NETVÆRKS- OG KOMMUNIKATIONSSIKKERHED

TCP/IP

daka@kea.dk

Agenda

- Exercise from last week
- Nmap scanner
- Packet building
- TCP attack
- ARP spoofing

Exercise from last week

- Gather all you can of information with passive reconnaissance
- How much can you learn from that?
- Did you all stick to passive reconnaissance?
- Where is the border between passive and active reconnaissance?
- Take it a step further now and do active scan.
  - No weaponization, no exploit, no port scans
  - Work in groups, and present your findings next week

NMap

- Scanner tool
- Can apply various approached for detecting open ports
- Uses the RFC 793
- Can be detected by most IDS and IPS systems today

NMap

- Can do OS fingerprinting
- Run the command (replace ip address with your machines IP)
  - `nmap -O -v 192.168.65.1`
- Make sure that your wireshark is running
- What types of packets are sent and why?

NMap

- Some of the scanning modes are more aggressive that others

- Find out how the following command finds the different hosts on a network using wireshark(replace ip address with you own)
  - `nmap –vv –n –sn –T4 192.168.65.1/24`

- Run it again against a specific target and snif
  - `nmap –vv –Pn –sS –A 192.168.65.1`

NMap

- What is the difference between -sS and –sT? (run in wireshark)
  - `nmap –vv –Pn –sT –A 192.168.65.1`

- How do we know if a firewall is there?
  - Consider using –sA
  - A RST is sent back in case is it is open or closed
  - Open: connection possible
  - Closed: No service availiable
  - Filtered: firewall drops packet

Packet building

- Packets are not magical!

- Windows
  - Colasoft packet builder (http://www.colasoft.com/packet_builder/)
  - Engage packet builder (http://www.engagesecurity.com/products/engagepacketbuilder/)
  - TCP inspection (https://docs.microsoft.com/da-dk/sysinternals/downloads/tcpview)
  - RawCap (http://www.netresec.com/?page=RawCap)
  - Most of these tools require that you run them as administrator

