

MP3.1 Networking



ECE 422/CS 461

Goals

Checkpoint 1

- Learn how to use Wireshark
- Identify network activities
- Identify attacks or vulnerabilities

Checkpoint 2

- Attack a network and extract information
- Programmatically detect attacks from network traces

Required Tools

Checkpoint 1

- Wireshark (either 32 bit or 64 bit version)

Checkpoint 2

- Wireshark (**32 bit**)
- Aircrack-ng Suite
- Nmap
- Python 2.7
- dpkt Python library

Objective

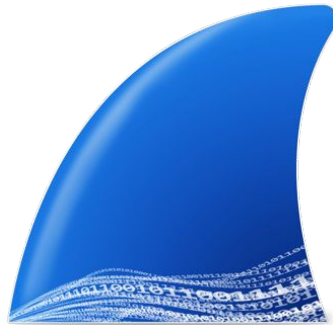
Learn how to use Wireshark

- Packets details
- Filters
- Built-in features

Identify network activities

- Gateway
- Active vs. passive FTP
- HTTPS connection

Wireshark

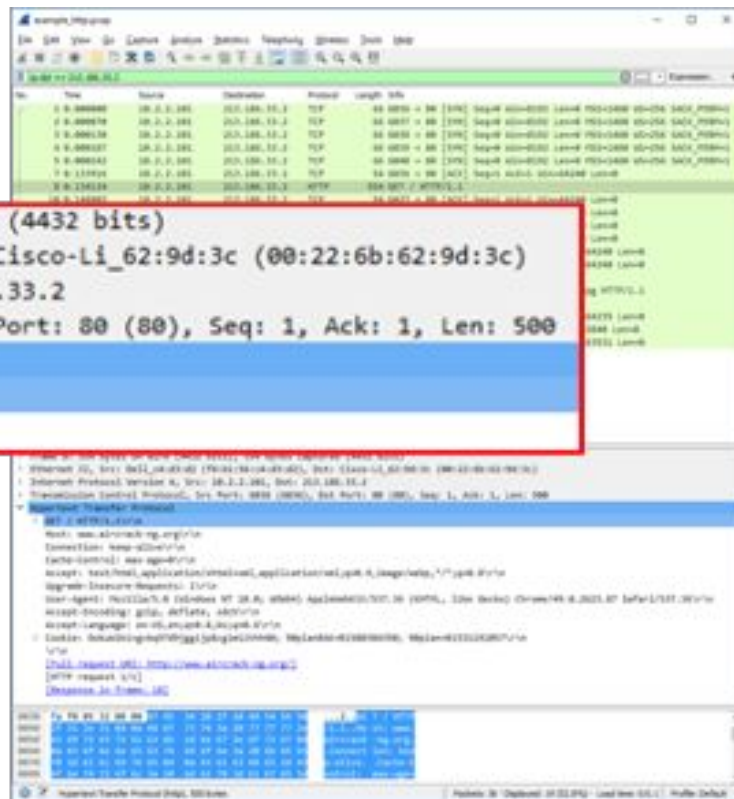


Digging through “packet details”

```
> Frame 8: 554 bytes on wire (4432 bits), 554 bytes captured (4432 bits)
> Ethernet II, Src: Dell_c4:d3:d2 (f8:b1:56:c4:d3:d2), Dst: Cisco-Li_62:9d:3c (00:22:6b:62:9d:3c)
> Internet Protocol Version 4, Src: 10.2.2.101, Dst: 213.186.33.2
> Transmission Control Protocol, Src Port: 6036 (6036), Dst Port: 80 (80), Seq: 1, Ack: 1, Len: 500
▼ Hypertext Transfer Protocol
  > GET / HTTP/1.1\r\n
    Host: www.aircrack-ng.org\r\n
```

Everything that Wireshark can tell you about packet

IP addr, port numbers, MAC addresses, hostname,
data, etc.




