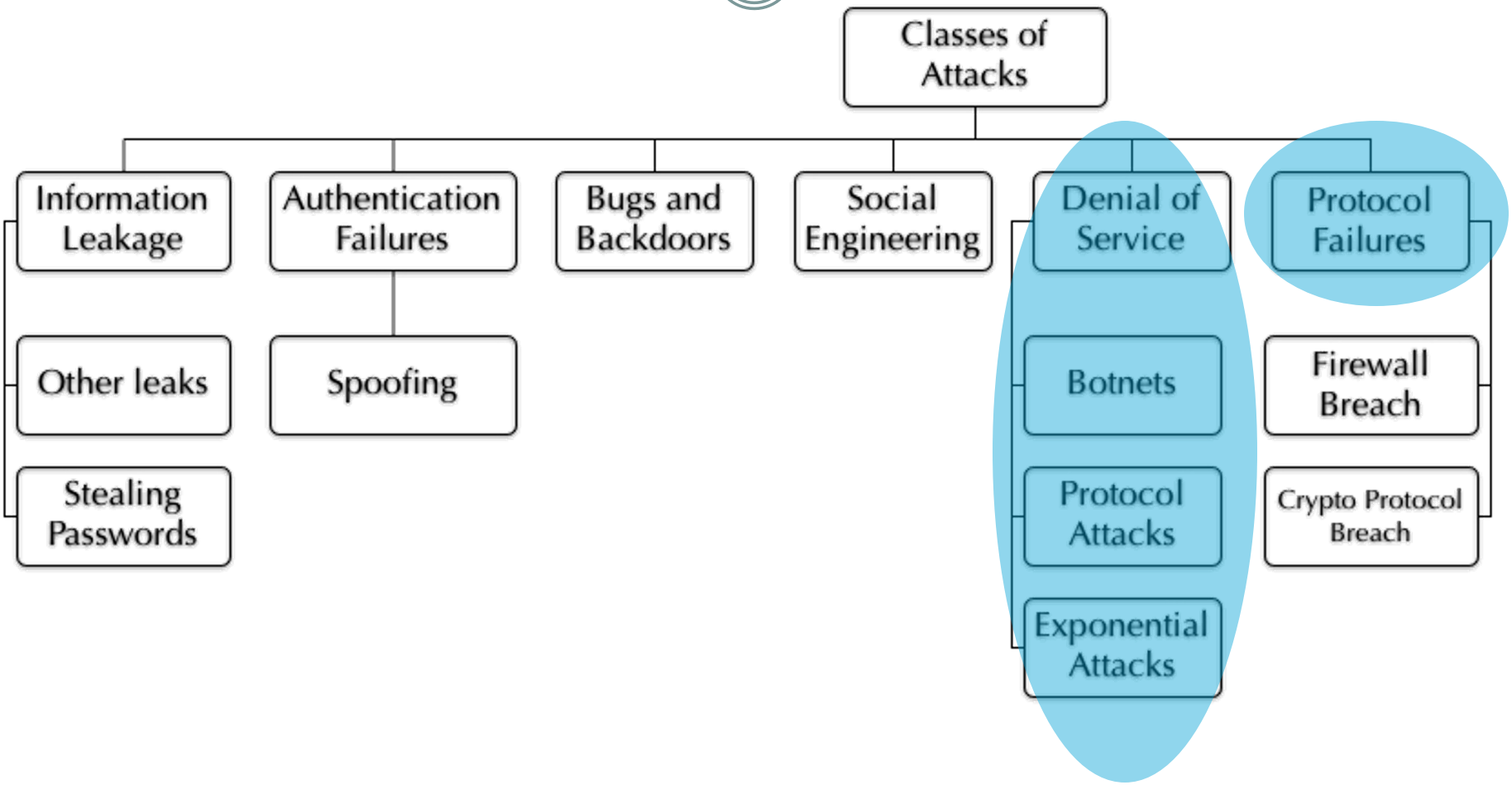
Lecture 3

SPECIFIC ATTACKS

Classes of Attacks



Some Remarks

- Specific attacks and vulnerabilities in protocols will change with time
  - New protocols will be used, new attacks will be developed and new vulnerabilities detected
  - It is impossible to be aware of and know all possible specific attacks
  - Steps taken to mitigate or protect against an attack may affect services - **risk Vs cost**
    - Example TCP and DNS
- Need a **common sense approach**
  - **Prevention** is important but not sufficient
  - **Detection** and response are important

Some Remarks - II

- Objectives of looking at some specific attacks
  - To get an idea of how very obvious simple protocols can be abused by people with malicious intent
  - To understand that it may be necessary to know all the nitty-gritty details of protocols to figure out how attacks are being launched
  - To realize the importance of some of the mitigation/protection schemes to be discussed in later classes

Sources for finding out about new vulnerabilities and incidents

- A very useful source of information is CERT
  - https://www.kb.cert.org/vuls/
  - Lists latest vulnerabilities, incidents, fixes and suggestions to minimize damage
- Thai-CERT - The Thai Computer Emergency Readiness Team
  - https://www.thaicert.or.th/
  - SecurityFocus - a vendor neutral site
  - https://www.securityfocus.com
- Packet Storm
  - http://www.packetstormsecurity.org/
- Common Vulnerabilities and Exposures
  - https://cve.mitre.org/

Specific Attacks

- Protocol Failure Attacks
  - Abusing ICMP
  - TCP SYN flood attack and TCP Sequence Number Attack
  - SMTP Flooding Attack
- Denial of Service (DoS) and Distributed DoS (DDoS) Attacks
- DNS Attacks
- Email Attacks
- Malicious Programs

Protocol Failures

- Definition
  - A protocol does not take into account potential misuse correctly in its design and fails to meet objective
- Causes
  - Poor design (e.g. insufficient randomness in generating TCP sequence numbers)
  - Pitfalls that are hard to detect except **by experience**

Protocol Failures in IP

- IPv4
  - IP addresses used as authenticators
    - IP addresses were meant to be end-point identifiers for routing, not as authenticators for services
- IPv6
  - Renumbering from IPv4 to IPv6
    - No process to differentiate between valid and fraudulent renumbering
  - Access control
    - There are many types of IPv6 addresses and using them as authenticators can be dangerous