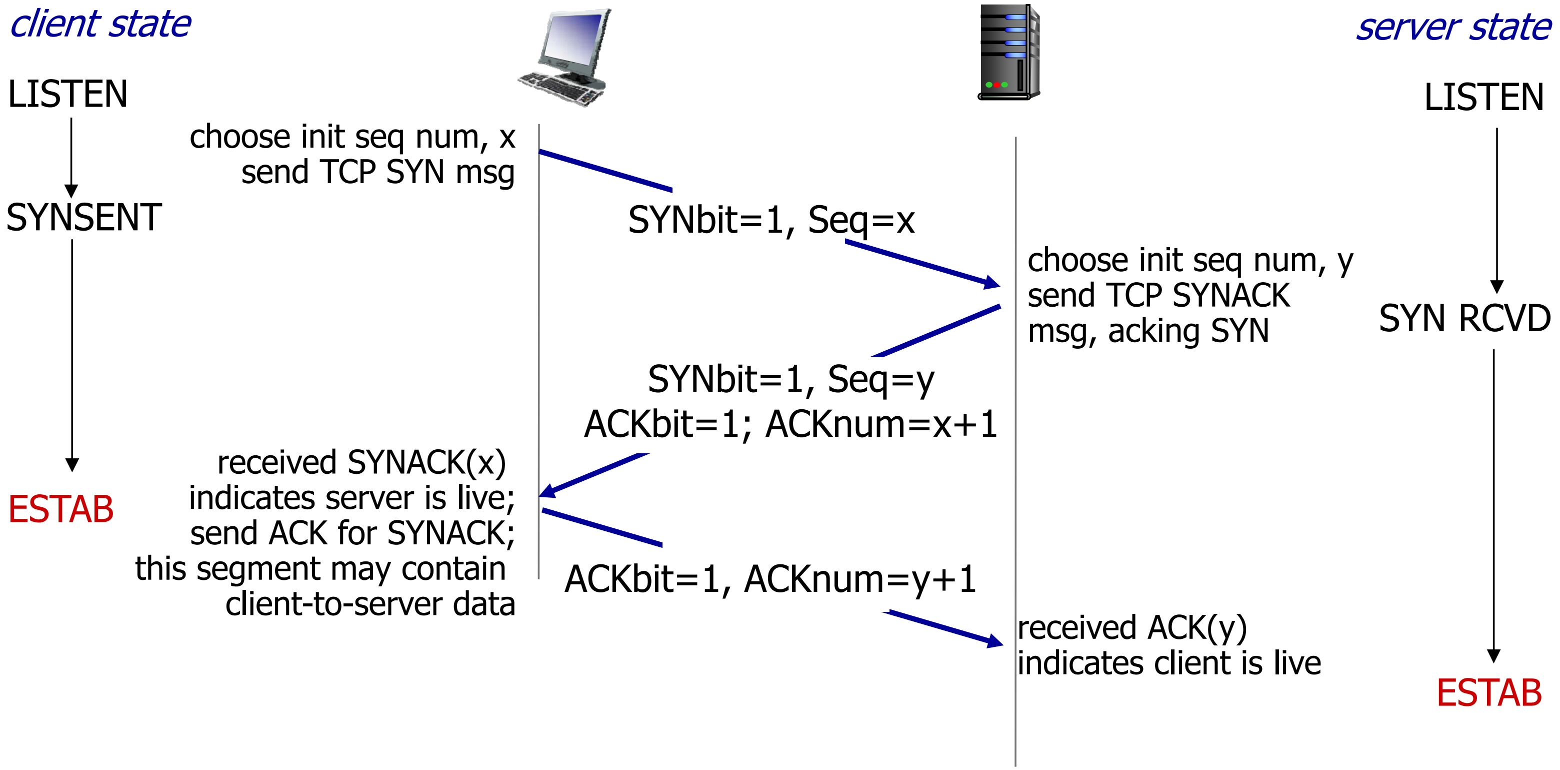
Sending custom packets

TCP attacks

- Abusing some of the features in TCP

- TCP 3-way handshake can form a basis for multiple attacks
  - Does not require a already established connection
  - TCP is connection oriented and therefore uses resources
  - TCP handshake is very common and the basis of all traffic

