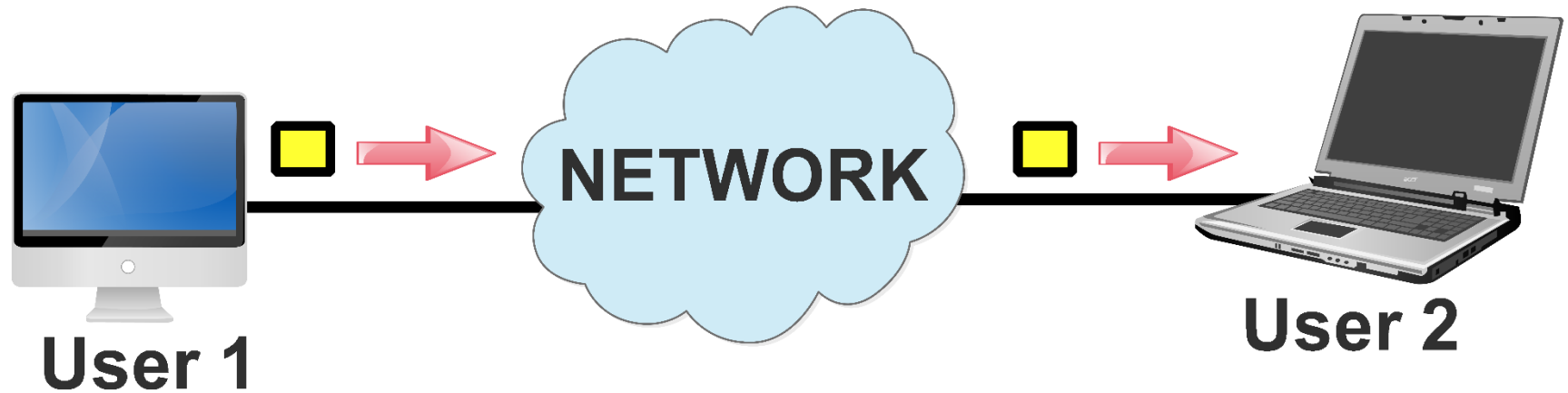


# Intro to basic network tools

Lakiotakis Emmanouil

HY335B – Computer Networks  
Spring 2017

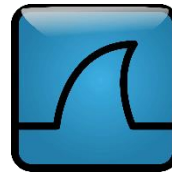
# Intro to networks



**Network:** a group of two or more computer systems linked together

# Packet sniffers

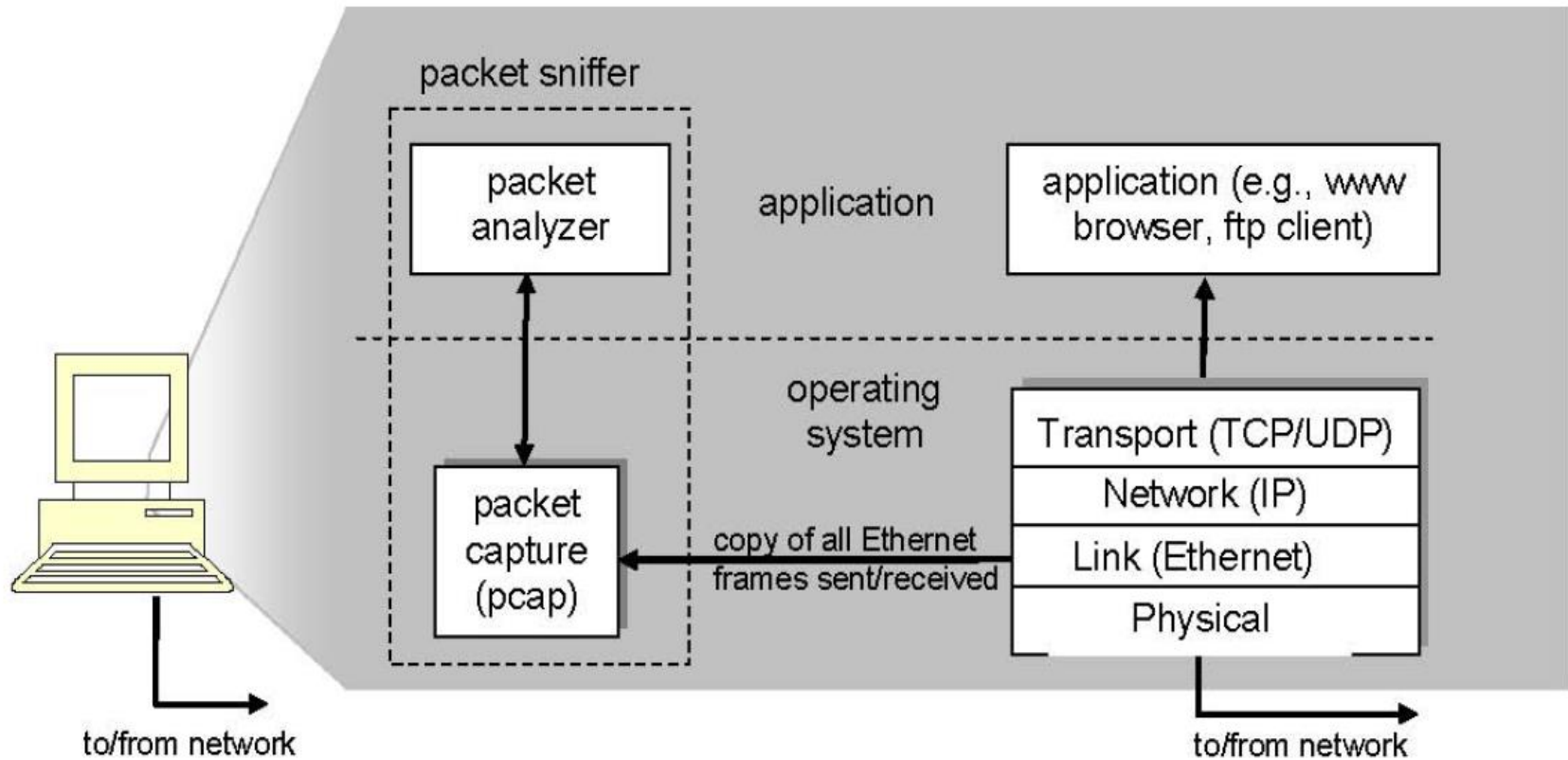
- Packet sniffer: tool used for capturing network packets
- Well-known tools:
  - Wireshark
  - tcpdump
  - Ettercap



# Wireshark

- Passive operation
- Consists of 2 major components
  - Packet capture library
  - Packet analyzer
- Available for all operating systems (Windows, UNIX, MAC OSX)
- Webpage: <https://www.wireshark.org/>

# Basic operation



# Graphical User Interface

Command  
menus

Display  
filter

Packet-  
listing

Packet-  
header  
details

Packet-  
contents

The image shows the Wireshark graphical user interface. At the top is a menu bar with options: File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, and Help. Below the menu is a toolbar with various icons for file operations, capture control, and analysis. A display filter bar is present with the text 'Apply a display filter ... <Ctrl-/>'. The main packet list pane displays a table of captured packets. The table has columns for No., Time, Source, Destination, Protocol, Length, and Info. The packets are color-coded: orange for ARP, green for SSDP, blue for UDP, and yellow for VRRP. Below the packet list is the packet details pane, which shows the structure of the selected packet (Frame 468). It includes fields like Ethernet II, Src, Dst, and Address Resolution Protocol (request). At the bottom is the packet bytes pane, showing the raw data in hexadecimal and ASCII. The status bar at the very bottom indicates 'Ethernet: <live capture in progress>', 'Packets: 612 · Displayed: 612 (100.0%)', and 'Profile: Default'.