Protocol Failures in TCP

- TCP 3-way handshaking needs to SYN to allocate resources
- TCP sequence number attacks for **session hijacking**
- TCP Reset (RST) is implemented but not widely used

Attacks on TCP

- Many types of attacks are possible using TCP
- Information leakage
  - Scan for open ports to detect available services with known vulnerabilities
- Denial of service
  - Exploit TCP behavior to deny services to legitimate connection requests
- Session hijacking
  - Allow a legitimate connection to be created and then insert yourself into the connection

TCP SYN Flood Attack

- TCP is connection oriented
  - There is a 3-way handshake and it keeps track of the state (sequence numbers, window sizes etc.)
  - The state has to be created for each connection - data structures such as socket, protocol information, time to live, checksum etc.
  - Most systems introduce a limit on the amount of memory that can be allocated to connection states
- SYN Flood Attack
  - Oscar has no intention of completing the three-way handshake
  - His goal is to exceed the limits set on the number of connections waiting to be given service
  - This way, any new connection requests will be dropped

How it works

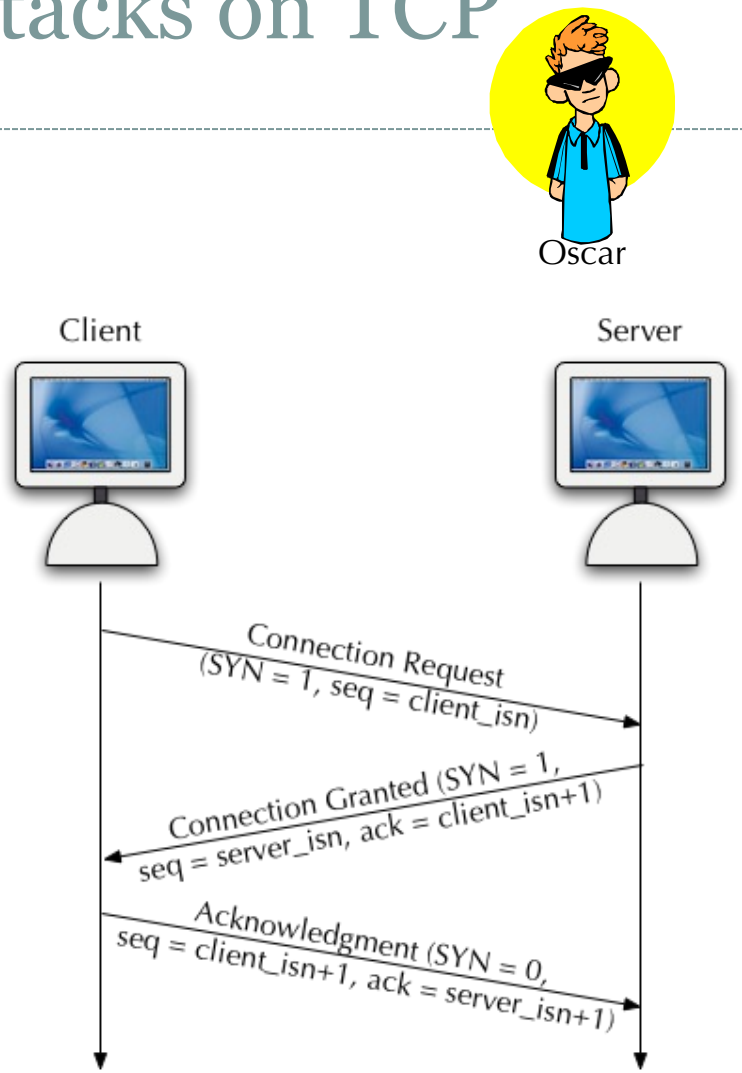
- Each port can support only so many half-open connections
  - Such connections are dropped after a timeout
- Oscar sends many TCP SYN segment to the victim host using spoofed IP addresses
  - The victim responds with SYN ACKs and waits for the final ACKs
  - The final ACKs never arrive creating a backlog of open connections
  - New connections are dropped till the queue shortens but a flood of SYN packets ensures that the queue is never below the threshold for accepting connections
- Real-life example
  - Attack in February 2000 on Yahoo! and other web servers

Some remarks

- Brute force SYN flood attacks send as many SYN packets as possible per unit time
  - Many intrusion detection systems can detect such attacks rapidly
- Elegant SYN flood attacks
  - Disguise the packets as if they are legitimate
  - Make sure that the source address is actually routable so that it cannot be blocked
- It is hard to stop SYN flood attacks completely
  - One solution is to increase the number of open connections allowed by adding memory and resources at the web server
- SYN floods have been used for session hijacking - Mitnick Attack

Sequence number attacks on TCP

- Let us revisit the connect process
- Suppose Oscar wants to masquerade as a legitimate client from outside a LAN
  - He sends a SYN packet with a spoofed IP address to the server
  - The server responds to the legitimate client with the SYNACK
  - To complete the connection, Oscar has to know what server_isn is
- Oscar has to either sniff the LAN or guess the sequence number using information leaks



Guessing Sequence Numbers

- Many TCP implementations use predictable ways of generating sequence numbers
  - Old versions of Berkeley implementation used to increment the sequence number 128 times a second
  - The recommendation in the TCP specification is to increment it 250000 times a second
- The idea is that the round trip time measured or predicted by Oscar will be random enough to prevent him from guessing the sequence number
- Oscar can still guess a range of sequence numbers and send several packets back to the server - at least one will be correct
  - Similar to the birthday attack on DNS

Guessing Sequence Numbers -2

- The random number generator can be reverse engineered under certain circumstances
- Collect previous sequence numbers
- Subject them to analysis
  - Many types of analyses exist
  - Phase-space analyses
- In some cases, with knowledge of three prior sequence numbers, Oscar can guess the next one with 100% probability

See Michael Zalewski's work at
http://alon.wox.org/tcpseq.html
http://lcamtuf.coredump.cx/newtcp/

