



Murdoch
UNIVERSITY

Network Security

ICT287 Computer Security



Overview

- Discuss recon and scanning techniques and relevant tools (attack & defence)
 - **Port/vulnerability scanning**
 - **Sniffing**
- Look into **spoofing attacks**, discuss how they work and how they can be prevented/mitigated
- Look at **(D)DoS attacks** (brief)
- Discuss **threat modelling techniques**

Hacking Phases

Recon and scanning are two initial hacking phases



Reconnaissance

- Process of identifying targets and developing methods necessary to attack those targets successfully
- Approaches can vary greatly and include both technical and non-technical methods
- Not by definition illegal
 - Many techniques are completely legal (but may be frowned upon)
 - Others are legal when conducted by authorised parties (usually as part of a penetration test)
 - Some legal techniques may be “outlawed” by your ISP or organization network policies



<http://pinterest.com>

Scanning

- **Port scanning:** scanning for information such as running hosts, open ports, services and applications running
- **Vulnerability scanning:** scan for vulnerabilities or weaknesses in accessible services
- **Network Mapping:** discover network structure
 - Subnets, routers, switches, firewalls, ..
 - Map of network useful to navigate around
- Mostly legal but may violate orgs policies
- If scans are detected target might notify your ISP



<http://pinterest.com>

Legal Reconnaissance/Scanning

Legal activities

- Looking up information about company available on Internet (e.g. whois, Facebook)
- Calling with problem requiring customer service assistance
- Interviewing member of staff for school project
- Physical entry of facility (if legal), including attending tour of facility
- Making friends with somebody who works there or used to work there and asking questions

Questionable Reconnaissance/Scanning

Questionable activities

- Performing port scan or other scans
- Reading names on mail sitting on mail cart
- Looking at document lying loose on desk
- Picking up copy of employee newsletter
- Looking through garbage
- War driving (searching for open wireless nets)

Illegal Reconnaissance/Scanning

Illegal activities

- Developing “front” company for purpose of robbing or defrauding
- Stealing garbage (or other things)
- Entering home or office to look for information
- Installing keylogger
- Installing sniffer



Categories of Reconnaissance

- Reconnaissance techniques can be divided into two categories: Active and Passive
- Passive recon usually relies on existing data sources without any overt interaction with the target
- Active recon involves some level of direct interaction and is more likely to be detected

Passive Reconnaissance

Don't directly interact with target

E.g. searching public records or news

Active Reconnaissance

Directly interact with target by some means

E.g. phoning up the technical department

Passive Reconnaissance

- Use existing information about the target to gain information
- Internet whois queries
 - Who is in charge of the network?
 - How can they be contacted
 - Network enumeration
- Domain Name System (DNS) reconnaissance
- Public websites
 - Who is in charge of the company?
 - Where are they located?
- Social media presence
 - Where did someone go to school / university?
 - Where have they lived?

WHOIS information for cisco.com :

```
[Querying whois.verisign-grs.com]
[Redirected to whois.melbourneit.com]
[Querying whois.melbourneit.com]
[whois.melbourneit.com]
```

```
Domain Name..... cisco.com
Creation Date..... 1987-05-14
Registration Date.... 2011-04-06
Expiry Date..... 2012-05-16
Organisation Name.... Cisco Technology, Inc.
Organisation Address. 170 W. Tasman Drive
Organisation Address.
Organisation Address. San Jose
Organisation Address. 95134
Organisation Address. CA
Organisation Address. UNITED STATES
```

```
Admin Name..... Info Sec
Admin Address..... 170 West Tasman Drive
Admin Address.....
Admin Address..... San Jose
Admin Address..... 95134
Admin Address..... CA
Admin Address..... UNITED STATES
Admin Email..... infosec@cisco.com
Admin Phone..... +1.4085273842
Admin Fax..... +1.4085264575
```

```
Tech Name..... Network Services
Tech Address..... 170 W. Tasman Drive
Tech Address.....
Tech Address..... San Jose
Tech Address..... 95134
Tech Address..... CA
Tech Address..... UNITED STATES
Tech Email..... dns-info@cisco.com
Tech Phone..... +1.4085279223
Tech Fax..... +1.4085267373
Name Server..... NS1.CISCO.COM
Name Server..... NS2.CISCO.COM
```

Active Reconnaissance

- Use more direct methods of information gathering
- Physical techniques
 - Sending emails or making phone calls
 - Meeting individuals or attending public-facing premises
 - Dumpster diving
 - Physical intrusions (breaking and entering)
- Accessing web pages, network services (legally)
- Preliminary network and port scanning
 - Identify running applications and services

Social Engineering

Social engineering refers to manipulation of people into doing certain actions or giving up confidential information

Use psychological principles to convince target to comply

It works, for most part, because most people are trusting and helpful

The weakest link in any security scheme is often the user



PEBKAC

<http://pinterest.com>

More about this in topic about
Human Factors

Physical Intrusion

- Foremost traditional technique of social engineering

Learning schedules of organization

Knowing floor plan of building or buildings

“Baselining” security procedures

Hacker can develop fake ID cards



<https://nikkinicole36.wordpress.com/2013/04/30/boundary-hopping/>

Dumpster Diving

- Attackers look for sales receipts and paperwork containing personal data or credit card information
- (Shredded) documents can lead to data leaks
- Drafts of letters are sometimes left whole in trash
- Company directories, catalogs, unused or misprinted labels and policy manuals
- Old hardware may still contain data



Immutable Laws of Security

- **Law #3: If a bad guy has unrestricted physical access to your computer, it's not your computer anymore**

<https://technet.microsoft.com/library/cc722487.aspx#EIAA>

Scanning Tools

Port scanner

- Examines and reports condition (open or closed) of port and application/service listening on that port, if possible
- Examples: nmap, UnicornScan, masscan

Vulnerability scanner

- Find (and fix) vulnerabilities in remote machines on network
- Software tool that examines and reports about vulnerabilities on local and remote hosts
- Examples: Nesus, OpenVAS/GVM, Qualys, (nmap)



Evolution of Scanners

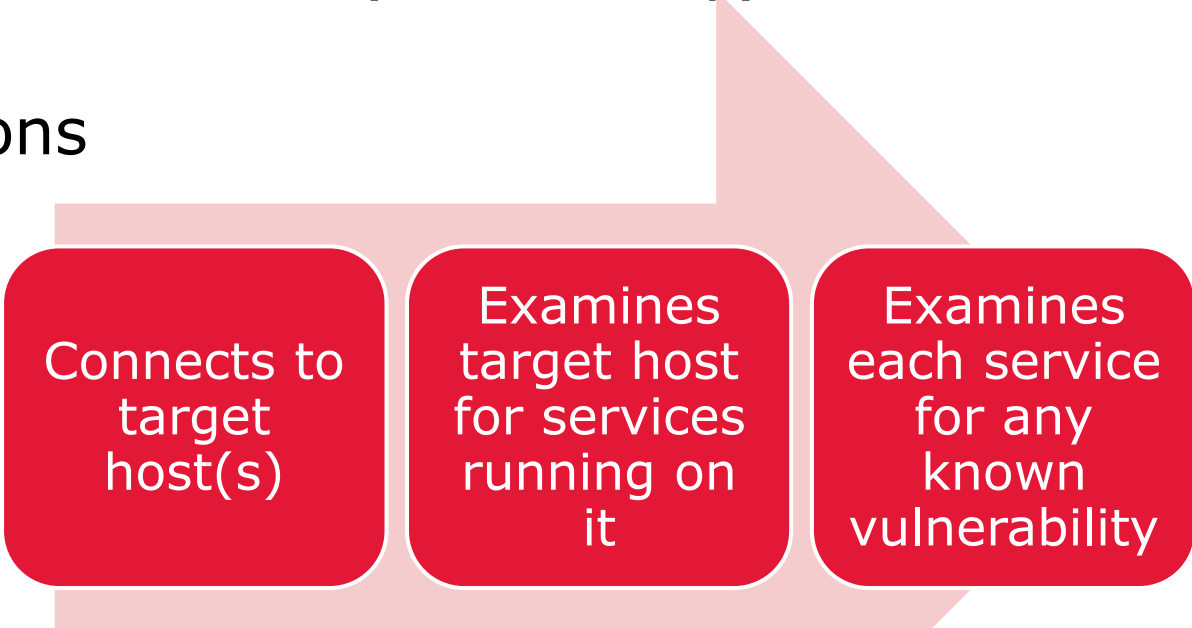
- Scanners first appeared even before ARPANET to monitor connections between mainframes and dumb terminals
- Internet was launched in 1970s
- Early UNIX-like systems had no security at all, legitimate users would connect by dialing specified telephone number with modem
 - Led to invention of new tool, the **war dialer**

<http://csdb.dk/release/?id=99106>



How Scanners Work

- Scanners automate process of examining network weaknesses
- Scanners are not (necessarily) heuristic
- Functions



Connects to
target
host(s)