Display Filters

Apply a display filter ... <=>/>						
No.	Time	Source	Destination	Protocol	Length	Info
121	0.195554	10.183.158.36	10.183.158.4	FTP-D...	128	FTP Data: 62 bytes
122	0.195559	10.183.158.4	10.183.158.36	TCP	66	36328 → 10895 [ACK] Seq=1 Ack=63 Win=29312 Len=0 TSval=316609339 TSecr=316609339
123	0.195569	10.183.158.36	10.183.158.4	TCP	66	10895 → 36328 [FIN, ACK] Seq=63 Ack=1 Win=29056 Len=0 TSval=316609339 TSecr=316609339
124	0.195666	10.183.158.4	10.183.158.36	TCP	66	36328 → 10895 [FIN, ACK] Seq=1 Ack=64 Win=29312 Len=0 TSval=316609339 TSecr=316609339
125	0.195736	10.183.158.36	10.183.158.4	TCP	66	10895 → 36328 [ACK] Seq=64 Ack=2 Win=29056 Len=0 TSval=316609339 TSecr=316609339
126	0.195789	10.183.158.36	10.183.158.4	FTP	90	Response: 226 Transfer complete.
127	0.195861	10.183.158.4	10.183.158.36	TCP	66	46540 → 21 [ACK] Seq=85 Ack=289 Win=29312 Len=0 TSval=316609339 TSecr=316609339
128	0.195916	10.183.158.4	10.183.158.36	TCP	66	46540 → 21 [FIN, ACK] Seq=85 Ack=289 Win=29312 Len=0 TSval=316609339 TSecr=316609339
129	0.195942	10.183.158.36	10.183.158.4	TCP	66	21 → 46540 [FIN, ACK] Seq=289 Ack=86 Win=29056 Len=0 TSval=316609339 TSecr=316609339
130	0.195953	10.183.158.4	10.183.158.36	TCP	66	46540 → 21 [ACK] Seq=86 Ack=290 Win=29312 Len=0 TSval=316609339 TSecr=316609339

tcp						
No.	Time	Source	Destination	Protocol	Length	Info
132	0.682141	10.183.158.6	10.183.158.36	FTP	74	Request: TYPE I
133	0.682154	10.183.158.23	10.183.158.4	FTP	97	Response: 200 Switching to Binary mode.
134	0.682159	10.183.158.6	10.183.158.36	FTP	81	Request: SIZE flag.txt
135	0.682168	10.183.158.23	10.183.158.4	FTP	74	Response: 213 62
136	0.682173	10.183.158.6	10.183.158.36	FTP	72	Request: PASV
137	0.682181	10.183.158.23	10.183.158.4	FTP	117	Response: 227 Entering Passive Mode (10,183,158,36,42,143).
138	0.682185	10.183.158.6	10.183.158.36	TCP	74	36328 → 10895 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=316609339 TSecr=316609339
139	0.682193	10.183.158.23	10.183.158.4	TCP	74	10895 → 36328 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=316609339 TSecr=316609339
140	0.682197	10.183.158.6	10.183.158.36	TCP	66	36328 → 10895 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=316609339 TSecr=316609339
141	0.682204	10.183.158.23	10.183.158.4	FTP	81	Request: RETR flag.txt
142	0.682209	10.183.158.6	10.183.158.36	FTP	132	Response: 150 Opening BINARY mode data connection for flag.txt (62 bytes).
143	0.682214	10.183.158.6	10.183.158.36	FTP-D...	128	FTP Data: 62 bytes
144	0.682219	10.183.158.6	10.183.158.36	TCP	66	36328 → 10895 [ACK] Seq=1 Ack=63 Win=29312 Len=0 TSval=316609339 TSecr=316609339
145	0.682224	10.183.158.6	10.183.158.36	TCP	66	10895 → 36328 [FIN, ACK] Seq=63 Ack=1 Win=29056 Len=0 TSval=316609339 TSecr=316609339
146	0.682229	10.183.158.6	10.183.158.36	TCP	66	36328 → 10895 [FIN, ACK] Seq=1 Ack=64 Win=29312 Len=0 TSval=316609339 TSecr=316609339
147	0.682234	10.183.158.6	10.183.158.36	TCP	66	10895 → 36328 [ACK] Seq=64 Ack=2 Win=29056 Len=0 TSval=316609339 TSecr=316609339
148	0.682239	10.183.158.6	10.183.158.36	FTP	90	Response: 226 Transfer complete.
149	0.682244	10.183.158.6	10.183.158.36	TCP	66	46540 → 21 [ACK] Seq=85 Ack=289 Win=29312 Len=0 TSval=316609339 TSecr=316609339
150	0.682249	10.183.158.6	10.183.158.36	TCP	66	46540 → 21 [FIN, ACK] Seq=85 Ack=289 Win=29312 Len=0 TSval=316609339 TSecr=316609339
151	0.682254	10.183.158.6	10.183.158.36	TCP	66	21 → 46540 [FIN, ACK] Seq=289 Ack=86 Win=29056 Len=0 TSval=316609339 TSecr=316609339
152	0.682259	10.183.158.6	10.183.158.36	TCP	66	46540 → 21 [ACK] Seq=86 Ack=290 Win=29312 Len=0 TSval=316609339 TSecr=316609339

Display Filters

 `ip.dst == 10.244.130.80`

Shows packets that contain the information you are interested in

Examples: <https://wiki.wireshark.org/DisplayFilters>

Filter expression basics and syntax:

https://www.wireshark.org/docs/wsug_html_chunked/ChWorkBuildDisplayFilterSection.html

Filter reference: <https://www.wireshark.org/docs/dfref/>

What if you want to see MAC addresses of all packets instead of filtering by specific one?