Attack Feasibility of Different OSs Preliminary results

OS	Feasibility
Win2k/XP	12%
Solaris	0.02%
Mac OS X	0%
Cisco IOS	0%

Mitnick Attack

- This attack used SYN floods and session hijacking together
- Idea:
  - Allow a legitimate connection to be set up between a client and a server
  - Flood one of the parties with SYN packets thereby making them unavailable for response
  - Masquerade as the party that has been silenced by the SYN flood
- Mitnick first probed the target to determine who is logged on
  - Used finger, showmount and rpcinfo
  - Most sites block finger and rpcinfo from outside hosts
  - Mitnick used these to determine the way TCP sequence numbers were created by the target

Mitnick Attack 2

- Steps:
  - 1: Send a TCP SYN packet with a spoofed IP address to an X-terminal
  - 2: X-Terminal replies with a SYNACK (it trusts the spoofed IP address)
  - 3: The server at the spoofed IP address is under a SYN flood attack. So it cannot respond with a RST
  - 4: Generate a valid ACK (need to guess `server_isn`)
  - 5: Send commands to install backdoors in the system
- What did this attack use?
  - Address based authentication <bad>
  - Known characteristics of TCP - DoS, sequence numbers that can be guessed

Exploiting TCP Behavior

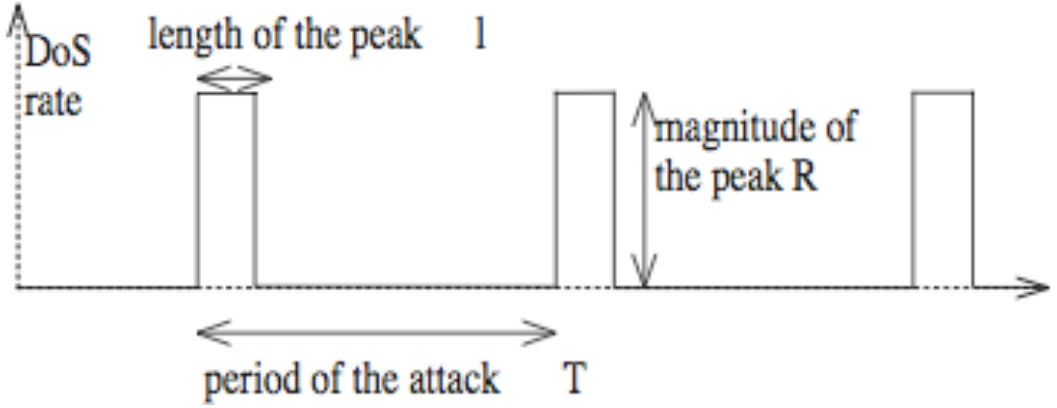
- TCP behavior can be exploited to throttle throughput of sessions
- TCP has two time-scales for congestion control
  - Additive Increase Multiplicative Decrease on the order of 10-100 ms
  - Severe congestion - on the order of seconds
- In the second case, the congestion window is set to 1
  - TCP tries to transmit one lost packet roughly every second (retransmission time-out - RTO)
  - The RTO is doubled if subsequent losses occur

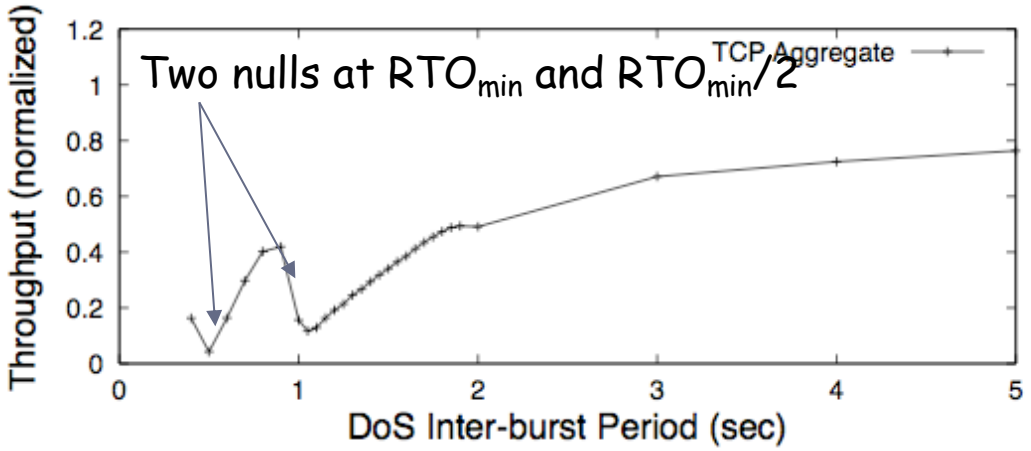
Low Average Rate Attacks on TCP[1]

- Idea
  - Force TCP into the second timescale by inducing severe packet losses at a router
  - Introduce a DoS flow with a periodic burst equal to the TCP RTO value
- Impact
  - If the DoS burst tracks the RTO value, the TCP flow's throughput is reduced to zero
  - If it is approximately the same as the RTO, the throughput still degrades

[1]Low Rate TCP-Targeted DoS Attacks, A. Kuzmanovic and E. W. Knightly, SigComm, 2003

Some example results[1]





- The rate of DoS need not be large
  - Sledgehammer Vs Shrew
- As the period increases, the rate of DoS reduces
  - Impact can still be high
- Affects individual TCP flows as well as aggregates

[1]Low Rate TCP-Targeted DoS Attacks, A. Kuzmanovic and E. W. Knightly, ACM SigComm, 2003

Protocol Failure in ICMP

- ICMP is commonly used in network devices and hosts
- ICMP broadcast is allowed from outside network
- IP broadcast address is allowed for ICMP
- Echo request and reply can be used easily by any device or host
- ICMP Route Redirect message can be misused

ICMP Attack (Flooding)

- Smurf Attack
  - Given the following conditions
    - An intermediate network allows ICMP broadcast requests from outside
    - Intermediate network has many hosts and high bandwidth
    - Target site has low bandwidth
  - Oscar can send an ICMP broadcast request to all nodes in the intermediate network
  - They amplify the ICMP request with responses that flood the victim network resulting in denial of service
- Tribe Flood Network (TFN)
  - This is a DDoS attack with zombies that perform TCP SYN Floods, Smurf or UDP flooding attacks
  - Communication between Master and Zombies is through ICMP echo replies

ICMP Attacks (Malicious Code)

