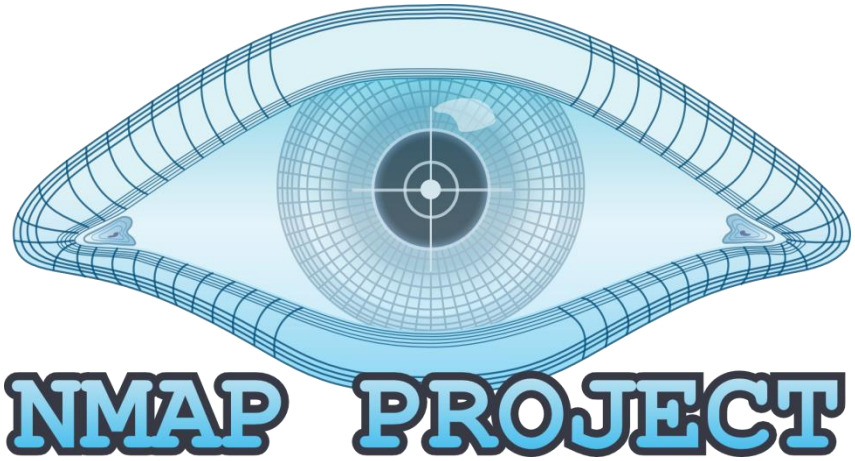
Network Security (NetSec)

Summer 2015
Exercise 4: Reconnaissance



Source: http://nmap.org

Prof. Dr.-Ing. Matthias Hollick

Technische Universität Darmstadt
Secure Mobile Networking Lab - SEEMOO
Department of Computer Science
Center for Advanced Security Research Darmstadt - CASED

Mornewegstr. 32
D-64293 Darmstadt, Germany
Tel.+49 6151 16-70922, Fax. +49 6151 16-70921
http://seemoo.de or http://www.seemoo.tu-darmstadt.de

Milan Schmittner
milan.schmittner@seemoo.tu-darmstadt.de

Nmap (1)

Extremely popular
- usually run over Linux
- rich feature set, exploiting raw sockets
- need root to use all features

Ping sweeping
- over any range of IP addresses
- with ICMP, SYN, ACK
- OS determination

Port scanning
- Over any range of ports
- Almost any type of TCP, UDP packet

Source IP address spoofing
- Decoy scanning

Packet fragmentation

Timing Options

Further information:
http://nmap.org/book/man.html

Nmap (2)