Custom Columns

No.	Time	Source	Destination	Length	Protocol
23	1.801809	10.2.2.101	68.180.77.151	55	SSL
24	1.806705	68.180.77.151	10.2.2.101	66	TCP
25	2.109691	00:22:6b:62:9d:3c	ff:ff:ff:ff:ff:ff	60	ARP



No.	Time	Source	Destination	SrcMAC	DstMAC	Length	Protocol
23	1.801809	10.2.2.101	68.180.77.151	f8:b1:56:c4:d3:d2	00:22:6b:62:9d:3c	55	SSL
24	1.806705	68.180.77.151	10.2.2.101	00:22:6b:62:9d:3c	f8:b1:56:c4:d3:d2	66	TCP
25	2.109691	00:22:6b:62:9d:3c	ff:ff:ff:ff:ff:ff	00:22:6b:62:9d:3c	ff:ff:ff:ff:ff:ff	60	ARP

Custom Columns

Right-click column header > Column preferences

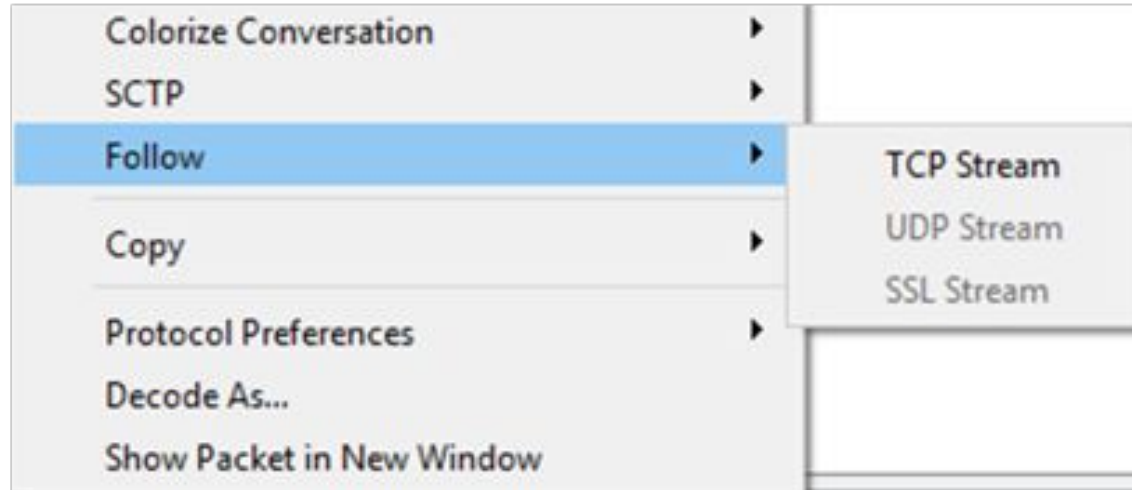
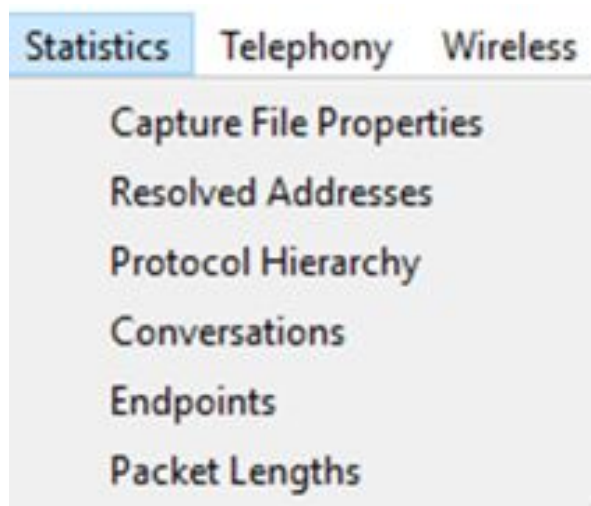
or Edit > Preferences > Appearance: Columns

