



**POLITECNICO**  
MILANO 1863



# Introduction to Wireshark

**Francesco Musumeci**

# Wireshark Lab

- It is useful to be able to “view the protocols in action” and be able to “play with the protocols”
- To do this, the ideal is a message manipulation tool (Protocol Data Unit - PDU)
- With Wireshark it is possible to observe the network protocols in action on your computer, interact and exchange messages with the protocol entities running both on the local PC and somewhere else on the Internet



# Packet sniffer

- The basic tool of Wireshark captures (sniffs) messages received and transmitted from your PC
- It is an entirely passive module
  - It cannot send packets on its own
  - The packets it receives are not explicitly addressed to the sniffer (it is “transparent” to protocols)
  - It does not change the action of the protocols: the packets it intercepts are copied (dumped), but regularly delivered to the true destination

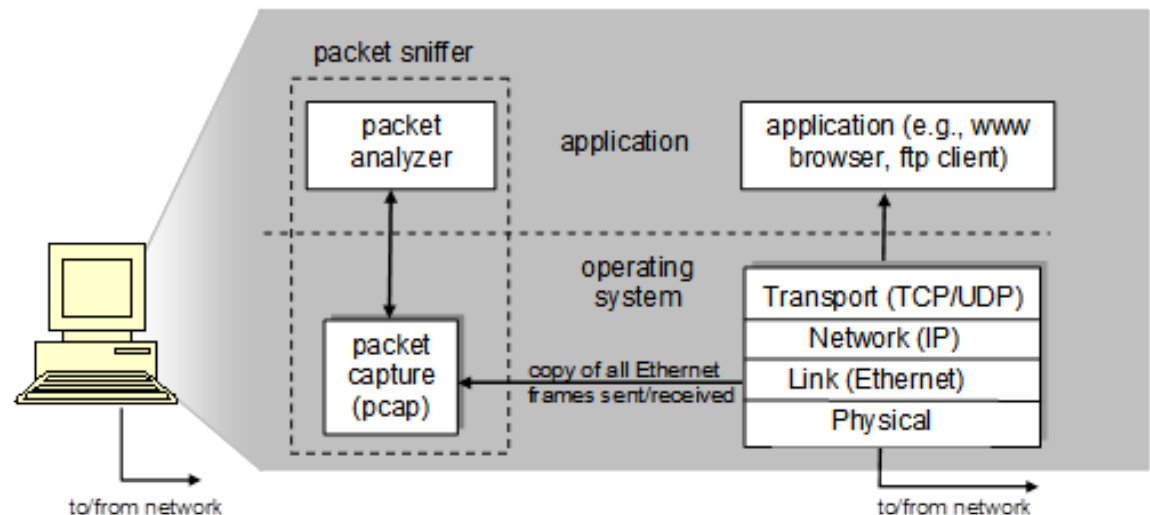


Figure 1: Packet sniffer structure

# Packet sniffer

- It consists of two parts
- Packet capture library
  - It receives a copy of all the data-link frames received/transmitted by the PC
- Packet analyzer
  - It is able to recognize the structure of protocol messages
  - It shows the contents of all fields of a captured protocol message

