

Tcpdump and Wireshark

COS 461

Muneeb Ali

History



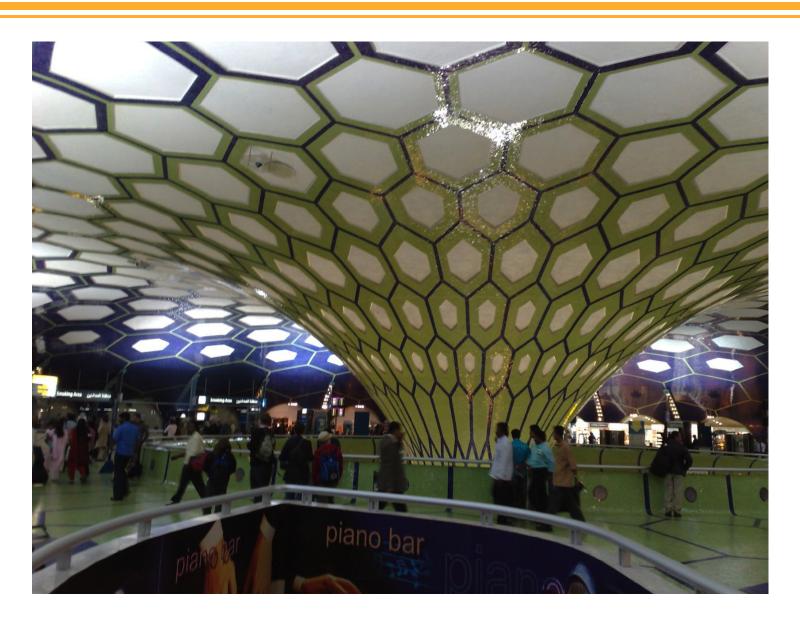
- Tcpdump, 1987, Van Jacopson et al. (Lawrence Berkeley Lab)
- Wireshark, 1998, Ethereal open source, 1-500 developers





- A story ...







- A story ...
- Analyze network protocols



- A story ...
- Analyze network protocols
- (Don't) Hack



tcpdump



```
[1] [2] [3]
15:04:15.532786 IP w4-laptop.CS.Princeton.EDU.17968 >
81-233-76-7-no73.tbcn.telia.com.ipdcesgbs: P 1380:1460(80) ack 1 win 15941
[4] [5] [6] [7] [8] [9] [10] [11]
```

- [1] Timestamp
- [2] Source IP
- [3] Source Port
- [4] Destination IP
- [5] Destination Port
- [6] TCP Flags
- [7] TCP Sequence Number
- [8] TCP Last Sequence Number
- [9] TCP Packet Length
- [10] ACK Flag (next expected)
- [11] Window size (host expects)

tcpdump



- tcpdump -i <interface>
- tcpdump -A (print packet content)
- tcpdump -c <num> (exit after num packets)
- tcpdump -v (verbose e.g., TTL, ICMP checkcum etc.)
- tcpdump –vv (even more verbose)

. . . .

Expressions:

- tcpdump port <port>
- tcpdump dst host <host>
- tcpdump src host <host>
- tcpdump length <length> (packet >= length)
- tcpdump tcp (ip proto p or ip6 proto p, where p=tcp)

. . .

Tcpdump examples (Mac OS X)



- Use "ifconfig" or "sudo tcpdump -D" to get a list of interfaces
- "sudo tcpdump -i en1" dumps the traffic on en1 (wireless interface)
- "sudo tcpdump -i lo0 port 3333" dumps traffic on lo0 (local) filtered by port 3333
- "sudo tcpdump -i en1 udp" dumps only udp packets

In the proxy server demo, we used "telnet localhost 3333"
GET http://muneeb.org HTTP/1.0

- Will find this useful for assignment 2 and 3, but even now you can telnet to your server and type/send the text that the server should output
- -More examples of tcpdump at http://openmaniak.com/tcpdump.php

Wireshark examples (Mac OS X)



- Download wireshark from http://www.wireshark.org/download.html
- Run with "sudo wireshark" (otherwise the interfaces won't be accessible)
- Select interface e.g., "en1"
- Enter "filter" like "http and ip.src=<ip address> and ip.dst==<ip address>
- Didn't get time to cover this, but if you enter filter "ssh" and try to look at packet data of ssh you'll see that the data is encrypted
- This is not the case with HTTP you can see the data other people are requesting!
- Security is important e.g., FTP passwds sent without encryption, so always use SFTP!
- Wireshark "filters" are a little different from tcpdump "expressions"
 See http://openmaniak.com/wireshark_filters.php
- Use "Analyze -> Follow TCP stream" to construct entire streams (very useful)
- Can use files to store/read traces e.g., tcp-raw.pcap (download from wireshark website)

Questions?



Thank you!
Happy tcpdumping!