Examines
target host
for services
running on
it

Examines
each service
for any
known
vulnerability

Simplest Type of Scan

- Ping Scan
 - Demonstrates whether remote host is active by sending ICMP echo request packets to that host
 - Host will respond with ICMP echo response

```
C:\Users\nik>ping 192.168.1.107

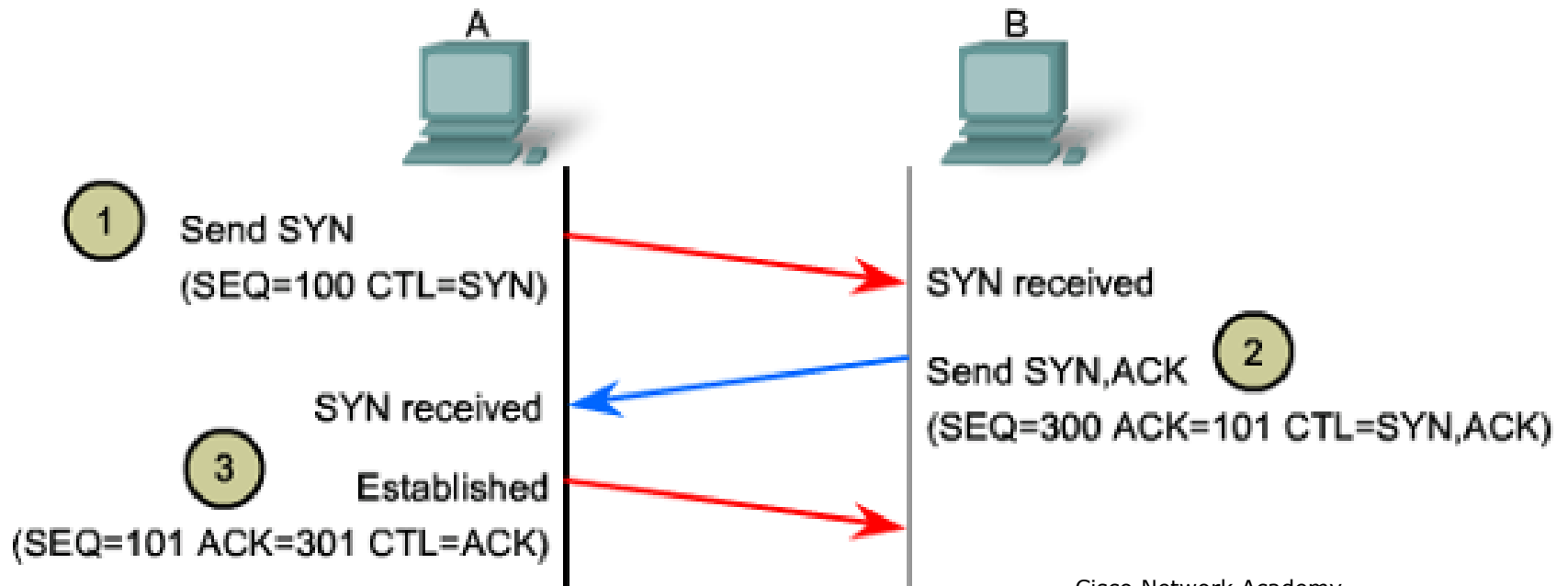
Pinging 192.168.1.107 with 32 bytes of data:
Reply from 192.168.1.107: bytes=32 time=5ms TTL=64
Reply from 192.168.1.107: bytes=32 time=1ms TTL=64
Reply from 192.168.1.107: bytes=32 time=6ms TTL=64
Reply from 192.168.1.107: bytes=32 time=4ms TTL=64

Ping statistics for 192.168.1.107:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 6ms, Average = 4ms
```

Types of Scans

- TCP Connect Scan
 - Attempts to make TCP connections with all/selected ports on remote system
 - Target host transmits connection-succeeded messages for active ports
 - User does not need root privileges to perform TCP connect scanning
 - Almost all IDSs recognize this scanning
- Half-Open Scan
 - TCP connection scan that does not complete the connections

TCP Three-Step Handshake



Cisco Network Academy

Types of Scans (continued)

- Half-Open Scan (continued)
 - Only SYN message is sent from the scanner
 - Reply may be SYN/ACK, indicating port is open
 - Attacker replies with RST to avoid detection
 - More stealthy and efficient, but no application info
 - Many IDSs can detect these
 - Root or system administrator privileges are required to perform half-open scanning (raw sockets)

Types of Scans (continued)

- UDP Scan
 - Examines status of UDP ports on target system
 - Scanner sends 0-byte UDP packet to all ports on target host
 - If port is closed, target host replies with *ICMP unreachable* message
 - ICMP error message may be filtered by firewall on path though and some routers handle ICMP in slow path - retries are needed making UDP scan slower than TCP scan

Types of Scans (continued)

- IP Protocol Scan
 - Examines target host for supported IP protocols
 - Scanner transmits IP packets to each protocol on target host
 - If target host replies with ICMP *unreachable message* to scanner
 - Then target host does not use that protocol

Scanning Without Leaving Trace

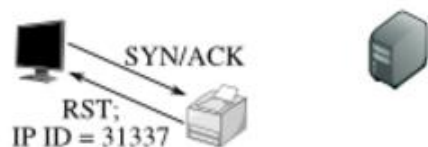
- Idle Scan
 - More difficult but hides scanner's IP address
 - Bounce scan off "zombie host"
 - Utilises IP ID field - IP ID is unique number in each IP packet and in many OS number is increased by one for each packet sent

Scanning Without Leaving Trace

- How Idle Scan works
 1. Pick zombie and probe zombie's IP ID and record it, e.g. send TCP SYN to zombie
 2. Forge SYN packet from zombie and send it to desired port on target; depending on port state, target's reaction may or may not cause zombie's IP ID to be incremented
 3. Probe zombie's IP ID again; target port state is determined by comparing this new IP ID with the one recorded in step 1

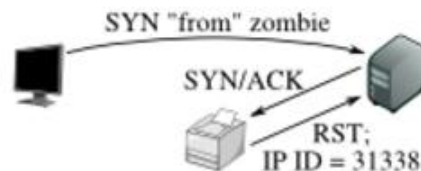
Figure 5.1. Idle scan of an open port

Step 1: Probe the zombie's IP ID.



The attacker sends a SYN/ACK to the zombie. The zombie, not expecting the SYN/ACK, sends back a RST, disclosing its IP ID.

Step 2: Forge a SYN packet from the zombie.



The target sends a SYN/ACK in response to the SYN that appears to come from the zombie. The zombie, not expecting it, sends back a RST, incrementing its IP ID in the process.

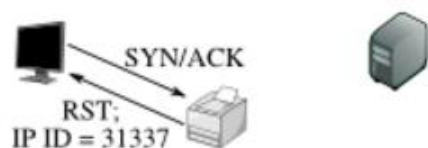
Step 3: Probe the zombie's IP ID again.



The zombie's IP ID has increased by 2 since step 1, so the port is open!

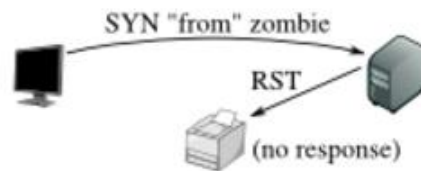
Figure 5.2. Idle scan of a closed port

Step 1: Probe the zombie's IP ID.



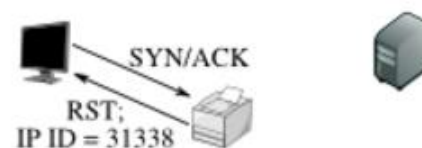
The attacker sends a SYN/ACK to the zombie. The zombie, not expecting the SYN/ACK, sends back a RST, disclosing its IP ID. This step is always the same.

Step 2: Forge a SYN packet from the zombie.



The target sends a RST (the port is closed) in response to the SYN that appears to come from the zombie. The zombie ignores the unsolicited RST, leaving its IP ID unchanged.²⁷

Step 3: Probe the zombie's IP ID again.



The zombie's IP ID has increased by only 1 since step 1, so the port is not open.

Service Identification

- Regular port scans only tell you whether host is listening on given port

PORT	STATE	SERVICE
80/tcp	open	http

- This information is useful but incomplete – there's not enough information to plan attack
- Vulnerability scanners (and some port scanners) can attempt to identify specific software and version running

PORT	STATE	SERVICE	VERSION
80/tcp	open	http	Apache httpd 2.4.6 ((CentOS) PHP/5.4.16)

Service Identification (cont.)

- Service identification isn't always perfect
 - Scanners will look at banners and responses and try to match them against known version strings
 - May just output fingerprint if no match is found
- If no banner or information is returned then service identification won't be possible

```
root@kali:~# nc murdoch.edu.au 80

HTTP/1.1 400 Bad Request
Date: Sun, 01 Mar 2020 13:06:37 GMT
Server: Apache/2.4.25 (Debian)
Content-Length: 365
Connection: close
Content-Type: text/html; charset=iso-8859-1
```

Sniffers (or Packet Sniffers)



Applications that monitor, filter and capture data packets transferred over network

<https://theloadstar.co.uk/watch-out-for-suspicious-packages/>

- Sniffers are nearly impossible to detect in operation
- Can be implemented from nearly any computer

Sniffer Operation

- Sniffer must work with type of network interface supported by OS
- Sniffers look only at traffic passing through network interface adapter on machine where sniffer is running
- Traffic you can see depends on network technology
 - Ethernet (legacy vs. switched)
 - WiFi

Bundled Sniffers

Packaged with specific operating systems, e.g.

Network Monitor comes bundled with Windows

tcpdump comes with many open source UNIX-like operating systems, like Linux

Easy if you are sniffing your own network and you happen to be admin anyway...