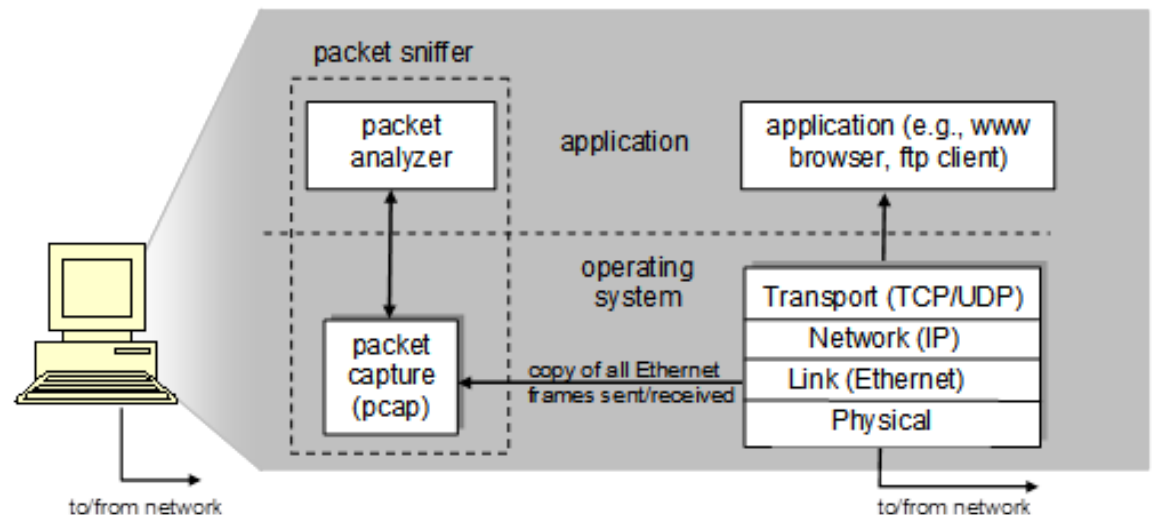


Figure 1: Packet sniffer structure

# Message analysis

- The Packet Analyzer exploits the recursive encapsulation carried out by the protocol levels
- It allows you to analyze a message captured at multiple protocol levels (onion skin approach)
- Example
  - Ethernet frame
  - IP packet
  - TCP segment
  - HTTP message
  - Content of the HTTP message