- Winfreeze
  - Causes a susceptible host to attack itself
  - Makes use of the ICMP route redirect message
  - The victim gets a message saying that the optimum route to some random IP addresses is the victim itself!
  - Works on some vulnerable Windows NT hosts
- Loki (ICMP Tunneling)
  - Loki is the Norse god of trickery and mischief
  - Loki installs itself as a server on a compromised host and uses ICMP as the tunneling protocol to communicate with clients
  - Loki server could send the client all kinds of files from the compromised host

ICMP Attack (Information Gathering)

- Trace Route
  - The traceroute command is used to discover the routes that packets actually take when traveling to their destination.
- Port Scanning
  - ICMP Error Messages (Protocol/Port Unreachable) can be used to find out the open ports to an IP address or a LAN segment.
- OS Fingerprinting
  - Fingerprinting is a technique to find out what kind of OS the server is running by looking at the response of the ICMP packet.
- ICMP Route Discovery
  - The ICMP router discovery protocol will discover the IP address of the neighboring routers. The ICMP router discovery messages are called “Router Advertisements” or “Router Solicitations”.

SMTP Protocol Failure

- No authentication for SMTP
- SMTP agent sends message to SMTP server without the human intervention
- Receiving Mail Servers are interested in the receiver's email address, not the sender's email address

SMTP Flooding Attack

- Attackers can misuse this by masquerading to be a legitimate sender
- Attackers can attack receivers with a Denial of Service attack by filling the receiver's mailbox with a large-size attachment or many of emails known as SPAM
- Source Email's address spoofing to lure victims to follow the instructions or links in the Email
  - Called Phishing Attack

Other Common Protocol Failures

- WEP = Wired Equivalent Privacy
  - Uses a stream cipher where key stream can be repeated often
  - Uses CRC for checking authenticity
  - Easy to decrypt, modify and inject traffic
- MIME
  - Considers content of e-mails as trustworthy
  - Can retrieve files from an FTP site and reassemble them overwriting existing files of the same name
  - Viruses can avoid detection by using fragmentation
- RPC and RPCbind
  - Use any port number assigned to them
  - Difficult to detect using packet filters

Denial of Service

ATTACKS ON AVAILABILITY

Denial of Service Attacks

- Attack against availability of resources
  - Bandwidth, information, computing resources, software at the client/server side etc.
  - An estimate is that there are anywhere between 20 and 40 attacks per hour
- Types
  - Network denial of service
    - Typically involves flooding a target with packets at the link or network layers
  - Service denial
    - Provide false information, interrupt information or crash server
  - DoS on the client side
    - Crash the client software
- It is **impossible** to prevent denial of service
  - Lawsuits may be easy but tracing the perpetrator is hard

Network Link Flooding Attacks

- Idea:
  - Create more traffic on a link than the capacity of the link
  - Example: T1 line, 200 KBps is sufficient
  - Requirement:
    - Connecting link must have more capacity than the target link
- Examples
  - Directed broadcast (ICMP) – **Smurf attack**
  - SMTP flooding and SPAM
    - Send 10 MB attachments from several clients to the same mail server rapidly
  - Small services attacks
    - Commands like *echo* on UNIX are used for maintenance but can be used for Smurf style attacks
    - Locate two echo servers and send a packet from one to the other creating an infinite loop

Some Attacks on Service

- TCP SYN Flood attack
  - Create several half open connections so that legitimate requests are dropped
- BGP attacks
  - False routing messages ensure that packets never reach the destination
- NIS bogus backup server attack
  - NIS services have a backup server
  - If the address of the backup server is spoofed, NIS services may be affected
- Deregister services with RPCbind
  - A fabricated deregister message to rpcbind can prevent actual rpc services from being used

Client Attacks

- Specially crafted URLs can
  - Crash a browser
  - Hang the mouse
  - Create false alarms
- Java applets
  - Open so many windows that the client has no resources to do anything else

Distributed DoS

- Received immense attention since the attack of February 2000 on popular web sites
- How it typically works
  - Oscar uses bugs and backdoors to install *zombie* or *agent* programs on many machines in many domains
  - Oscar installs a *master* program on some other machines with a list of the zombies
  - Oscar waits
  - At the time of strike, he sends a message to the master indicating the address of the target
    - Messages may be encrypted
  - The master sends messages to all zombies to attack the target
- Zombies launch attack – flood target link or server
  - This could be a Smurf attack, TCP SYN flood, UDP flood etc.

Botnets

- The *zombies* in a Botnet are not all used for the DDoS attack
  - Oscar typically creates a huge network of such zombies or bots
- What are botnets used for?
  - Obviously DDoS
  - Spamming
  - Distributed vulnerability scanning
  - Cryptanalysis ☹
- How are botnets created?
  - Typically using bugs and backdoors

Classification of DDoS Attacks

- Degree of automation
- Exploited weakness to deny service
- Source address validity
- Attack rate dynamics
- Possibility of characterization
- Persistence of agent set
- Victim type
- Impact on victim

Source: J. Mirkovic and P. Reiher, "A Taxonomy of DDoS Attack and DDoS Defense Mechanisms," ACM Computer Communications Review, Vol. 34, No. 2, April 2004

Degree of Automation - I

- Manual
  - Oscar has to manually scan machines for vulnerabilities, break into them and then initiate the onset of attacks
- Semiautomatic
  - There is a “Master” and “Agent” (zombie)
    - Recruiting, exploiting and infecting are automated
  - Oscar still instructs the agents - attack type, victim, duration, time through the master
    - Instruction can be direct or indirect
      - Direct => master must know who the agents are and vice versa => IP address of master is hard coded in the agents
        - Detection is easier
      - Indirect => some IRC channel or some other subliminal channel is used for communication
        - Detection is harder

Degree of Automation - II

- Automatic
  - The use phase is also automated
  - Avoids need for communication between Master and Agents
- Attack code is pre-programmed
  - Start time of attack, victim , attack type, duration, etc.
  - Attacker is minimally exposed
- Disadvantage - it is inflexible :-)

Scanning Strategies

- Automatic and semi-automatic DDoS schemes use worms or trojans
  - 3 million scans reported per day based on analyses of firewall logs from 1600 networks
- Scanning has two phases - host and vulnerability scanning
  - Hosts selected can be random, may have a hit list, and so on
  - Can scan for one particular port on all machines or use a list
  - May be coordinated and/or stealthy