Wireshark (Ethereal)

- Probably best-known and most powerful free network protocol analyzer
 - For UNIX/Linux, Windows, ...
- Allows to capture packets from live network and save them to capture file on disk
- Data can be captured off wire from network connection
 - Ethernet, WiFi, IEEE 802.11 FDDI, PPP, token-ring or X.25 interfaces

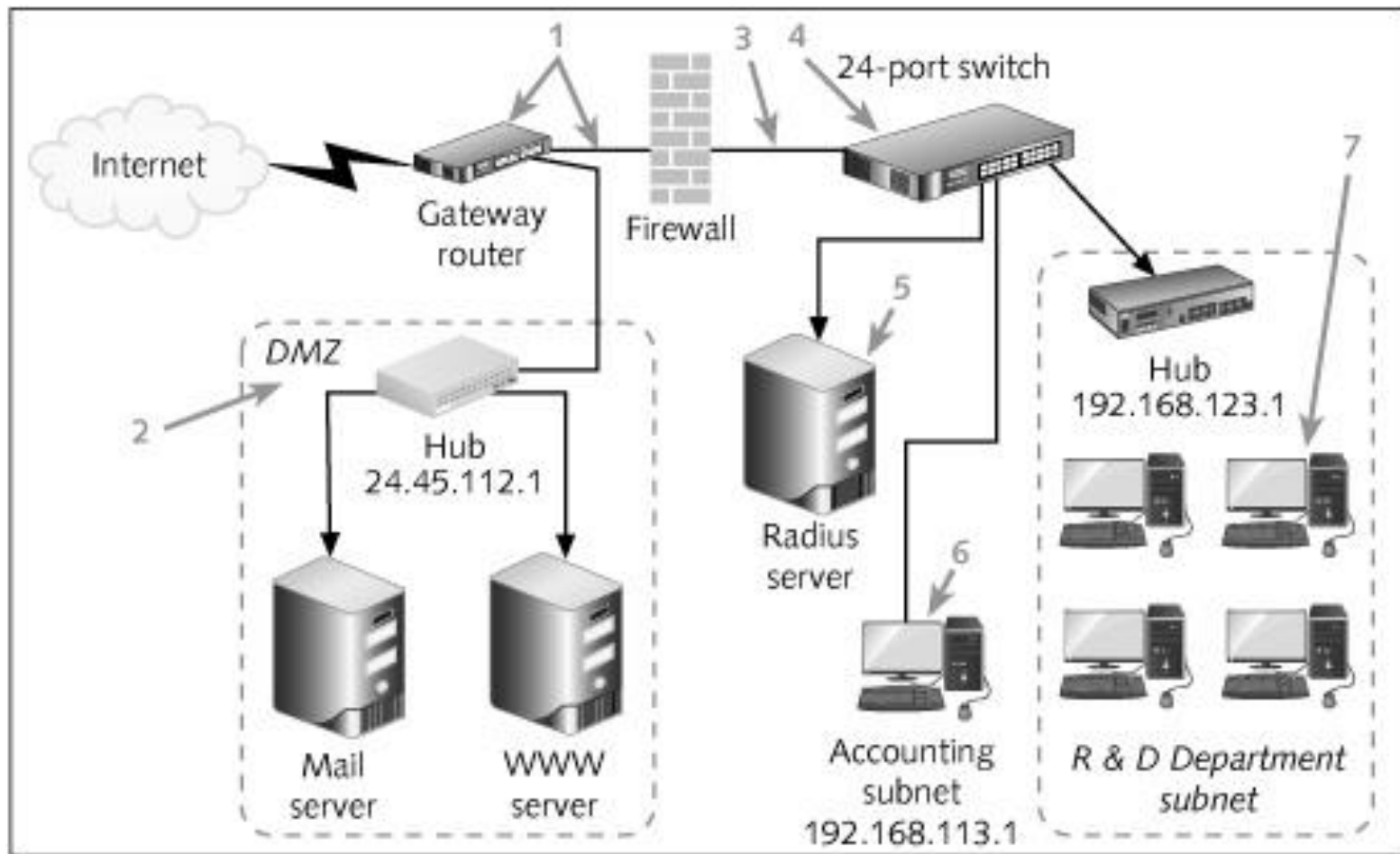
<https://www.wireshark.org/>



Placement of Sniffer

- Can be placed in many locations depending on available access and strategy (only capture required data)
- Sniffers are normally placed on
 - Computers
 - Cable connections (optical, electrical splitters)
 - Routers (based on Linux/FreeBSD)
 - Network segments connected to Internet
 - Network segments connected to servers that receive passwords

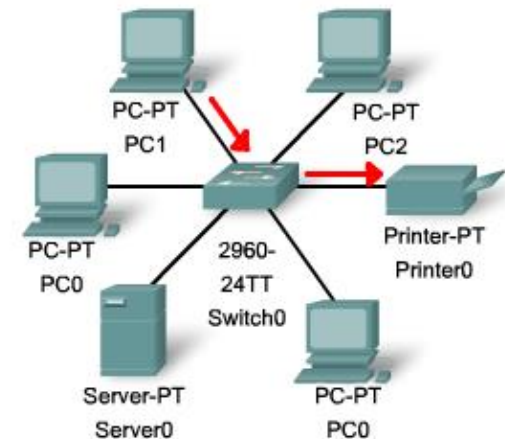
Placement of Sniffer (continued)



© Cengage Learning 2014

Recall: Data Transfer over Switched Ethernet

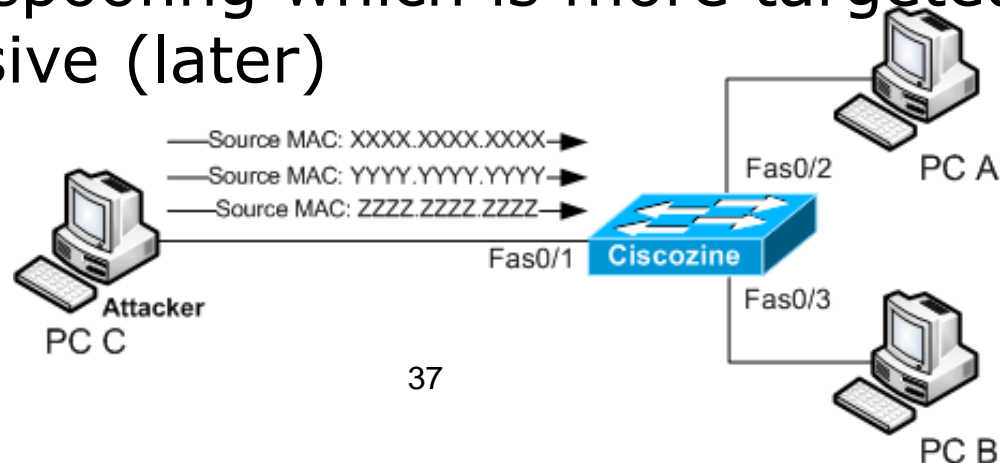
- Ethernet frames are sent to whole LAN segment
- All network adapters on LAN can receive frame
- Every adapter then compares destination MAC address in frame with its own MAC address
 - Every frame not destined for adapter is dropped
- In modern networks, switches only send frame out on port where destination is connected (switch has mapping of port \leftrightarrow MAC address)



But we want to see other traffic as well

Sniffing in Switched Ethernet Networks

- Force switch to flood packets (very invasive)
 - Flood switch with random MAC addresses
 - Switch's MAC table fills with bogus MAC addresses
 - When table fills up, switch will begin sending out received frames on every port
 - Attacker can capture other users' data frames
- Use ARP spoofing which is more targeted and far less invasive (later)



Sniffing in WiFi IEEE 802.11

- Wireless LANs are very common these days
- Replaced wired Ethernet to some degree
- Wireless transmissions can be received by all stations in range of sender
- Can sniff all traffic exchanged
- Monitor network from parking lot
- However, if WiFi security is used (WPA) and sniffer does not know shared secret, sniffer cannot decode frame content

Promiscuous Mode

- ICT169: receiving frames are only passed to network layer if destination MAC in frame is equal to interface's MAC
- Actually, NIC in promiscuous mode can retrieve any data packet being transferred throughout Ethernet network segment irrespective of destination MAC address
- Sniffer puts network card into promiscuous mode by using programmatic interface
- libpcap/winpcap is library commonly used by sniffers

Detecting Sniffer

- Since sniffer technology is passive, so it is difficult to detect sniffers
- You can only (maybe) detect whether suspect is running his or her network interface in promiscuous mode (SniffDet tool)



Protecting Against Sniffer

- Heart of defense against sniffer is to make the data inconvenient to use
- Encourage use of standards-based encryption, such as:

Secure Sockets Layer (SSL/TLS)

Secure Shell (SSH)

IPSec

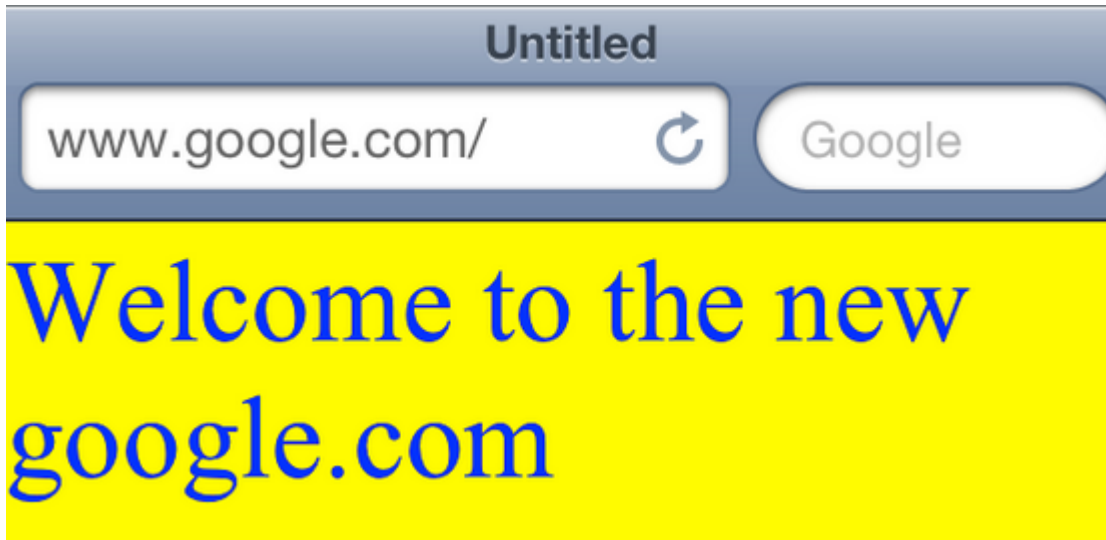
WPA (WiFi)