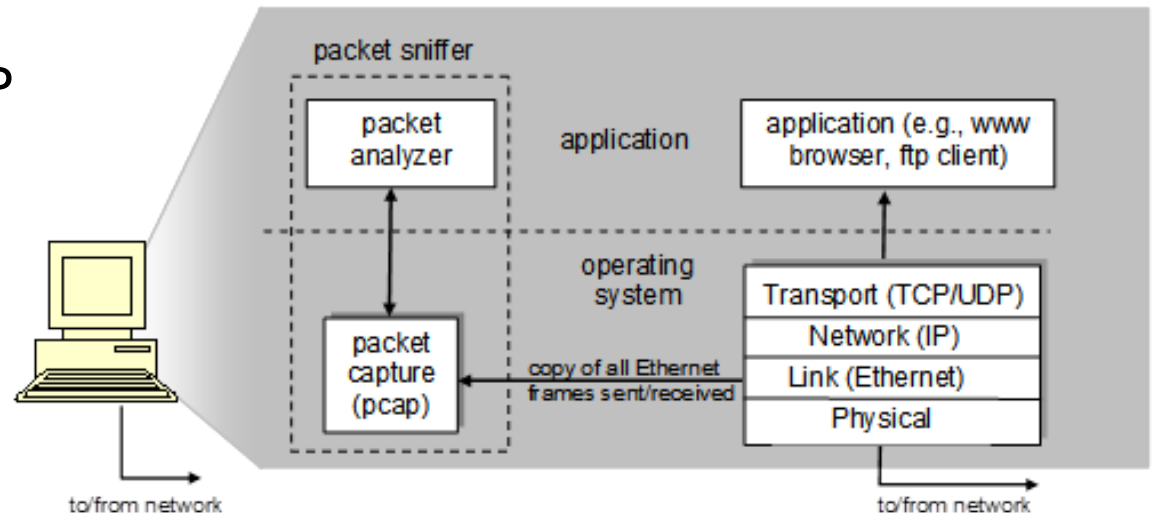


Figure 1: Packet sniffer structure

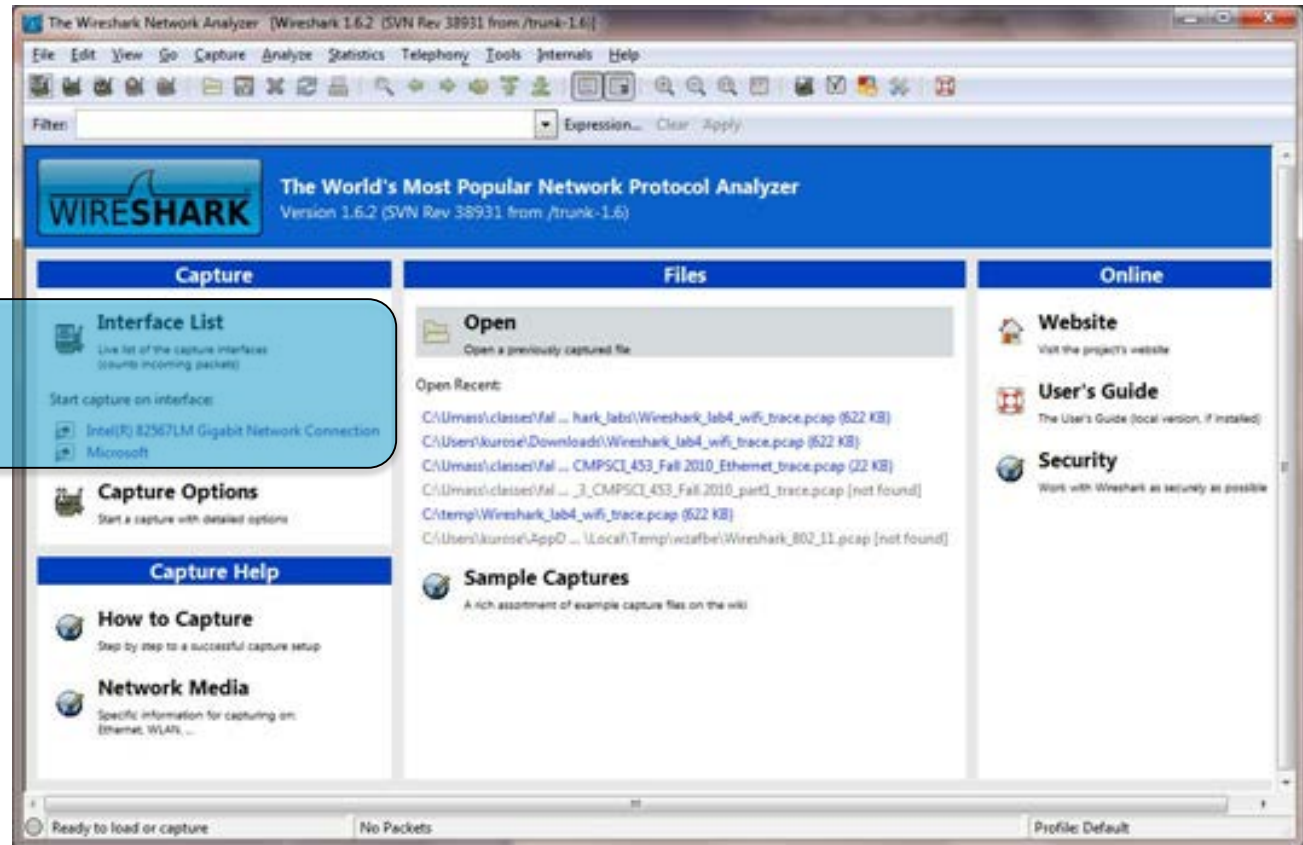
# Wireshark

- Most popular open-source free-ware sniffer program
  - <http://www.wireshark.org/>
- It uses the *Packet Capture Library* available in the operating system
  - Windows (WinPCap)
  - Linux/Unix (libpcap)
  - Mac (libpcap)
- Well documented and with a large community of users
  - User guide: [http://www.wireshark.org/docs/wsug\\_html\\_chunked/](http://www.wireshark.org/docs/wsug_html_chunked/)
  - Man pages: <http://www.wireshark.org/docs/man-pages/>
  - Detailed FAQ: <http://www.wireshark.org/faq.html>
- The analyzer supports the formats of hundreds of protocols
- It can operate on many different network interfaces including: Ethernet, PPP and SLIP, 802.11 WiFi



# Wireshark

- Installation (<http://www.wireshark.org/download.html>)
  - The necessary Lan Packet Capture Library is automatically installed in the OS if not already present
- Startup screen (Win):



PC interface list.  
Clicking twice on  
one, the capture  
starts



# Capture display

- Command menu
  - File: to save or open captures
  - Capture: to start capturing
  - ...
- Packet List
  - Packet number (assigned by Ws), capture time, source/destination addresses, protocol, specific fields
  - It can be ordered
  - Protocol type shows the highest level that caused the sending

The screenshot shows the Wireshark interface with a packet capture list and details of a selected packet. The packet list shows several packets, with packet 4 selected. The details pane shows the structure of the selected packet, including Ethernet II, Internet Protocol, and Hypertext Transfer Protocol. The packet content is displayed in hexadecimal and ASCII.