Displayed	Title	Type	Field Name	Field Occurrence
<input checked="" type="checkbox"/>	No.	Number		
<input checked="" type="checkbox"/>	Time	Time (format as specified)		
<input checked="" type="checkbox"/>	Source	Source address		
<input checked="" type="checkbox"/>	SrcMAC	Custom	eth.src	0
<input checked="" type="checkbox"/>	Destination	Destination address		
<input checked="" type="checkbox"/>	DstMAC	Custom	eth.dst	0
<input checked="" type="checkbox"/>	Protocol	Protocol		
<input checked="" type="checkbox"/>	Length	Packet length (bytes)		
<input checked="" type="checkbox"/>	Info	Information		

Built-in Features

Menu (e.g. Statistics)

Packets/Packet details (e.g. Follow TCP stream)



Ex) Follow TCP Stream

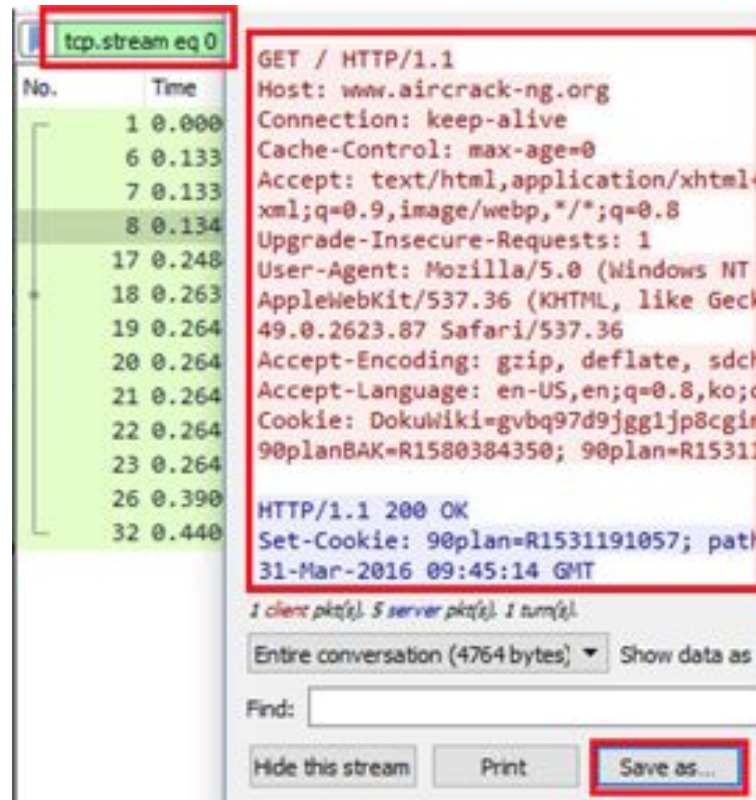
Shows all packets in the same TCP stream `tcp.stream eq x`

Opens a new window that shows content of all relevant packets in readable format

Option to save a file

What information does Wireshark use to detect different conversations?

- IP address, port, sequence number



Built-in Features - Name Resolution

View > Name Resolution > Resolve Physical/Network/Transport Address

Wireshark converts numerical addresses into (more) human readable formats
(https://www.wireshark.org/docs/wsug_html_chunked/ChAdvNameResolutionSection.html)

While useful, the conversion often fails and may give you wrong information (e.g. wrong hostname)

Try “Resolve Network Address” on 3.1.1.pcap. Try it on IllinoisNet. Try it at home.

