.1
20	6.021368000	fe80::4d02:e547:23c7::ff02::c	fe80::4d02:e547:23c7::ff02::c	SSDP	208	M-SEARCH * HTTP/1.1
29	10.015018000	fe80::4d02:e547:23c7::ff02::c	fe80::4d02:e547:23c7::ff02::c	HTTP	100	GET / HTTP/1.1
32	13.025693000	fe80::4d02:e547:23c7::ff02::c	fe80::4d02:e547:23c7::ff02::c	HTTP	100	GET / HTTP/1.1
45	16.036454000	fe80::4d02:e547:23c7::ff02::c	fe80::4d02:e547:23c7::ff02::c	HTTP	100	GET / HTTP/1.1
51	20.030520000	fe80::4d02:e547:23c7::ff02::c	fe80::4d02:e547:23c7::ff02::c	HTTP	100	GET / HTTP/1.1
63	22.315988000	fe80::4d02:e547:23c7::ff02::c	fe80::4d02:e547:23c7::ff02::c	HTTP	100	GET / HTTP/1.1
122	23.041580000	fe80::4d02:e547:23c7::ff02::c	fe80::4d02:e547:23c7::ff02::c	HTTP	100	GET / HTTP/1.1
271	26.041998000	fe80::4d02:e547:23c7::ff02::c	fe80::4d02:e547:23c7::ff02::c	HTTP	100	GET / HTTP/1.1
275	26.323225000	fe80::4d02:e547:23c7::ff02::c	fe80::4d02:e547:23c7::ff02::c	HTTP	100	GET / HTTP/1.1
307	29.539940000	fe80::4d02:e547:23c7::ff02::c	fe80::4d02:e547:23c7::ff02::c	HTTP	580	GET / HTTP/1.1
314	30.034725000	fe80::4d02:e547:23c7::ff02::c	fe80::4d02:e547:23c7::ff02::c	HTTP	100	GET / HTTP/1.1
318	33.045349000	fe80::4d02:e547:23c7::ff02::c	fe80::4d02:e547:23c7::ff02::c	HTTP	100	GET / HTTP/1.1
320	36.055920000	fe80::4d02:e547:23c7::ff02::c	fe80::4d02:e547:23c7::ff02::c	HTTP	100	GET / HTTP/1.1

Wireshark: Filter Expression - Profile: Default

Field name

- HSR_PRP_SUPERVISION - HSR/PRP Supervision (HSR/PRP)
- HSRP - Cisco Hot Standby Router Protocol
- HTTP - Hypertext Transfer Protocol
 - http.notification - Notification (TRUE if HTTP notification)
 - http.response - Response (TRUE if HTTP response)
 - http.request - Request (TRUE if HTTP request)
 - http.authbasic - Credentials
 - http.request.method - Request Method (HTTP Request Method)
 - http.request.uri - Request URI (HTTP Request-URI)
 - http.request.version - Request Version (HTTP Request Version)
 - http.request.full_uri - Full request URI (The full request URI)
 - http.response.code - Status Code (HTTP Response Status Code)
 - http.response.phrase - Response Phrase (HTTP Response Phrase)