More Protection

- At OSI layer-2 (link layer)
 - Enable port security on switch
 - Enforce static ARP
- At OSI layer-3 (network layer)
 - IPSec paired with secure, authenticated naming services (DNSSEC)
- Firewalls can be mixed blessing
 - Sniffers are most effective behind firewall, where legacy cleartext protocols are often allowed by corporate security policy

Spoofing



- “Authenticate” one machine to another by using forged packets
- Misrepresenting sender of message to cause human recipient to behave in certain way
- Impacts two critical issues for networked systems
 - Trust
 - Authentication

Spoofting (continued)

Authentication is less critical when there is trust

- Trust and authentication have inverse relationship
- Computer can be authenticated by its IP address or MAC address
- Initial authentication is based on source address in trust relationships

IP addresses or MAC addresses were not designed as authenticators

- Most fields in IP header can be changed (forged)
- Most fields in Ethernet header can be changed (forged)

Types of Spoofing

- Depending on information attacker has
 - Blind spoofing
 - Non-blind/Active spoofing
- Depending on protocol
 - MAC spoofing or ARP (Address Resolution Protocol) spoofing
 - IP spoofing
 - Web spoofing
 - DNS (Domain Name System) spoofing

Blind vs. Active Spoofing

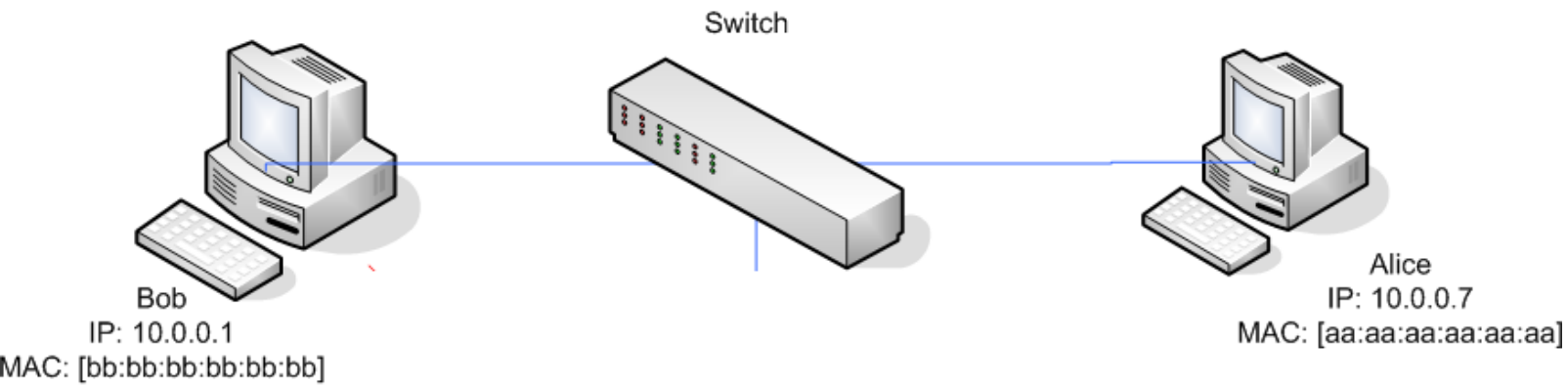
- Blind Spoofing
 - Only one side of relationship under attack is in view
 - Hacker is not aware of how network transmission happens
- Non-blind/Active Spoofing
 - Hacker can see communication of both parties and respond accordingly
 - Hacker can perform various actions, such as sniffing data, corrupting data, changing contents of packets or deleting packets



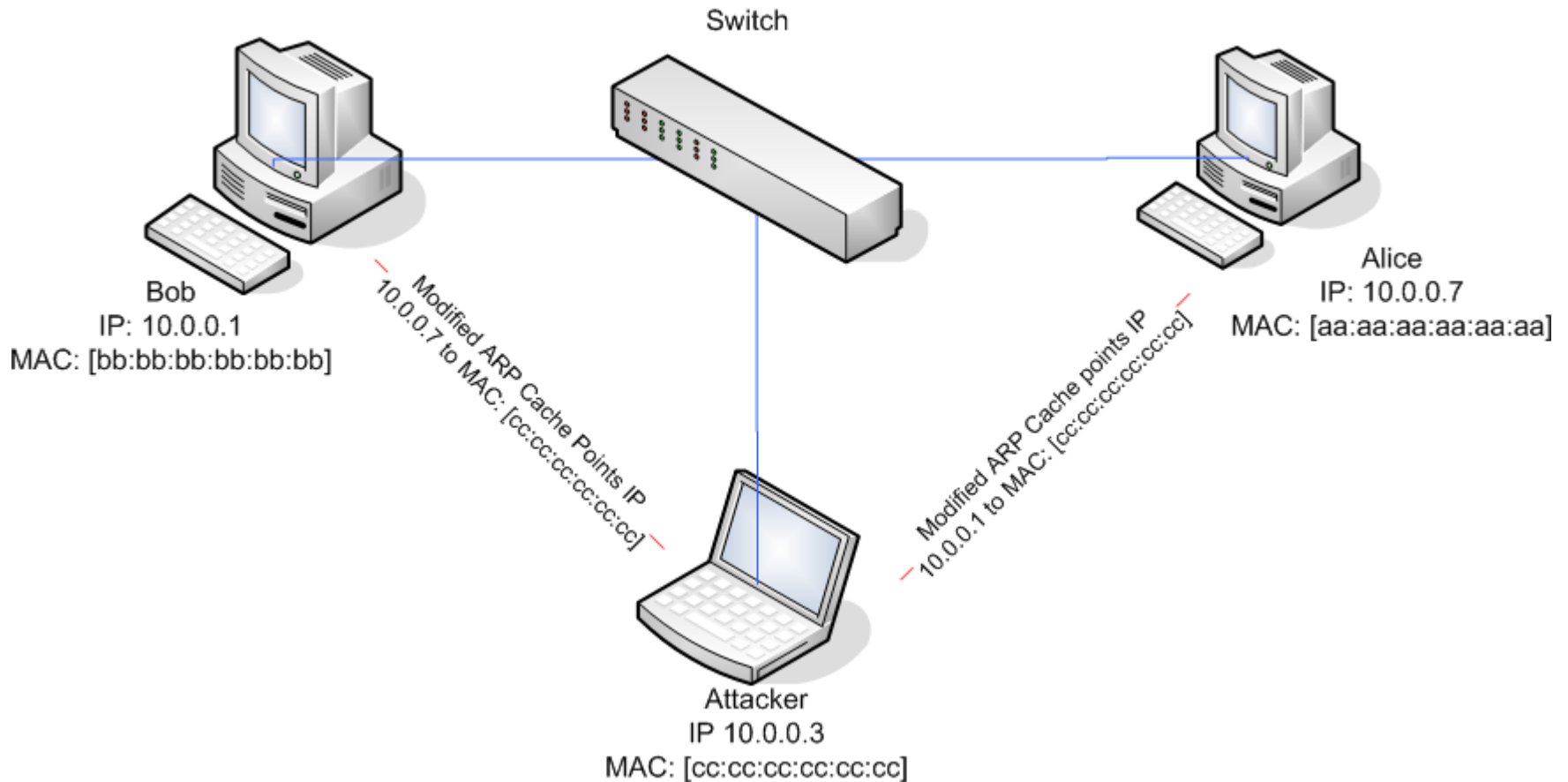
ARP Spoofing

- Modifying Address Resolution Protocol (ARP) table and Media Access Control (MAC) table in switches
 - ARP table: IPs and corresponding MAC addresses
 - MAC table: MAC addresses & corresponding switch ports
- Hosts/routers searches ARP table for destination computer's MAC address, switches search MAC table for outgoing port
- ARP spoofing attack involves detecting broadcasts with target's IP address
 - Then responding with MAC address of hacker's computer
- Also, can generate fake replies (without queries)

ARP Spoofing (continued)



ARP Spoofing (continued)



IP Spoofing Attack

- Spoofing is very easy if we don't care about replies or whether spoofed packets are properly received
 - (D)DoS attack
 - Decoys when port scanning
- In this case use spoofing to hide identity, but could also try to blame somebody else

Process of IP Spoofing Attack

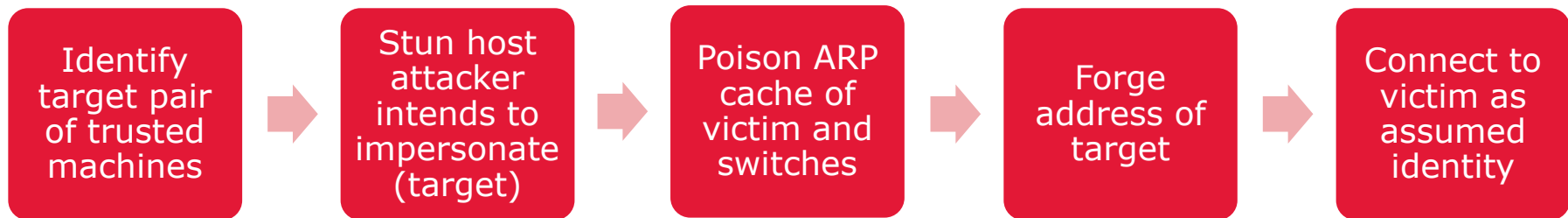
- IP protocol takes care of forwarding packets to destination
 - But IP is unreliable
- Often TCP is used on top of IP, as TCP provides reliable data transport
 - Sequence numbers
 - Three-way handshake
- TCP Initial Sequence Number (ISN) for each direction is chosen pseudo-randomly by sender
- TCP stack will not accept packets with sequence numbers outside certain window