**command menus**

**display filter specification**

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.1.46	128.121.50.122	TCP	1163 > http [SYN] Seq=0 Len=0 MSS=1460
2	0.127987	128.121.50.122	192.168.1.46	TCP	http > 1163 [SYN, ACK] Seq=0 Ack=1 win=57
3	0.128232	192.168.1.46	128.121.50.122	TCP	1163 > http [ACK] Seq=1 Ack=1 win=5553
4	0.153700	192.168.1.46	128.121.50.122	HTTP	GET /news/ HTTP/1.1
5	0.329641	128.121.50.122	192.168.1.46	TCP	[TCP segment of a reassembled PDU]
6	0.330376	128.121.50.122	192.168.1.46	HTTP	[TCP Previous segment lost] Connection
7	0.330467	192.168.1.46	128.121.50.122	TCP	1163 > http [ACK] Seq=517 Ack=1082 win=64
8	0.342042	128.121.50.122	192.168.1.46	TCP	[TCP Retransmission] TCP segment of a reassembled PDU

**listing of captured packets**

**details of selected packet header**

```
Frame 4 (710 bytes on wire (710 bytes captured) on interface eth0)
  Ethernet II, Src: Netgear_61:8e:6d (00:09:5b:61:8e:6d), Dst: Westell_9f:92:b9 (00:0f:db:9f:92:b9)
  Internet Protocol, Src: 192.168.1.46 (192.168.1.46), Dst: 128.121.50.122 (128.121.50.122)
  Transmission Control Protocol, Src Port: 1163 (1163), Dst Port: http (80), Seq: 1, Ack: 1, Len: 858
  Hypertext Transfer Protocol
  GET /news/ HTTP/1.1\r\n
  Host: www.wireshark.org\r\n
  user-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.1.4) Gecko/20070515 Firefox/2.0.0.4\r\n
  Accept: text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5\r\n
  Accept-Language: en-us,en;q=0.5\r\n
  Accept-Encoding: gzip,deflate\r\n
  Accept-Charset: iso-8859-1,utf-8;q=0.7,*;q=0.7\r\n
  keep-alive: 300\r\n
  connection: keep-alive\r\n
  Referer: http://www.wireshark.org/faq.html\r\n
  cookie: __utma=87653150.62471437.1181007382.1181007382.1181169142.2; __utmz=87653150.1181007382.1.1.1. utr\r\n
  \r\n
```

**packet content in hexadecimal and ASCII**

```
0000 00 0f db 9f 92 b9 00 09 5b 61 8e 6d 08 00 45 00 ..... [a.m..e.
0010 02 b8 0f 25 40 00 80 06 74 51 c0 a8 01 2e 80 79 ...NB...TQ....y
0020 32 7a 04 8b 00 50 ed bc 8e 1b 4e c6 f1 18 50 18 2z...P...N...P.
0030 ff ff 77 74 00 00 47 45 54 20 2f 6e 65 77 73 2f ...wt..GET /news/
0040 20 48 54 54 50 2f 31 2e 31 0d 04 48 6f 73 74 3a HTTP/1.1..Host:
0050 30 77 77 77 2e 77 69 72 65 73 68 61 72 6b 2e 6f www.wireshark.o
0060 72 67 0d 0a 55 73 65 72 2d 41 67 65 6e 74 3a 20 rg..user -Agent:
0070 4d 6f 7a 69 6c 6c 61 2f 35 2e 30 20 28 57 69 6e Mozilla/ 5.0 (win
0080 64 6f 77 73 3b 20 55 3b 20 57 69 6e 64 6f 77 73 dows; U; windows
0090 20 4e 54 20 35 2e 31 3b 20 65 6e 2d 55 53 3b 20 NT 5.1; en-us;
00a0 72 76 3a 31 2e 38 2e 31 2e 34 29 20 47 65 63 6b rv:1.8.1.4) Gecko
00b0 6f 2f 32 30 30 37 30 35 31 35 20 46 69 72 65 66 o/200705 15 Firef
```





# Capture display

- Header details
  - Details on the selected packet
  - Info about the frame, IP packet, and UDP/TCP segment that contain the message
    - They can be expanded or compressed
  - Info on high level protocol
  - Packet contents
    - In ASCII and hexadecimal format
- Filter fields
  - To select or hide packets based on various filtering criteria

The screenshot shows the Wireshark interface with a packet capture list and details pane. The packet list shows several packets, with packet 4 selected. The details pane shows the structure of the selected packet, including Ethernet II, Internet Protocol, Transmission Control Protocol, and Hypertext Transfer Protocol. The packet content is displayed in both hexadecimal and ASCII formats.