Input

```
nmap [Scan Type] [Options] <target hosts>
```

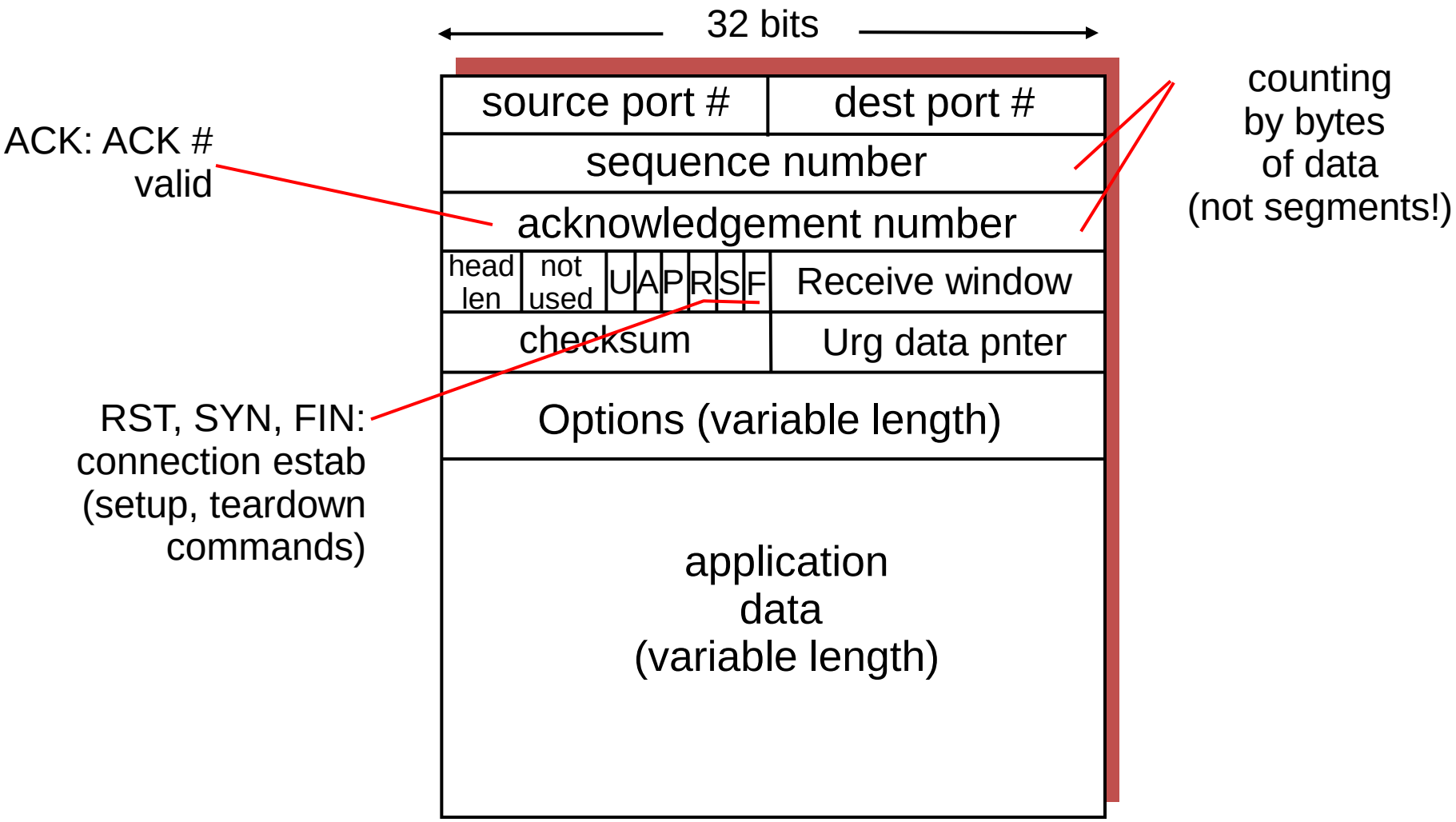
- Default for port scanning: ports 1-1024 plus ports listed in nmap service file

Output

- open ports: syn/ack returned; port is open
- unfiltered ports: RST returned: port is closed but not blocked by firewall
- filtered ports: nothing returned; port is blocked by firewall