Process of IP Spoofing Attack (continued)

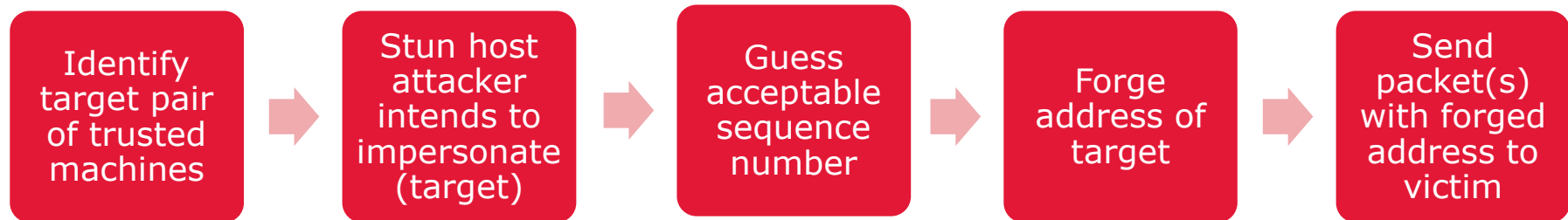
- Successful spoofing attack with TCP requires more than simply forging single header
- **Spoofing whole connection**
 - Requires sustained dialogue between machines for minimum of 3+ packets
 - Typically requires spoofing on L2 as well, e.g. MAC spoofing and ARP spoofing/poisoning
- **Spoofing some packets within existing victim connection**
 - Why? For example, can disrupt existing connection by sending RST or insert crafted packets
 - Need to know/guess sequence number in acceptable range

Process of IP Spoofing Attack (continued)

To spoof trusted machine relationship, attacker must:



Spoof whole connection



Spoof packets within existing connection

TCP Sequence Number Guessing

- In old days relatively easy as initial sequence numbers (ISNs) were predictable (IETF RFC1948) and hackers can connect to victim to see how quickly ISN advances and then make educated guesses
- Modern operating system use approach described in IETF RFC6528 to defend against sequence number spoofing
- Keyed hash function is now used for ISN generation
- Very hard to guess acceptable sequence numbers for off-path attackers (blind attacks)
- But still easy to find acceptable sequence number if attacker can sniff traffic between victim and target

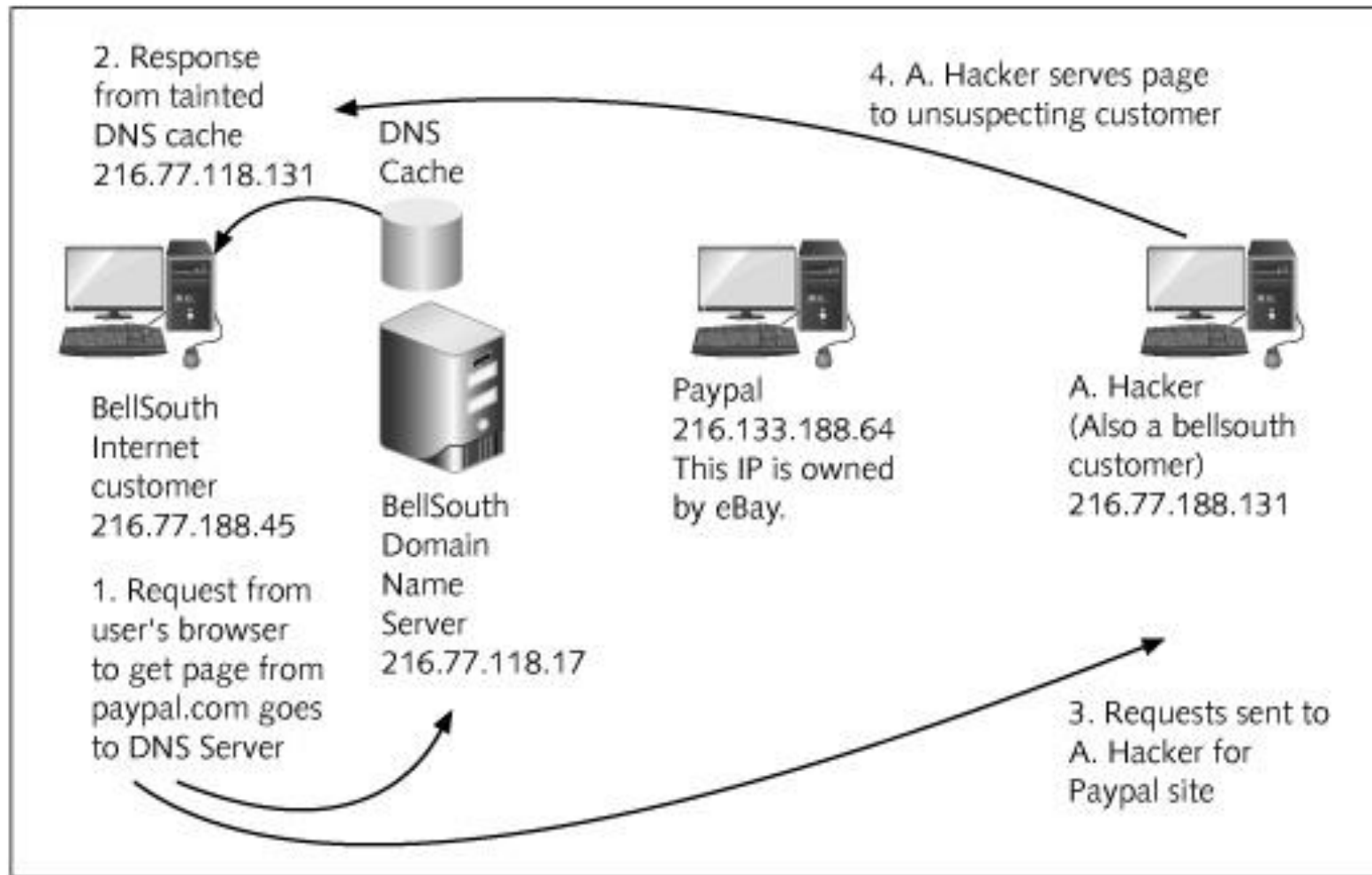
Process of IP Spoofing Attack (continued)

- If you can sniff traffic between target and victim, you know acceptable sequence number space
- Stun host that you want to impersonate with SYN flood (or SYN attack), Ping of Death, or some other denial-of-service (DoS) attack
- Forging IP is as easy as reconfiguring network interface
- Forging MAC address is equally simple (e.g. ifconfig option on Unix)
- ARP spoofing is also easy as there are many tools (https://en.wikipedia.org/wiki/ARP_spoofing)
- After attack done, release target machine

DNS Spoofing/Poisoning

- Hacker introduces corrupt DNS data that links domain name of target to hackers IP address
- Altering IP address directs users/victims to hacker's computer
- Victims are accessing hacker's computer
 - Under impression that he or she is accessing different, legitimate site
 - Attacker can set up fake but real looking web site (and use it for phishing)

DNS Spoofing/Poisoning



© Cengage Learning 2014



<https://freedomhacker.net/google-malaysia-hacked-defaced-bangladeshi-hackers-3959/>

Example of DNS spoofing
from 2013...

Web Spoofing

- Create fake web site with design of target website
- Often has similar URL

Misleadingly named links, e.g. display
`http://www.secure.com`, but lead to
`http://phisher.com`

Cloaked links, e.g.
`http://www.secure.com@phisher.com`

Punycode attacks

- Often used for phishing
- Goes hand in hand with DNS spoofing or man in middle attacks

Prevention and Mitigation

- Wherever possible, avoid trust relationships that rely upon IP addresses (or MAC addresses) only
- Ingress filtering on routers/firewalls (vs. outside attacks)
- Egress filtering on routers/firewalls (vs. inside attacks)
- Use MAC filtering on switches (IP-MAC binding)
- Use secure protocols like IPSec, SSL/TLS
- Use DNSSEC against DNS poisoning
- Limit access to OS, e.g. don't allow configuration of network interfaces, spoofing or editing of hosts file
- On Linux, use TCP wrappers to allow access only from certain systems (ACL for service access)

Prevention and Mitigation (continued)

- To avoid or defend against ARP poisoning
 - Use methods to deny changes to ARP table without proper authorization (standard?)
 - Employ static ARP tables (not scalable)
 - Filter rogue ARP responses
 - Anti-ARP spoofing techniques, e.g. detect multiple entries in ARP table with different IP but same MAC address (ARP snooping)
 - Monitor changes to ARP tables
 - Port Security (limit MAC addresses per port)

Denial of Service (DoS) Attacks

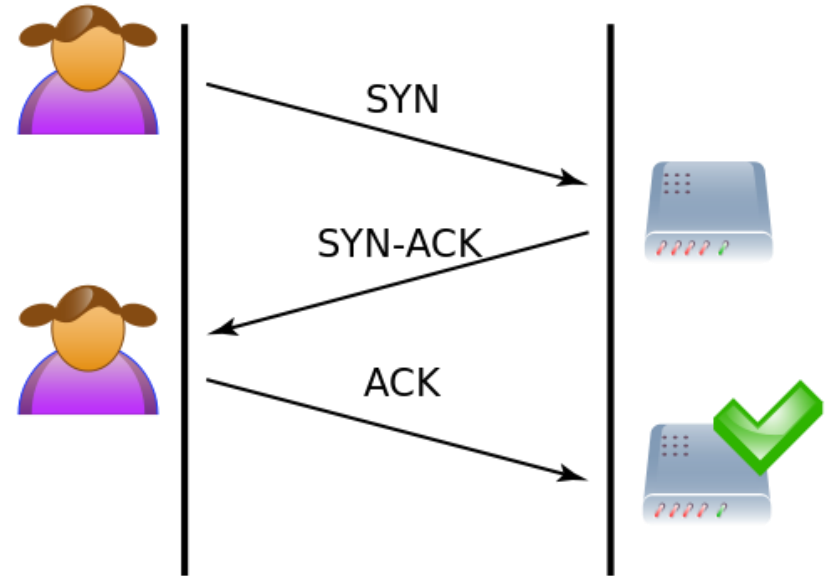
- Deny legitimate users' access to service
 1. Exploit vulnerabilities to render service unusable
 2. Generate lots of network traffic, so legitimate traffic is dropped
- We will focus more on 2. here