Common Network Activities



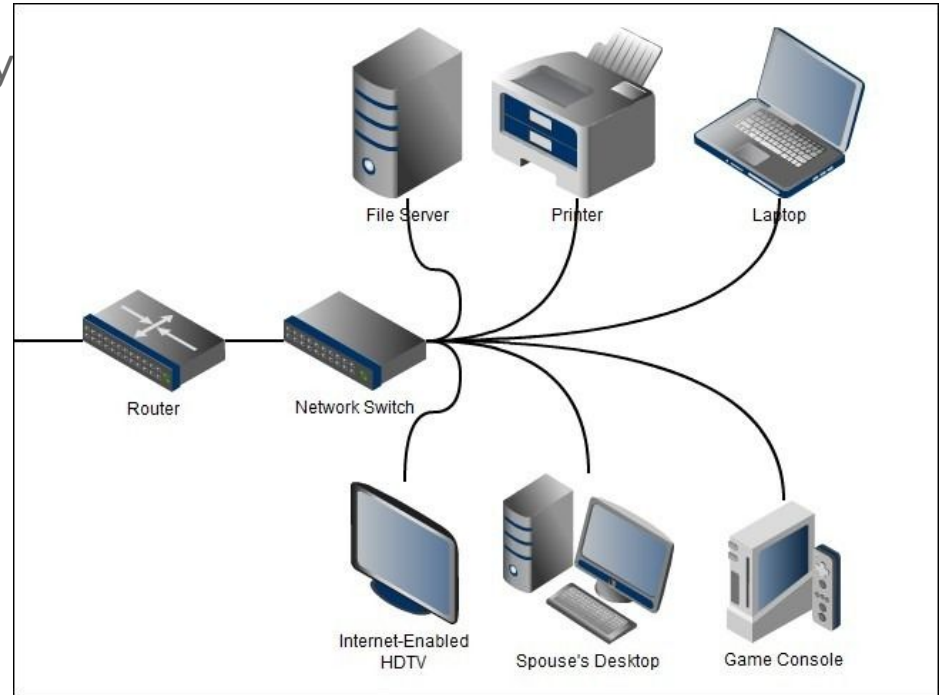
Gateway

A default gateway... [forwards] packets on to other networks... The gateway is by definition a router.

A router is a networking device that forwards data packets between computer networks.

https://en.wikipedia.org/wiki/Default_gateway

[https://en.wikipedia.org/wiki/Router_\(computing\)](https://en.wikipedia.org/wiki/Router_(computing))



Identifying a gateway

Assumption: all external traffic goes through the network's gateway

Look at the packets between a local host and a number of different external hosts (e.g. websites). Check the MAC addresses of the external hosts. Are they different?

See what other IP addresses are mapped to that MAC address.

Source	SrcMAC	Destination	DstMAC
10.2.2.101	f8:b1:56:c4:d3:d2	telemetry.battle.net	00:22:6b:62:9d:3c
10.2.2.101	f8:b1:56:c4:d3:d2	www.aircrack-ng.org	00:22:6b:62:9d:3c

Why is MAC addr. not used to detect TCP streams?

IP-to-MAC not necessarily a 1:1 mapping

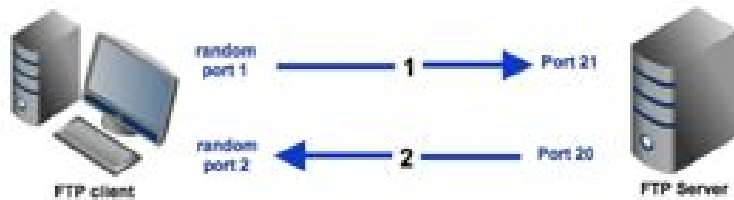
1 MAC address can be mapped to multiple IP addresses (like in previous slide)

1 IP address can be mapped to multiple MAC addresses (e.g. IP spoofing)

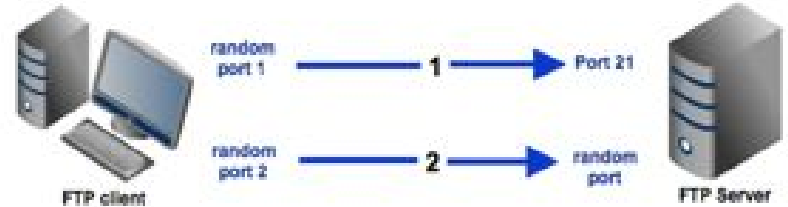
How to see the complete mapping?

- Filter by source/destination MAC address
- Sort on IP address

Active vs. Passive FTP



Active FTP



Passive FTP

Explanation:

<http://www.jscape.com/blog/bid/80512/Active-v-s-Passive-FTP-Simplified>

With FTP session examples: <http://slacksite.com/other/ftp.html>

HTTPS Connections

TLS Handshake

- Client sends a Client Hello offers a list of Cipher Suites
- Server responds with Server Hello that contains chosen Cipher Suite

The first few milliseconds of an HTTPS connections

(<http://www.moserware.com/2009/06/first-few-milliseconds-of-https.html>)

Tips

Start focusing on one conversation.

Try to understand the result shown by Wireshark and make sure it is as expected.

Get familiar with filter syntax and take advantage of it. Expressions made of multiple filters will save you from tedious scrolling.

Try capturing your own network traffic and analyze it.

Don't make assumptions and limit your search from the beginning.

Reminders

Make sure to read the MP3 release post on Piazza

- Do NOT use lab computers, for your own good

Please read the Checkpoint 2 setup and start setting up your environment as soon as possible

- Some parts are easier when there are less people working on them at the same time