See Appendix for further examples

Excursus: TCP Segment Structure



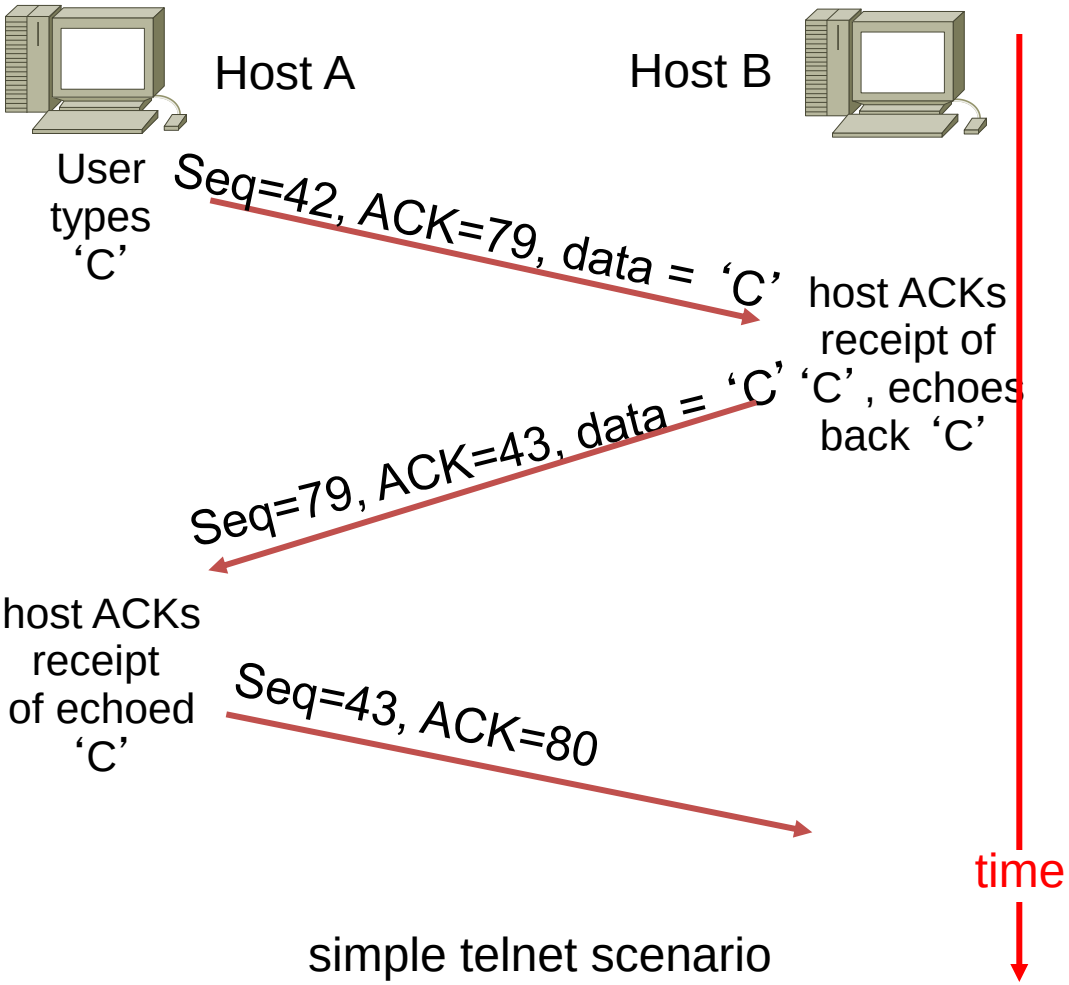
Excursus: TCP seq. #'s and ACKs

Seq. #'s:

- byte stream "number" of first byte in segment's data

ACKs:

- seq # of next byte expected from other side



simple telnet scenario

Excursus: TCP Connection Establishment

Three way handshake:

Step 1: client host sends TCP SYN segment to server
- SYN=1, ACK=0
- specifies initial seq #
- no data

Step 2: server host receives SYN, replies with SYN-ACK segment
- SYN=1, ACK=1
- server host allocates buffers
- specifies server initial seq. #

Step 3: client receives SYN-ACK, replies with ACK segment, which may contain data
- SYN=0, ACK=1

TCP: Reset packet

If machine receives a TCP packet it is not expecting, it responds with TCP packet with RST bit set.

- For example when no process is listening on destination port

For UDP, machine returns ICMP “port unreachable” instead

Nmap (3): ping sweep

```
nmap -sP -v 116.27.38/24
```

Sends ICMP echo request (ping) to 256 addresses

Can change options so that pings with SYNs, ACKs…

- **-sP = ping**
- **-v = verbose**

Nmap (4): polite port scan

```
nmap -sT -v target.com
```

Attempts to complete 3-way handshake with each target port

Sends SYN, waits for SYNACK, sends ACK, then sends FIN to close connection

If target port is closed, no SYNACK returned

- Instead RST packet is typically returned

TCP connect scans are easy to detect

- Target (e.g. Web server) may log completed connections
- Gives away attacker's IP address