**command menus**

**display filter specification**

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.1.46	128.121.50.122	TCP	1163 > http [SYN] Seq=0 Len=0 MSS=1460
2	0.127987	128.121.50.122	192.168.1.46	TCP	http > 1163 [SYN, ACK] Seq=0 Ack=1 win=57
3	0.128232	192.168.1.46	128.121.50.122	TCP	1163 > http [ACK] Seq=1 Ack=1 win=65535
4	0.133700	192.168.1.46	128.121.50.122	HTTP	GET /news/ HTTP/1.1
5	0.329641	128.121.50.122	192.168.1.46	TCP	[TCP segment of a reassembled PDU]
6	0.330376	128.121.50.122	192.168.1.46	HTTP	[TCP previous segment lost] Connection
7	0.330467	192.168.1.46	128.121.50.122	TCP	1163 > http [ACK] Seq=637 Ack=1026 win=64
8	0.342042	128.121.50.122	192.168.1.46	TCP	[TCP retransmission] TCP segment of a reassembled PDU

**listing of captured packets**

**details of selected packet header**

```
Frame 4 (710 bytes on wire (710 bytes captured) on interface eth0)
  Ethernet II, Src: Netgear_g1:8e:6d (00:09:5b:61:8e:6d), Dst: Westell_r9f:92:b9 (00:0f:db:9f:92:b9)
  Internet Protocol, Src: 192.168.1.46 (192.168.1.46), Dst: 128.121.50.122 (128.121.50.122)
  Transmission Control Protocol, Src Port: 1163 (1163), Dst Port: http (80), Seq: 1, Ack: 1, Len: 858
  Hypertext Transfer Protocol
  GET /news/ HTTP/1.1\r\n
  Host: www.wireshark.org\r\n
  user-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.1.4) Gecko/20070515 Firefox/2.0.0.4\r\n
  Accept: text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5\r\n
  Accept-Language: en-us,en;q=0.5\r\n
  Accept-Encoding: gzip,deflate\r\n
  Accept-Charset: iso-8859-1,utf-8;q=0.7,*;q=0.7\r\n
  Keep-Alive: 300\r\n
  Connection: keep-alive\r\n
  Referer: http://www.wireshark.org/faq.html\r\n
  cookie: __utma=87653150.62471437.1181007382.1181007382.1181169142.2; __utmz=87653150.1181007382.1.1.1. utr\r\n
  \r\n
```

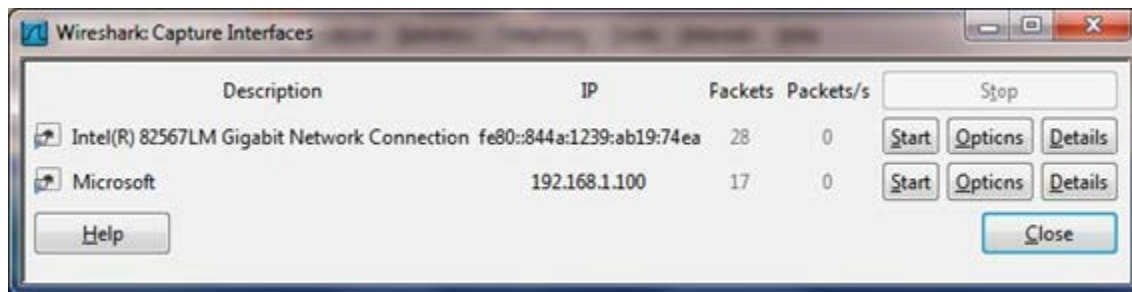
**packet content in hexadecimal and ASCII**

```
0000 00 0f db 9f 92 b9 00 09 5b 61 8e 6d 08 00 45 00 ..... [a.m.e.
0010 02 b8 0f 25 40 00 80 06 74 51 c0 a8 01 2e 80 79 ...NB...TQ....y
0020 32 7a 04 8b 00 50 ed bc 8e 1b 4e c6 f1 18 50 18 2z...P...N...P.
0030 ff ff 77 74 00 00 47 45 54 20 2f 6e 65 77 73 2f ...wt..GET /news/
0040 20 48 54 54 50 2f 31 2e 31 0d 04 48 6f 73 74 3a HTTP/1.1..Host:
0050 20 77 77 77 2e 77 69 72 65 73 68 61 72 6b 2e 6f www.wireshark.o
0060 72 67 0d 0a 55 73 65 72 2d 41 67 65 6e 74 3a 20 rg..user-Agent:
0070 4d 6f 7a 69 6c 6c 61 2f 35 2e 30 20 28 57 69 6e Mozilla/5.0 (win
0080 64 6f 77 73 3b 20 55 3b 20 57 69 6e 64 6f 77 73 dows; U; windows
0090 20 4e 54 20 35 2e 31 3b 20 65 6e 2d 55 53 3b 20 NT 5.1; en-US;
00a0 72 76 3a 31 2e 38 2e 31 2e 34 29 20 47 65 63 6b rv:1.8.1.4) Geck
00b0 6f 2f 32 30 30 37 30 35 31 35 20 46 69 72 65 66 o/200705 15 Firef
```