DoS: Ping of Death

- Very old, but famous DoS attack
- Sending large ICMP message would crash many OS's
- Fixed in all Operating systems around 1998
- Many system administrators started blocking all ICMP messages at firewall

DoS: TCP SYN Flood

- Under normal operation, TCP connection is setup with the three-way handshake
 - SYN
 - SYN-ACK
 - ACK



http://en.wikipedia.org/wiki/SYN_flood

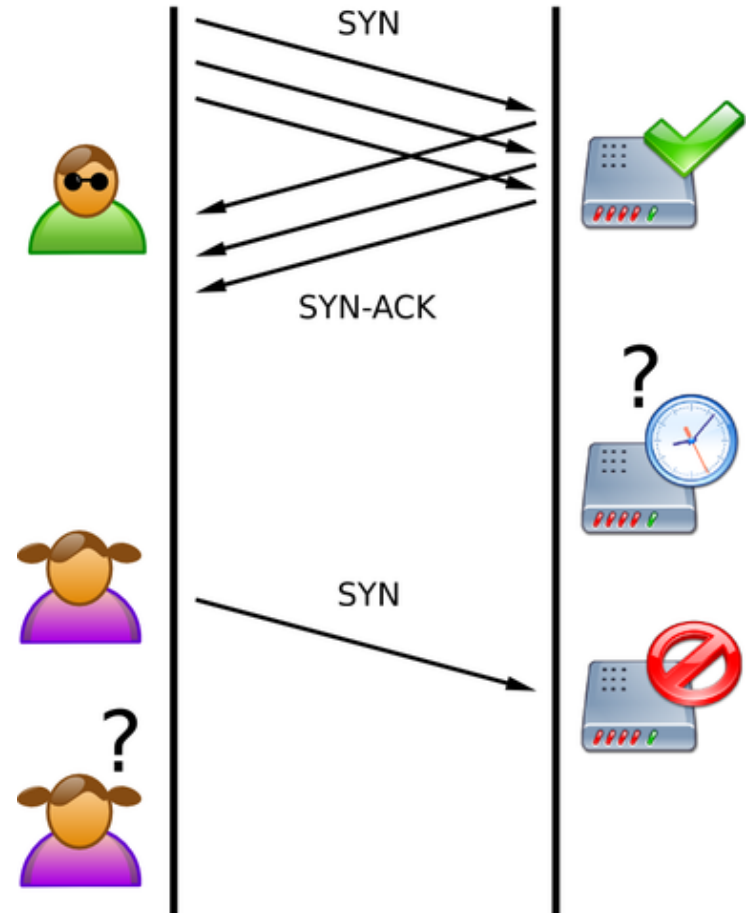
9000.pcap [Wireshark 1.6.2]

Filter: Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.16.0.10	192.168.0.102	TCP	74	38729 > http [SYN] Seq=0 Win=17920 Len=0
2	0.100985	192.168.0.102	172.16.0.10	TCP	74	http > 38729 [SYN, ACK] Seq=0 Ack=1 Win=0 Len=0
3	0.101009	172.16.0.10	192.168.0.102	TCP	66	38729 > http [ACK] Seq=1 Ack=1 Win=17920 Len=0
4	0.108011	172.16.0.10	192.168.0.102	HTTP	181	GET /fil [Packet size limited during capture]
5	0.208002	192.168.0.102	172.16.0.10	TCP	66	http > 38729 [ACK] Seq=1 Ack=1 Win=17920 Len=0

DoS: TCP SYN Flood

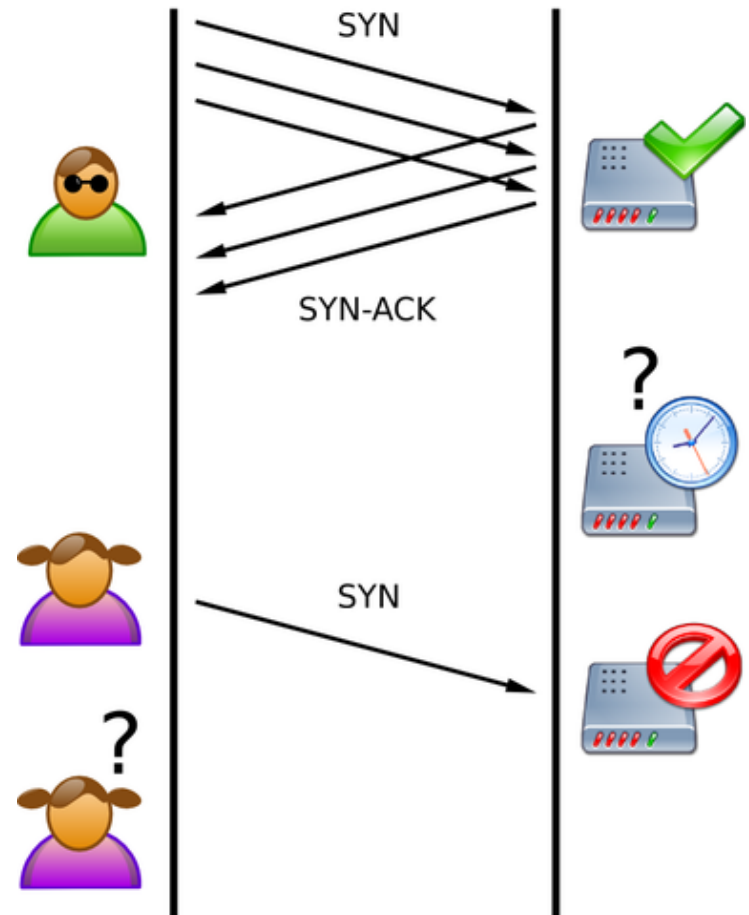
- TCP SYN Flood attack involves continually sending SYN messages
- Attacker never responds with final ACK, so server has huge number of half-open connections
- Half-open connections only time out after some time
- Will consume all resources on server and new legitimate connections won't be answered
- Typically attacker spoofs source IP address



http://en.wikipedia.org/wiki/SYN_flood

DoS: TCP SYN Flood

- Weakness discovered in 1994
- Various solutions such as flood monitoring/limiting on firewall/IDS and TCP SYN cookies
 - SYN cookies: no state created on SYN; instead state encoded in SYN-ACK and created on final ACK
- See RFC 4989: "TCP SYN Flooding Attacks and Common Mitigations"



http://en.wikipedia.org/wiki/SYN_flood

DoS: DNS Amplification

- DNS can be used for amplification DDoS attack
- DNS uses UDP – easy to send request with spoofed source IP set to target's IP
- Problem: open DNS resolvers, resolvers that respond to any clients
 - Same issue as with spam and open mail servers/relays
- DNSSEC perversely makes this worse as keys are large

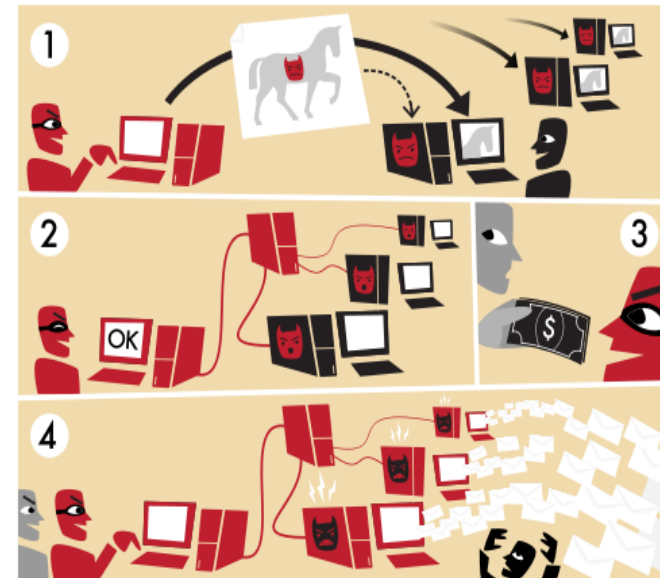
dig ANY isc.org @x.x.x.x

64 byte request results in response of over 3kB

;; QUESTION SECTION:
;isc.org. IN ANY
;; ANSWER SECTION:
isc.org. 4084 IN SOA ns-int.isc.org. hostmaster.isc.org. 2
isc.org. 4084 IN A 149.20.64.42 isc.org. 4084 IN MX 10 r
isc.org. 4084 IN MX 10 mx.ams1.isc.org.
isc.org. 4084 IN TXT "v=spf1 a mx ip4:204.152.184.0/21
isc.org. 4084 IN TXT "\$Id: isc.org,v 1.1724 2012-10-23 0
isc.org. 4084 IN NAPTR 20 0 "S" "SIP+D2U" "" _sip._udp
isc.org. 484 IN NSEC _kerberos.isc.org. A NS SOA MX TXT
isc.org. 4084 IN DNSKEY 256 3 5 BQEAAAAB2F1v2HWzC
...