Nmap (5) : TCP SYN port scan

```
nmap -sS -v target.com
```

Stealthier than polite scan

Send SYN, receive SYNACK, send RST

- Send RST segment to avoid an accidental DoS attack

Stealthier: hosts do not record connection

- But routers with logging enabled will record the SYN packet

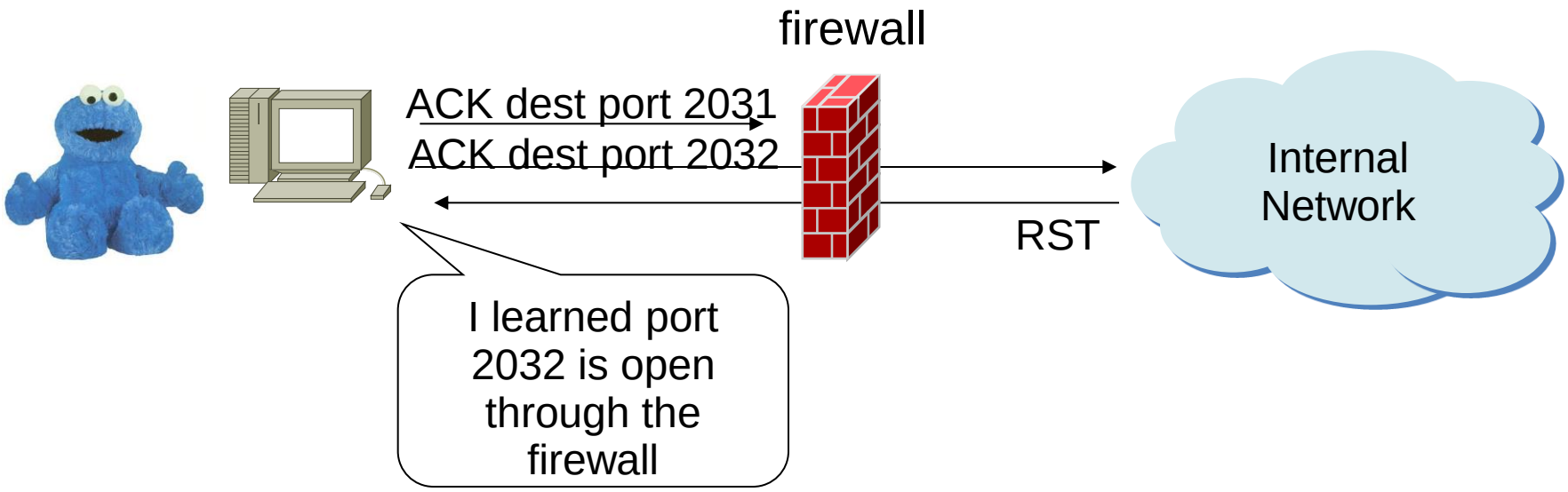
Faster: don't need to send FIN packet

Nmap (6): TCP ACK scans

Many filters (in firewalls and routers) only let internal systems hosts initiate TCP connections

- Drop packets for which ACK=0 (ie SYN packet): no sessions initiated externally

To learn what ports are open through firewall, try an ACK scan (segments with ACK=1)



Nmap (7): UDP port scans

UDP doesn't have SYN, ACK, RST packets

nmap simply sends UDP packet to target port

- ICMP Port Unreachable: interpret port closed
- Nothing comes back: interpret port open
  - False positives common

Nmap (8): Obscure Source

Attacker can enter list of decoy source IP addresses into Nmap

For each packet it sends, Nmap also sends packets from decoy source IP addresses

- For 4 decoy sources, send five packets

Attacker's actual address must appear in at least one packet, to get a result

If there are 30 decoys, victim network will have to investigate 31 different sources!

Nmap (9): TCP Stack Fingerprinting

In addition to determining open ports, attacker wants to know OS on targeted machine:

- exploit machine's known vulnerabilities
- sophisticated hacker may set up lab environment similar to target network

TCP implementations in different OSes respond differently to (illegal) combinations of TCP flag bits

Nmap (10): Fingerprinting

Nmap sends

- SYN to open port
- NULL to open port (no flag bits set)
- SYN/FIN/URG/PSH to open port
- SYN to closed port
- ACK to closed port
- FIN/PSH/URG to closed port
- UDP to closed port

Nmap includes a database of OS fingerprints for hundreds of platforms

- See nmap.org for further details

Nmap (11): examples

`nmap -v target.com`

- Scans all TCP default ports on target.com; verbose mode

`nmap -sS -O target.com/24`

- First pings addresses in target network to find hosts that are up. Then scans default ports at these hosts; stealth mode (doesn't complete the connections); tries to determine OS running on each scanned host

`nmap -sX -p 22,53,110,143 198.116.*.1-137`

- Sends an Xmas tree scan to the first half of each of the 255 possible subnets in the 198.116/16. Testing whether the systems run ssh, DNS, pop3, or imap

`nmap -v -p 80 *.*.2.3-5`

- finds all web servers on machines with IP addresses ending in .2.3, .2.4, or .2.5