Quick recap

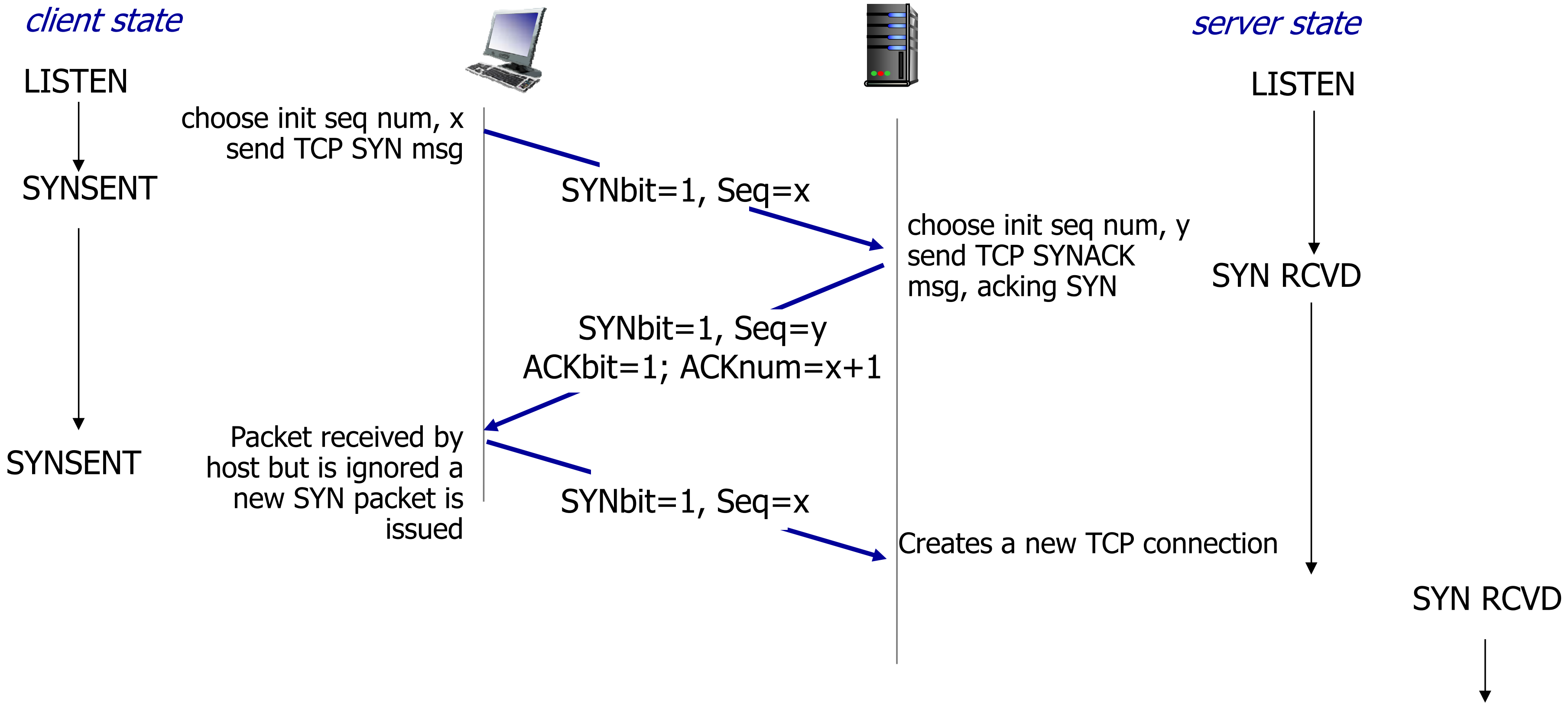


client state
LISTEN
SYNSENT
ESTAB
choose init seq num, x
send TCP SYN msg
SYNbit=1, Seq=x
server state
LISTEN
choose init seq num, y
send TCP SYNACK
msg, acking SYN
SYN RCVD
SYNbit=1, Seq=y
ACKbit=1; ACKnum=x+1
received SYNACK(x)
indicates server is live;
send ACK for SYNACK;
this segment may contain
client-to-server data
ACKbit=1, ACKnum=y+1
received ACK(y)
indicates client is live
ESTAB

Syn flood

- Exploiting the 3 way handshake by only using the syn flag
- Established "half-open" connections that eat up resources on the system
- Is somewhat dealt with by modern OS, but problem remains

Quick recap



client state

LISTEN

SYNSENT

SYNSENT

choose init seq num, x
send TCP SYN msg

SYNbit=1, Seq=x

choose init seq num, y
send TCP SYNACK
msg, acking SYN

SYNbit=1, Seq=y
ACKbit=1; ACKnum=x+1

Packet received by
host but is ignored a
new SYN packet is
issued

SYNbit=1, Seq=x

Creates a new TCP connection

server state

LISTEN

SYN RCVD

SYN RCVD

Lets try it…

- Follow my steps and look at the snif
- Are you succeeding into keeping the connection "half-open"?

TCP syn from the other side

- Lets try to now to do the same from the kali side

- We should be able to stop the RST with a simple firewall rule (replace the ip with your kali linux ip)
  - `iptables -A OUTPUT -p tcp --tcp-flags RST RST -s 192.168.65.131 -j DROP`

- Now lets build packets using python :-)

scapy – your new best friend

- A library/tool that is both a sniffer and a packet injector
- Can be used directly form commandline
- Can also be import fra a python program
- Lots of python scripts are built with it

scapy

- From your kali terminal enter scapy
- You will them get python terminal and you are ready to go
- Use `ls()` and `lsc()` to help you with the commands and protocols you want to issue.

scapy

- Most important commands include

send()	Sends a packet in layer 3
sendp()	Sends a packet in layer 2
sr()	Send and wait for response
sniff()	sniffs traffic
rdpcap()	import a pcap file

scapy

- You can sniff traffic simply by

```
pkts = sniff(count=5,filter=“tcp")
pkts.summary()
pkts[1].show()
```

- You can also instead import a cap file

```
pkts  = rdpcap(‘capture.cap')
```

scapy

- Try using the srflood() in scapy to flood a server with tcp syn
- You will need both an IP and a TCP headers
- Writing `ls(IP)` and `ls(TCP)` will provide you with details on what you can fill out

scapy

- Try with the following with wireshark open (change the IPs)

```
packet = IP(src="192.168.65.131",dst="192.168.65.1")/TCP(dport=80,flags="S")
srflood(packet)
```