No.	Time	Source	Destination	Protocol	Length	Info
590	51.799660	Force10N_8b:fe:b0	Broadcast	ARP	60	Who has 139.91.68.182? Tell 139.91.68.253
591	51.822870	Force10N_8b:fe:b0	Broadcast	ARP	60	Who has 139.91.68.43? Tell 139.91.68.253
592	52.251197	CiscoInc_b0:dd:da	Spanning-tree-(for-~	STP	60	Conf. Root = 0/68/00:18:74:2f:0d:40 Cost = 4 Port = 0x8127
593	52.652088	139.91.68.253	224.0.0.18	VRRP	60	Announcement (v2)
594	52.883057	Force10N_8b:fe:b0	Broadcast	ARP	60	Who has 139.91.68.218? Tell 139.91.68.253
595	53.120106	Dell_ff:17:ee	Broadcast	ARP	60	Who has 139.91.157.1? Tell 139.91.68.103
596	53.139089	139.91.68.59	239.255.255.250	SSDP	175	M-SEARCH * HTTP/1.1
597	53.147649	Dell_ff:17:ee	Broadcast	ARP	60	Who has 139.91.157.51? Tell 139.91.68.103
598	53.244720	139.91.68.122	139.91.68.255	BJNP	60	Scanner Command: Unknown code (2)
599	53.262491	Apple_db:42:ca	Broadcast	ARP	60	Who has 139.91.68.28? Tell 139.91.68.122
600	53.285798	Dell_ff:17:ee	Broadcast	ARP	60	Who has 23.101.30.126? Tell 139.91.68.103
601	53.666375	AirgoNet_68:d3:c4	Broadcast	ARP	60	Who has 139.91.68.71? Tell 139.91.68.39
602	53.698539	Force10N_8b:fe:b0	Broadcast	ARP	60	Who has 139.91.68.248? Tell 139.91.68.253
603	53.712235	139.91.68.253	224.0.0.18	VRRP	60	Announcement (v2)
604	54.251200	CiscoInc_b0:dd:da	Spanning-tree-(for-~	STP	60	Conf. Root = 0/68/00:18:74:2f:0d:40 Cost = 4 Port = 0x8127
605	54.301336	Dell_ff:17:ee	Broadcast	ARP	60	Who has 23.97.209.97? Tell 139.91.68.103
606	54.410828	139.91.92.23	239.255.255.250	SSDP	215	M-SEARCH * HTTP/1.1
607	54.410829	139.91.92.23	239.255.255.250	SSDP	215	M-SEARCH * HTTP/1.1
608	54.708496	139.91.68.34	139.91.68.255	UDP	63	38487 → 32412 Len=21
609	54.709359	139.91.68.34	139.91.68.255	UDP	63	60323 → 32414 Len=21
610	54.710257	139.91.68.34	239.255.255.250	SSDP	136	M-SEARCH * HTTP/1.1
611	54.723288	139.91.68.253	224.0.0.18	VRRP	60	Announcement (v2)
612	54.802097	Force10N_8b:fe:b0	Broadcast	ARP	60	Who has 139.91.68.43? Tell 139.91.68.253

> Frame 468: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0  
> Ethernet II, Src: Force10N\_8b:fe:b0 (00:01:e8:8b:fe:b0), Dst: Broadcast (ff:ff:ff:ff:ff:ff)  
> Address Resolution Protocol (request)

```
0000  ff ff ff ff ff ff 00 01  e8 8b fe b0 08 06 00 01  .....
0010  08 00 06 04 00 01 00 01  e8 8b fe b0 8b 5b 44 fd  .....[D.
0020  00 00 00 00 00 00 8b 5b  44 b0 00 00 00 00 00 00  .....[ D.....
0030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  .....
```

# Packet filtering (1/2)

- Protocol name

- tcp
- udp
- ip
- http

- Multiple protocol fields

- e.g `ip.addr == 10.43.54.65`



`ip.src == 10.43.54.65 or ip.dst == 10.43.54.65`



# Packet filtering (2/2)

- Logical operators
  - AND &&
  - OR or
  - EQUAL ==
  - INEQUAL !=
- Example: filter out all traffic from/to 10.43.54.65
- Solution:

`ip.addr != 10.43.54.65`  
**OR**  
`ip.src != 10.43.54.65 or ip.dst != 10.43.54.65`



# Packet capturing

- Select interface for capturing
- Press capture button to start capturing ()
- Apply filters
- Stop capturing ()
- File-> Export Specified Packets
- Trace: the group of exported captured packets