Propagation

- Central source
  - Once a vulnerability is detected, the code to infect a machine is downloaded from a central server
  - Single point of failure
  - Easy to detect and disable
  - Example: 1i0n worm
- Back-chaining
  - The host that scans also provides the infecting code
  - Once infected, the new victim starts the same process
  - Ramen and Morris worms used this
- Autonomous
  - Scanning and downloading occur in one step - Code-Red, E-mail based worms

Exploited Weakness

- Semantic
  - Exploit the “semantics” or features of a protocol
  - Example - TCP SYN Flood Attack uses the fact that servers allocate substantial resources to TCP connections upon receiving a SYN segment
- Brute Force
  - Initiate a vast amount of transactions that appear to be legitimate - example a large number of huge e-mails
  - Usually needs much higher volume of traffic to be generated

Source Address Validity

- Valid source address
  - Some machines cannot be used to spoof IP addresses since such functions are NOT available on the OS
  - Example Win95 and Win98
- Spoofed address
  - Most beneficial to Oscar
  - Many types and reasons for spoofing addresses
- Routable spoofed address
  - To reflect responses and deny service to the host whose address has been hijacked
- Non-routable spoofed address
  - Use reserved addresses like 192.168.0.0/16
    - Can be discarded easily
  - Use valid addresses but those that are unused in a network
    - Can provide useful information through traceback and backscatter

Spoofing Technique

- Random
  - Generate random 32 bit numbers and use them
- Subnet spoofed address
  - Use an address that belongs to the subnet space of the agent machine
  - Example: If the agent belongs to 136.142.117.0/24, it can spoof any address in the range 136.142.117.1 - 136.142.117.254
- En Route Spoofed Address
  - Uses the address of a machine that is on a subnet along the route to the destination
  - No known instances
- Fixed spoof address
  - Tries to fix blame on a particular list of hosts

Attack Rate Dynamics

- Attack Rate = Flow rate of stream from agent to victim
- Constant rate
  - Typical and starts suddenly overwhelming the victim
- Variable rate
  - Gradually increasing rate to avoid quick detection
  - Fluctuating rate - pulsing, periodic and so on

Possibility of Characterization

- Characterizable
  - Use TCP, IP and other protocol information to identify type of attack
  - Example: TCP SYN Flood Attack
  - Types:
    - Filterable - example if there are UDP or ICMP floods, a web server that minimally uses them can block them
    - Non-filterable - HTTP floods to web server or DNS request flood to name server
- Non-Characterizable
  - Attacks that use a mix of TCP SYN, ICMP, TCP ACK, UDP Flood packets
  - Needs careful examination and characterization is often subject to interpretation

Persistence of Agent Set

- Does the set of agents remain constant or change with time?
  - If the agent set changes with time, it will be more difficult to trace the perpetrators
- Constant agent set
  - They all act in the same manner
  - They receive the same set of commands
- Variable agent set
  - Oscar divides the agent set into subsets
  - One or more subsets are used at a time
  - An agent could belong to more than one subset

Victim Type

- Application
  - Objective of attack is to disable a service
  - Example: Bogus signature attack on an authentication server
  - Hard to detect application oriented attacks
- Host
  - Objective is to disable a host - crash, reboot or freeze it
    - Example: TCP SYN Flood attack
  - All attack packets carry the destination address of that host
  - Easy to detect
- Resource
  - Objective is to disable a critical resource such as a name server, router or link
- Network
  - Consume bandwidth of the network
  - Destination addresses can be any host in the subnet
- Infrastructure
  - Objective is to disable the operation of the global Internet or parts of it
  - Examples - attacks against root nameservers, core routers, certificate servers, and so on

Impact on Victim

- Disruption
  - Deny availability completely
- Dynamic recovery
  - May be possible to recover during or after attack
    - Example: Network bandwidth
  - Self-recovery, human intervention for recovery and non-recoverable cases
- Degradation
  - Deny availability partially but at immense economic cost

Challenges in Defending Against DDoS Attacks

- Lack of detailed attack information
  - Analyses of specific attacks exist
  - Data related to frequency, distribution, number of agents, effectiveness of response is not available
- Lack of benchmarks
- Difficulty of large-scale testing
  - How good is a particular defense on a large scale?
- Need for distributed response
  - Needs coordination across administrative domains
- Economic and social factors

Recovering from DDoS

- Mitigation is possible but not absolute solutions
- Techniques for mitigation
  - Ingress filtering at the edge
  - Prevention, detection and response
    - Filter bad packets (question is how?)
      - Need to get into nitty gritty details to filter packets
    - Improve data processing speeds
    - Add hardware and link capacity to handle normal load + attack
    - Hunt and shut down attacking sites
- More when we look at monitoring and intrusion detection

IP Traceback

- Where are the DoS related packets coming from?
- Types of traceback
  - Actively query routers
    - Victim develops a signature of the attack and proceeds hop by hop to see where the packets actually originated
  - Create a virtual overlay network for selective monitoring of flows and logging
  - Identify the path by reconstruction using probabilistic “packet marking”

Backscatter

- See: http://www.caida.org/outreach/papers/2001/BackScatter/
- Useful for identifying how prevalent DoS attacks are
- Idea
  - Capture packets that are sent in response to spoofed DoS packets **at unused IP addresses**
  - If Oscar chooses IP addresses at random and packets are captured in a sufficiently large address space, they provide a good "sample" for analysis
- Example work
  - AT & T researchers did this (see pages 116-117 of FIS)
  - Supercomputer center in San Diego also did this analysis

DNS Attacks

EXAMPLES OF SPECIFIC DOS ATTACK ON DNS PROTOCOL

Some Attacks on DNS

- Bugs and backdoors
  - Vulnerabilities in BIND
- Information Leakage
  - Zone transfer attacks
- Denial of Service
  - DNS Tainting or Cache Poisoning
- Birthday Attack

Vulnerabilities in BIND

- BIND is used by most DNS servers
  - One sample showed 45% of responses indicated some version of BIND[1]
- Different versions of BIND have different vulnerabilities
  - Buffer overflow, susceptibility to DoS, information leaks
  - In May 2004, even many root servers used vulnerable versions of BIND
- See for example:
  - CERT Advisory: http://www.cert.org/advisories/CA-2002-19.html