- What is this doing?
- How is your machine responding to this “flood”?
  - Look at your TCPview or your `netstat`

Scapy - Challenge

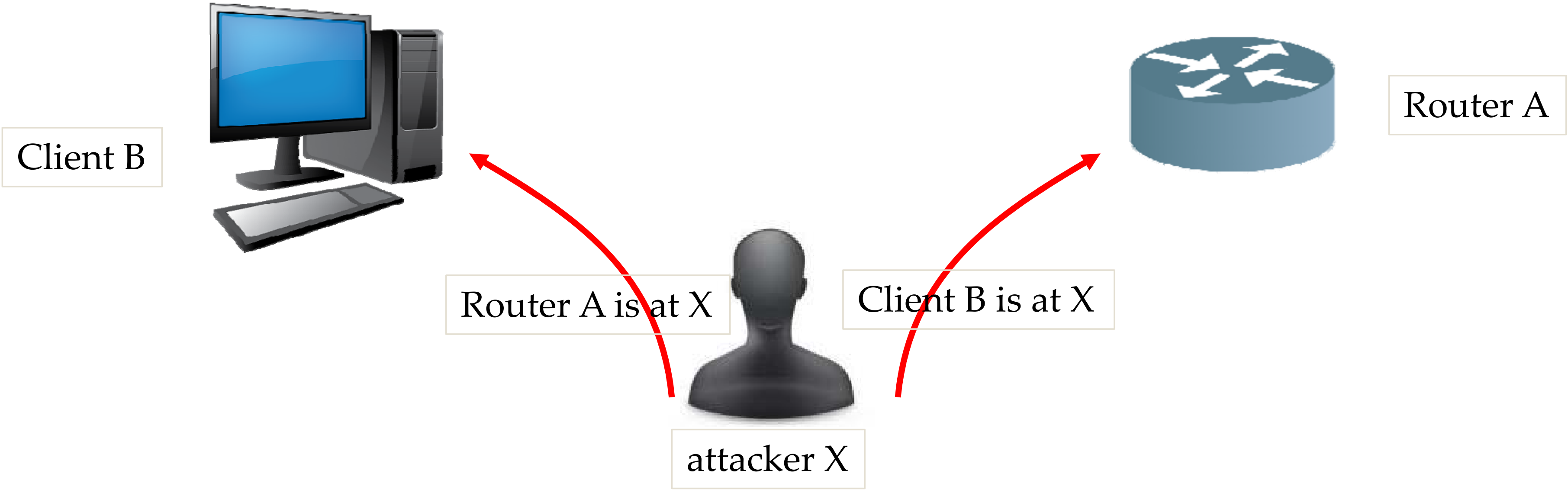
- Now write a program in python that will send 100 SYN packets in the following form
  - It will send the packets spoffing the ip address of the sender (`src`) to 10 addresses of your choosing
  - The source port (`sport`) in the TCP should also be at least 10 different ports
  - Ps. Use `send()` to send each packet
- The code you write should not be more than 5 lines long

Arp

- A wants to send datagram to B
  - B' s MAC address not in A' s ARP table.
- A broadcasts ARP query packet, containing B's IP address
  - dest MAC address = FF-FF-FF-FF-FF-FF
  - all nodes on LAN receive ARP query
- B receives ARP packet, replies to A with its (B's) MAC address
  - frame sent to A' s MAC address (unicast)
- A caches (saves) IP-to-MAC address pair in its ARP table until information becomes old (times out)
  - soft state: information that times out (goes away) unless refreshed
- ARP is "plug-and-play" :
  - nodes create their ARP tables *without intervention from net administrator*

Arp poisoning

- This can be exploited to perform MitM attack



ARP with scapy

- Send an ARP packet to a client on the network

- Use `ls(ARP)` to find you what you can set.
- First try to send any ARP packet and se If you can capture it
- Next step is to try to add ARP entries to a different machine
- Ultimately you want to make a MitM
- Ps: you might want to add Ethernet to your ARP packet `(Ether(…)/ARP(…))`

Arp spoofing with arpspoof

- Kali linux has got a built in app for doing just that.

- You can follow this guide
  - http://www.solutionsatexperts.com/arp-spoofing-attack-kali-linux/

For next time

- Make sure that you have done the execises above
- Create a presentation about your reconnaissance at KEA.