Relation

- is present
- ==
- !=
- >
- <
- >=
- <=
- contains
- matches

OK Cancel

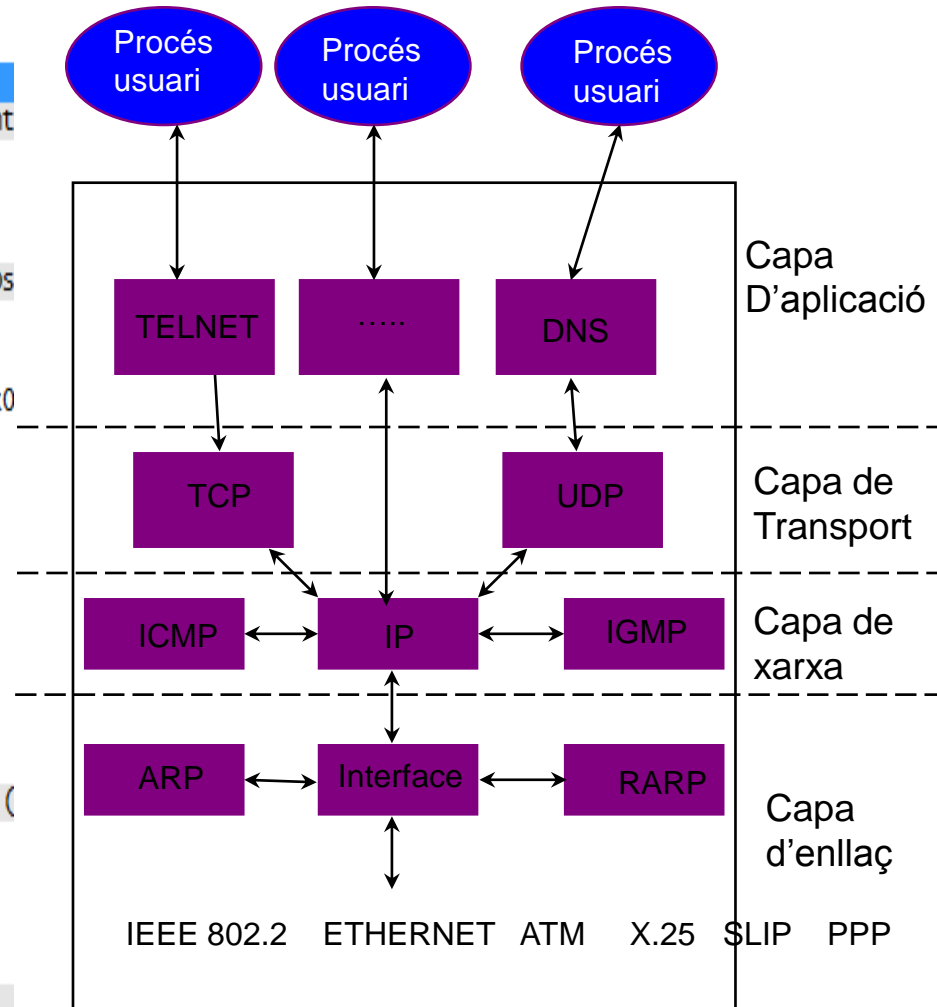
Filtrant paquets HTTP només

File: "C:\Users\xavier\AppData\Local\Temp\..." Packets: 321 Displayed: 15 Marked: 0 Dropped: 0 Profile: Default

Capes TCP/IP observables amb Wireshark

Frame 6: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)

- Ethernet II, Src: AsustekC_b1:0c:ad (00:e0:18:b1:0c:ad), Dst: Quant
 - Destination: QuantaCo_32:41:8c (00:c0:9f:32:41:8c)
 - Source: AsustekC_b1:0c:ad (00:e0:18:b1:0c:ad)
 - Type: IP (0x0800)
- Internet Protocol Version 4, Src: 192.168.170.8 (192.168.170.8), Ds
 - Version: 4
 - Header length: 20 bytes
 - Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x0)
 - Total Length: 60
 - Identification: 0x6f4c (28492)
 - Flags: 0x02 (Don't Fragment)
 - Fragment offset: 0
 - Time to live: 64
 - Protocol: UDP (17)
 - Header checksum: 0xf5f6 [correct]
 - Source: 192.168.170.8 (192.168.170.8)
 - Destination: 192.168.170.20 (192.168.170.20)
- User Datagram Protocol, Src Port: 32795 (32795), Dst Port: domain (
 - Source port: 32795 (32795)
 - Destination port: domain (53)
 - Length: 40
 - Checksum: 0x3432 [validation disabled]
- Domain Name System (query)



TCP/IP

El que no pot fer...

- No es pot utilitzar per traçar el mapa d'una xarxa.
- No és una eina que generi dades de la xarxa en mode passiu.
- Només mostra informació detallada sobre els protocols que pot entendre.
- Només pot capturar dades, com el tipus de sistema operatiu \ Interfícies \ i informació dels controladors (drivers).
- Un exemple d'això és la captura de dades a través de xarxes sense fils.

Conclusió

- Les funcions d'anàlisi sense fils (wireless) de Wireshark s'han convertit en una eina molt poderosa per a la solució de problemes i anàlisi de xarxes sense fils.
- Amb els filtres de pantalla de Wireshark i potents funcions, es poden “disseccionar” els protocols, es pot realitzar a través de grans quantitats de trànsit de paquets.
- Sense cap dubte, Wireshark és una eina d'avaluació de gran abast i una eina d'anàlisi de xarxes sense fil que ha de ser una part de cada auditor, enginyer, consultor i kit d'eines.