1 Source: M. Schiffman, "Bound By Tradition: A Sampling of the Security Posture of the Internets DNS Servers," Available at: http://www.packetfactory.net/

Impact

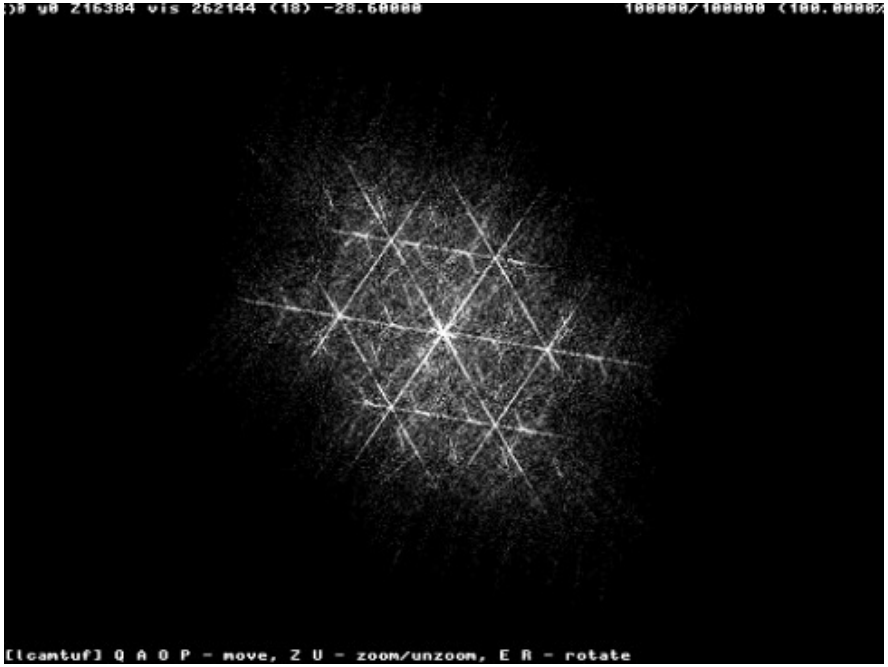
- DNS often operates with root privileges
  - Information about bugs revealed by dig can be used by Oscar to exploit the host or launch DoS attacks
- Many other protocols use BIND to resolve names and addresses
  - Example: Sendmail could be used to exploit the bug in BIND
- Major attack (2001)
  - erkms and 1ion worms deployed DDoS attacks using BIND vulnerabilities
- Some vulnerabilities may be minor, but still problematic
  - BIND v9 (prior to 9.2.1) could be shut down if a specially crafted packet was sent to it[1]

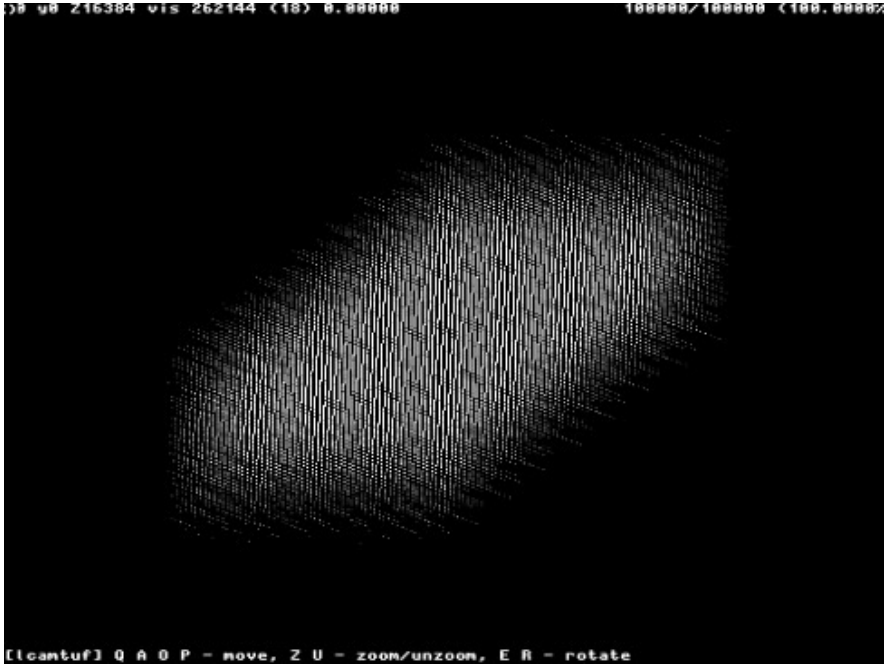
[1] http://www.cert.org/advisories/CA-2002-15.html

Tainting DNS Responses

- If a malicious host can assume the identity of a legitimate host (what does identity mean?), it can get access to many resources
- Versions of BIND before 8.1.1 had vulnerabilities
- Idea: Cache malicious or misleading data from a remote name server in the target/victims name server
- Steps:
  - Fool the victims name server to query Oscar's name server
  - Respond with a corrupted RR that victims name server caches
- DNS packets contain both a query and a response
  - Some versions of BIND cache responses in a query without checking if it is a valid response to a query
- Older versions used non-random sequence numbers making it easy for Oscar to spoof DNS responses and queries

DNS Sequence Number Prediction





Source: DNS Cache Poisoning, The Next Generation - By Joe Stewart, at http://www.securityfocus.com/guest/17905