Botnets and DDoS

- **Distributed Denial of Service (DDoS)** is DoS attack that comes from many sources
- Possible that these DDoS attacks may be done by collective group
- More often they come from **Botnet**
- Botnet is group of compromised hosts that will perform functions requested by Botnet handler
- More on botnets later



<http://en.wikipedia.org/wiki/Botnet>

(D)DoS Prevention and Mitigation

- Keep OS including network stack up to date with latest patches to fix vulnerabilities
- Prevent intrusion attacks
- Network (D)DoS attacks
 - Usual defense is to establish baseline for normal conditions (normal traffic patterns)
 - Then detect abnormal conditions, e.g. high traffic load, and filter traffic – also known as “traffic scrubbing”

Threat Modelling

- Identify threats to system in order to develop mitigations against them
- Risk management approach to develop secure systems
- Threat is circumstance or event that potentially compromises security and creates loss
- Loss can be damage to equipment/environment, monetary loss, reputational loss, IP loss
- Goals
 - Understand assets most desired by attackers
 - Understand most likely attack vectors
 - Use this understanding to safeguard system

Threat Modelling Methodologies

- Many different methodologies and we will only cover some here
 - STRIDE
 - DREAD
 - CVSS
 - PASTA
 - Attack Trees

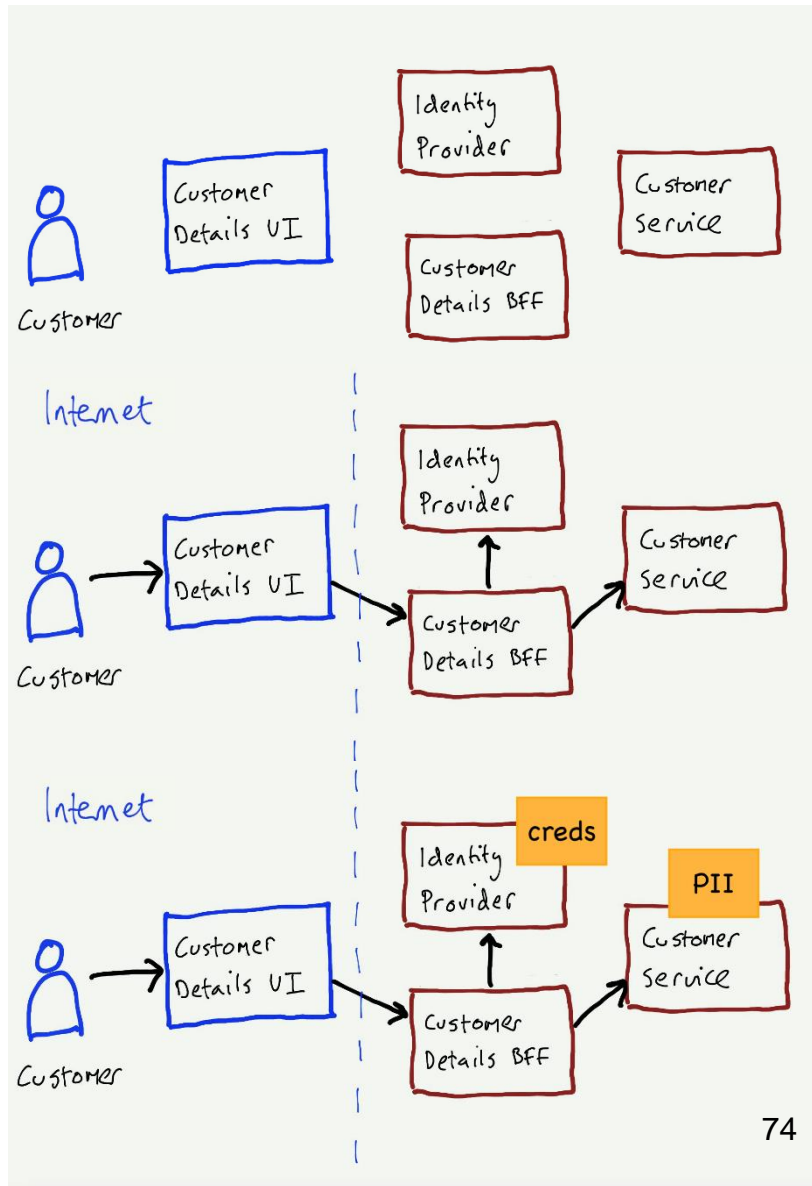
Broad Threats vs Technical Threats

- Broad threats emerge from world at large and are uncertain and hard to predict, this includes hacker groups, hacktivists, unhappy employees, human error, pandemics, new malware
- Technical threats are more fine-granular, such as particular weaknesses and vulnerabilities in software, missing security controls, flawed encryption
- As software dev it is easier to start from technical threats but don't forget bigger picture

Threat Modelling Process

- Most important to understand overall process
 - What are you building? Technical diagrams
 - What can go wrong? List of threats
 - What are you going to do? List of prioritised fixes
- Try to involve whole team (technical and non-technical) as this brings more perspectives and builds shared understanding
- Frequency depends on organization and processes – with agile methodology you could do threat modelling in every sprint
- <https://martinfowler.com/articles/agile-threat-modelling.html>

What are you building?



Building Blocks

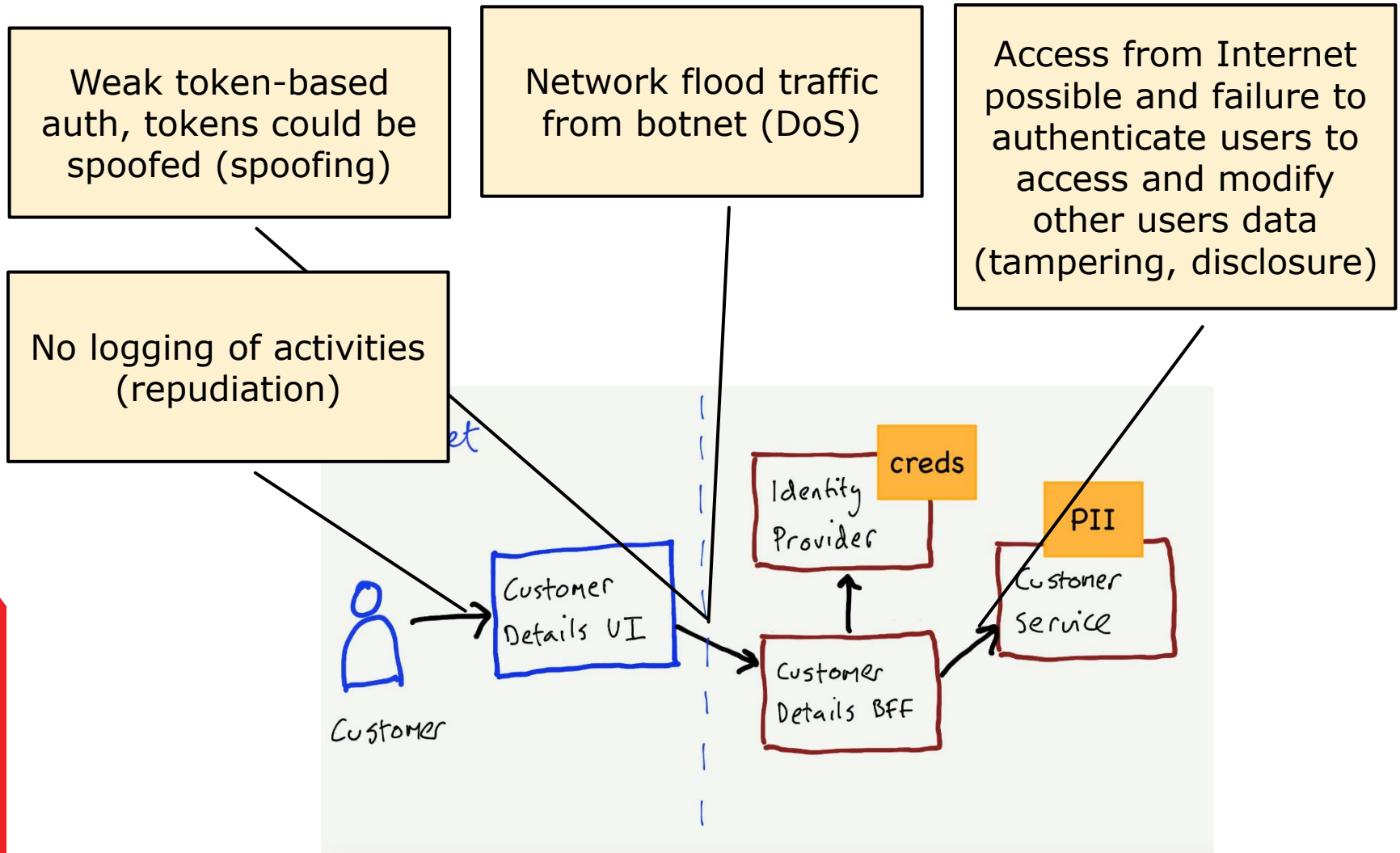
Data Flows

Assets

What can go wrong?

- **STRIDE** models threats into 6 categories
- **Spoofing**: using another user's authentication creds (violate authenticity)
- **Tampering**: modification of data (violate integrity)
- **Repudiation**: deny performing actions without other parties being able to prove otherwise (violate non-repudiation)
- **Information disclosure**: unauthorized access to confidential data (violate confidentiality)
- **Denial of service**: deny service to legitimate users (violate availability)
- **Elevation of privilege**: unprivileged user gains privileged access (violate authorization)

What can go wrong?



What are you going to do?