# Exercise: test run

- Open a browser
- Start Ws
- Menu: Capture → Interface
  - Start on the interface you want to use
- In the browser, enter the URL
- <http://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html>
  - After the INTRO-wireshark-file1.html page appears in the browser, stop the Ws capture



# Exercise: test run

- Write “http” in the filter window
- Look for the response to the HTTP GET message that contains the phrase that is displayed in the browser

The screenshot displays the Wireshark interface with the filter 'http' applied. The packet list pane shows the following entries:

No.	Time	Source	Destination	Protocol	Length	Info
813	43.946687	192.168.1.101	66.103.80.47	HTTP	181	GET /cgi-bin/alive?0001088 HTTP/1.1
816	43.996668	66.103.80.47	192.168.1.101	HTTP	60	HTTP/1.1 200 OK (text/plain)
826	44.497577	192.168.1.101	204.9.163.166	HTTP	333	POST /api/v1.0/pnr?language=en&plugin=...
828	44.507171	204.9.163.166	192.168.1.101	HTTP	271	HTTP/1.1 200 OK
835	45.629833	192.168.1.101	128.119.245.12	HTTP	489	GET /wireshark-labs/INTRO-wireshark-fil...
837	45.646802	128.119.245.12	192.168.1.101	HTTP	434	HTTP/1.1 200 OK (text/html)
838	45.670226	192.168.1.101	128.119.245.12	HTTP	429	GET /favicon.ico HTTP/1.1
839	45.687572	128.119.245.12	192.168.1.101	HTTP	564	HTTP/1.1 404 Not Found (text/html)
840	45.724273	192.168.1.101	128.119.245.12	HTTP	459	GET /favicon.ico HTTP/1.1
841	45.739188	128.119.245.12	192.168.1.101	HTTP	564	HTTP/1.1 404 Not Found (text/html)
847	48.670194	192.168.1.101	128.119.245.12	HTTP	459	GET /favicon.ico HTTP/1.1
848	48.689680	128.119.245.12	192.168.1.101	HTTP	564	HTTP/1.1 404 Not Found (text/html)

The selected packet (835) details are as follows:

- Frame 835: 489 bytes on wire (3912 bits), 489 bytes captured (3912 bits)
- Ethernet II, Src: MonMailPr\_0d:ca:8f (00:22:68:0d:ca:8f), Dst: Cisco-Li\_45:1f:1b (00:22:6b:45:1f:1b)
- Internet Protocol Version 4, Src: 192.168.1.101 (192.168.1.101), Dst: 128.119.245.12 (128.119.245.12)
- Transmission Control Protocol, Src Port: 57522 (57522), Dst Port: http (80), Seq: 1, Ack: 1, Len: 435
- Hypertext Transfer Protocol
  - GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1\r\n
  - Host: gaia.cs.umass.edu\r\n
  - User-Agent: Mozilla/5.0 (windows; u; windows NT 6.1; en-US; rv:1.9.2.22) Gecko/20110902 Firefox/3.6.22 (.NET CLR 3.5.30729)\r\n
  - Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8\r\n
  - Accept-Language: en-us,en;q=0.5\r\n
  - Accept-Encoding: gzip,deflate\r\n
  - Accept-Charset: ISO-8859-1,utf-8;q=0.7,\*;q=0.7\r\n
  - Keep-Alive: 115\r\n
  - Connection: keep-alive\r\n
  - \r\n
  - \r\n
  - [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html]