Real Examples

- Oscar sends a DNS query for the IP address of www.hillary2000.org but attaches a response (a.b.c.d) as well
  - Query is sent to many DNS servers
  - Buggy BIND in some DNS servers caches the response (a.b.c.d) which is for the site www.hillaryno.com
- Users attempt to reach hillary2000.org but are sent to hillaryno.com
  - Source: Northcutt & Novak, Network Intrusion Detection, An Analyst's Handbook
- Another example:
  - Use SPAM or pop-ups that a user will click on by accident or intentionally
  - The DNS server presents a query to Oscar's DNS server
  - Oscar's server poisons the DNS cache of victim's DNS server
  - Actual attack on Microsoft's DNS Servers in 2001 (CERT IN-2001-11)

http://www.cert.org/incident_notes/IN-2001-11.html

Birthday Attack on DNS

- New versions of BIND use random sequence numbers (65535 possible values) in the queries
- How can Oscar spoof the random number in a response to taint or poison the DNS cache?
- Two methods
  - Flaw in versions of BIND: DNS Server sends multiple queries for the same name if it receives requests from multiple hosts for the same name - Birthday Attack
  - Guess the random sequence number
- First attack involves Oscar sending many spoofed queries to the victim DNS Server (how many for a 50% chance of getting one sequence number right?)
  - Send the spoofed responses to the victim DNS server

DoS Attacks on Root Servers

- October 21, 2002: DDoS attack on root servers
- Zombies in a Botnet launched a ping flood on the 13 root servers
  - ping is legitimately used to check if a root server is alive
  - Seven were crippled completely
  - Four maintained functionality
- Most people did not observe any disruptions
  - DNS caching kept most DNS queries working properly
- The DDoS stopped after 1 hour abruptly unlike other DDoS attacks that fade away gradually or need to be stopped
- Mitigation
  - Use anycast addresses and distribute the root server
  - Download root-server zone map every day to local name server

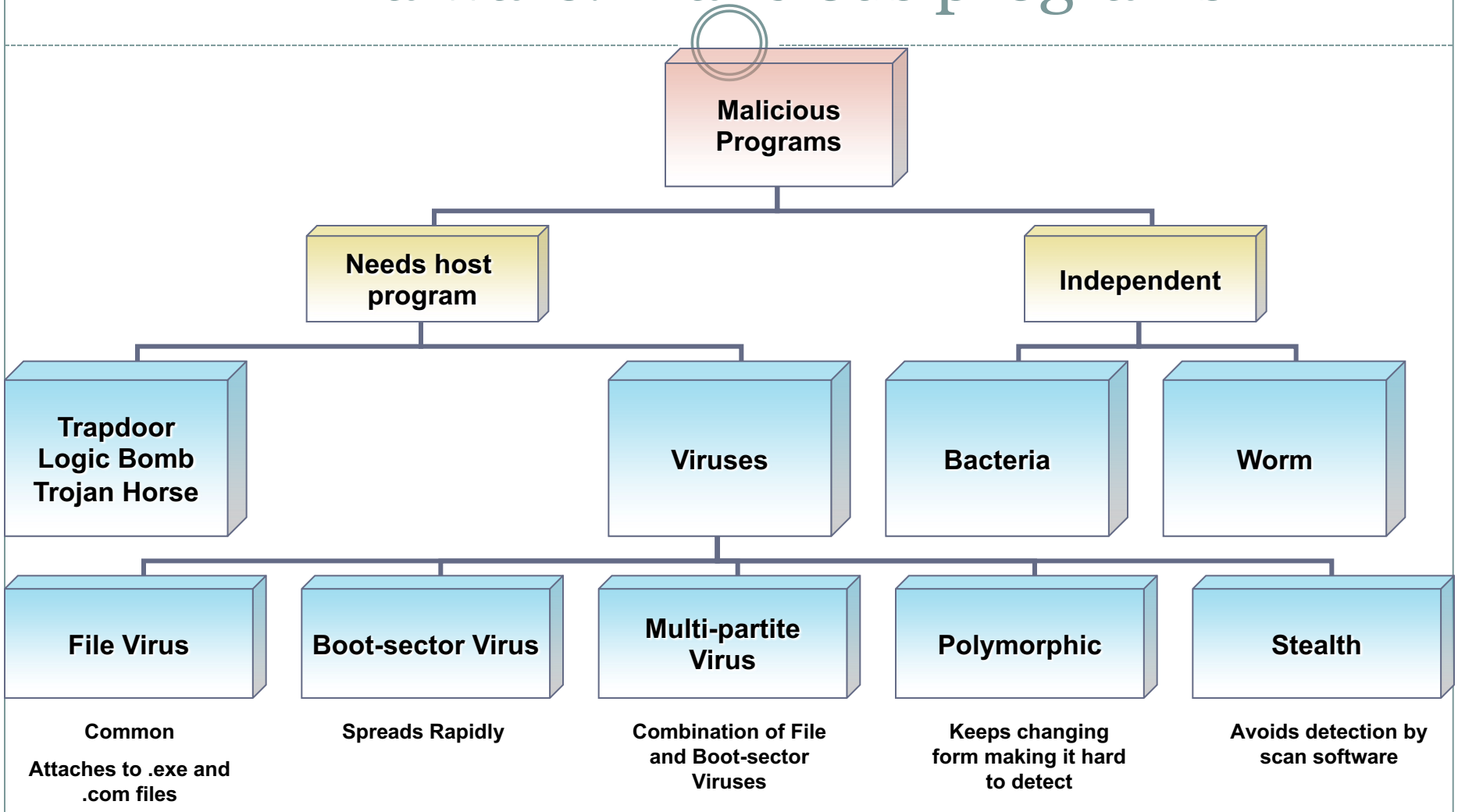
Email Attacks

- Hoax
  - Email asking to do things that is non-sense
  - Examples are Email chain, Lottery Winning , etc.
- SPAM
  - Email that annoys receivers
  - Mostly sent in mass to recipients creating a traffic bottleneck
- Phishing
  - Email that lures recipients to do things mostly related to financial actions