- Need to rank threats, identify fixes/mitigations and add to backlog
- **DREAD** can be used, but it has issues and has been abandoned by creators
- **Damage**: how bad would attack be?
- **Reproducibility**: how easy is it to reproduce attack?
- **Exploitability**: how much work is it to launch attack?
- **Affected users**: How many people will be impacted?
- **Discoverability**: how easy is it to discover the threat?
- CVSS is another approach

Common Vulnerability Scoring System (CVSS)

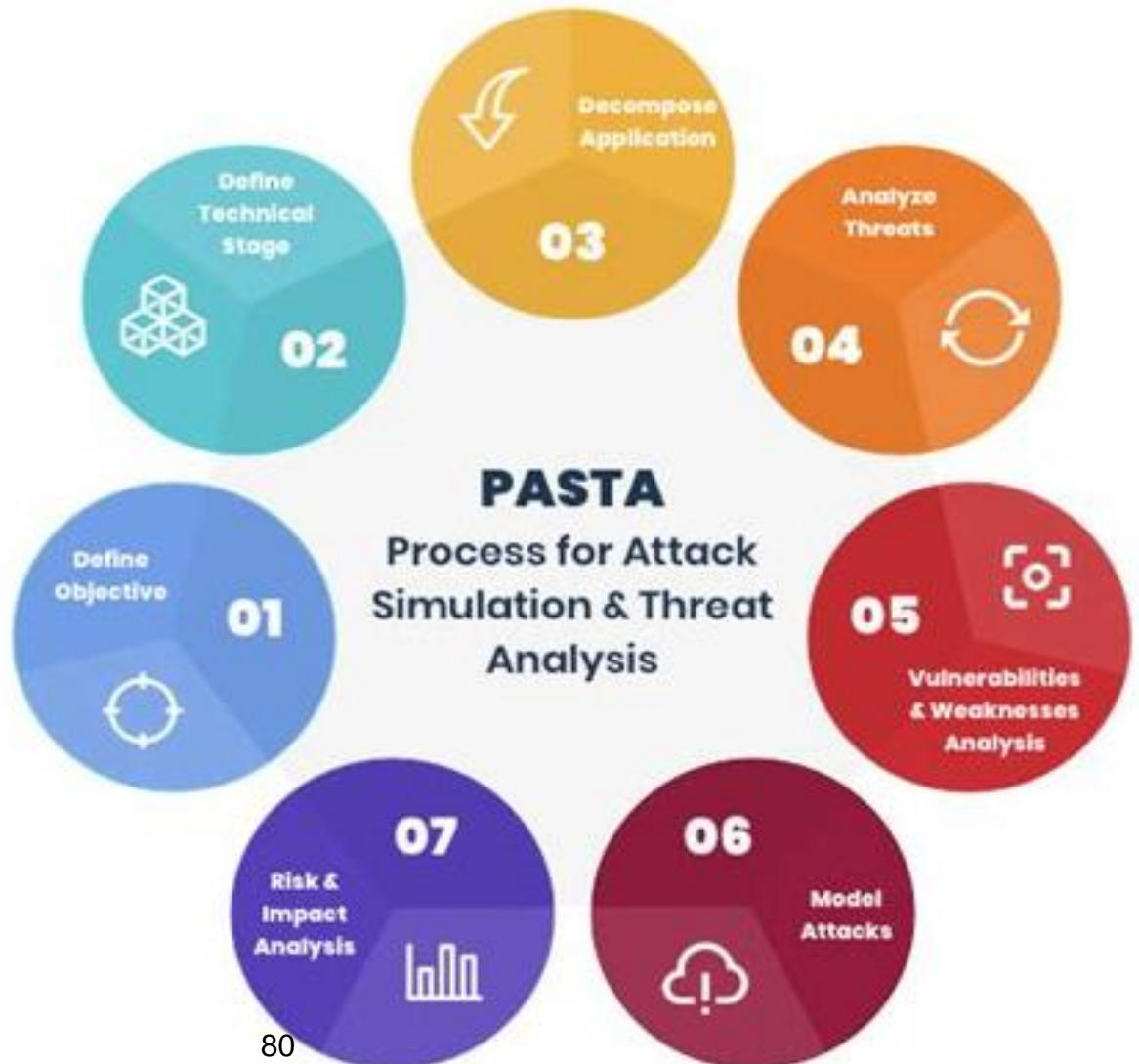
- Captures characteristics of vulnerability and produces numerical score reflecting its severity based on
 - Attack Vector
 - Attack Complexity
 - Confidentiality
 - Integrity
 - Availability
 - ...
- Numerical score can be translated into qualitative score (low, medium, high, and critical) to help assess and prioritize vulnerability management
- Published standard used by organizations worldwide and for Common Vulnerabilities and Exposures (CVE) register

What are you going to do?

- For lightweight approach, I think it is most useful to focus on damage/loss, exploitability and affected users
- Rather than having formula and creating single risk rating, alternative approach is to let everybody in team vote for riskiest threats
- Everybody gets fixed number of votes (smaller than number of threats) and votes for riskiest threats
- Tallying votes allows to identify most critical threats to start working on
- For each prioritized threat identify next steps such as fix/mitigation, dev activities, time frame, test criteria

PASTA

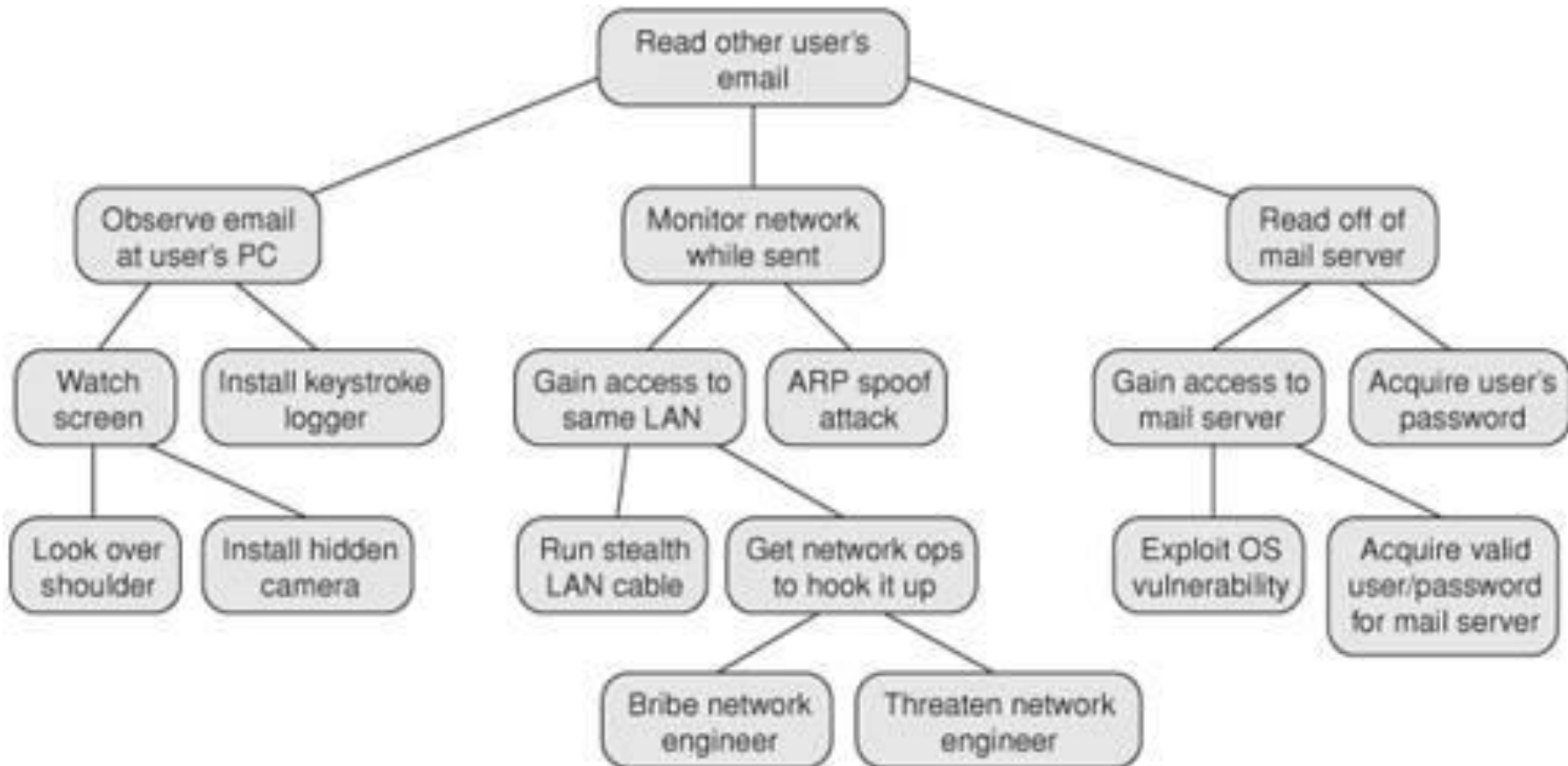
- Seven Stage Process for Attack Simulation and Threat Analysis (PASTA)
- Formal and rigorous but not very light-weight



Attack Trees

- Attack tree shows goals of attack (related to target asset) and events under which attack is successful
- Child nodes are events of which at least one must be satisfied to make parent node true (logical OR)
- Leaf nodes are directly achievable events
- Attack is complete when root node is satisfied
- Based on fault trees used in various domains and increasingly used in Cyber Security
- Great for understanding hierarchy of events/conditions that are required for attack, especially complex attacks
- Can be used to model cost of attack but we won't look at this here

Attack Trees Example



<https://flylib.com/books/en/3.211.1.42/1/>

Lecture Summary and Week Ahead

- Recon & scanning activities
 - Scanners: port & vulnerability scanners
 - Sniffers: placement, detection, role
 - Spoofing attacks and prevention/mitigation
 - DoS attacks and prevention/mitigation
 - Threat modelling
-
- Labs: Network Security, Scanning, Threat Modelling
 - Next week: Software Security