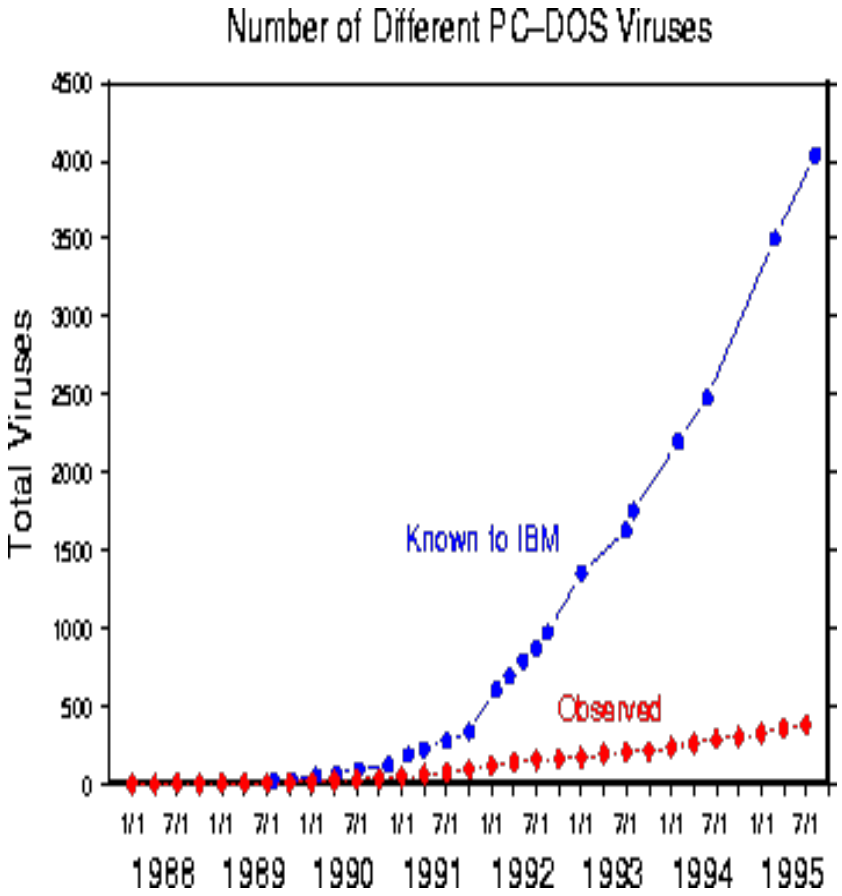
Exponential Attacks

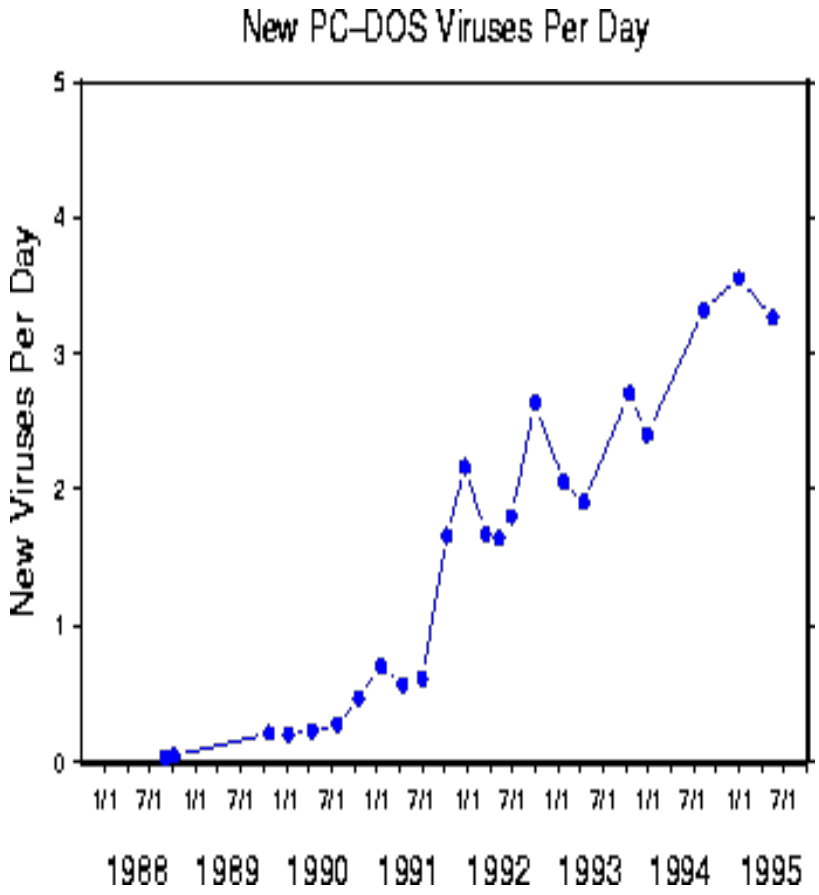
- Worms and viruses can spread rapidly
  - Exploit bugs in software and protocols
  - Around the world within minutes
  - Can cause severe economic harm
- Most solutions are reactive
  - Virus-scanning software
  - Shut off unnecessary services by default
  - Avoid monoculture

Malware: Malicious programs



Prevalence of computer viruses[1]





[1] *Source: IBM research: http://www.research.ibm.com/antivirus/index.htm*

Example: Code-Red

- History
  - Morris Worm was the first known worm in 1988
  - In June 2001, Microsoft IIS web servers were identified with having a buffer overflow bug
  - A patch was issued quickly, but not applied universally
  - Code-Red (I) v1 exploited this in July 2001
  - Code-Red (I) v2 started a week later
  - Code-Red (II) was let loose in August 2001
- Which TCP port did these worms probe?

Version differences

- Code-Red (I) v1 was memory resident
  - Rebooting would get rid of it
  - Each infected machine scanned IP addresses in the same order
- Code-Red (I) v2 was also memory resident
  - It scanned IP addresses using a random seed
  - More machines got infected than ever before
- Code-Red (II) was unrelated but contained the string Code-RedII
  - It was not memory resident
  - Became dormant for a day
  - Then started spreading

Summary of Code-Red Attach

- Intelligent
  - Did not probe loopback or multicast addresses
- Exponential
  - Number of infected hosts grew exponentially
  - Lower bound - 359000 unique IP addresses for Code-Red (I)
- Countries affected
  - US (43%), Korea (11%)
- Domains affected
  - A variety of domains - .net, .edu, .com

Other undesirable programs

- Spyware
  - Software that advertises, collects personal information, or changes the configuration of a computer
  - Typically does this without obtaining the user's consent
- Researchware?
- Ransomware!

Morals of the Story

WHAT DO THESE ATTACKS TELL US?

Assess your network

- Have proper policies in place
- Make sure systems are running only the allowed services
- Make sure that vulnerabilities are few in software tools that are being used
- Avoid monoculture?

Filter Packets

Protection

- If a malicious packet does not reach a host, it cannot cause harm
  - Host firewalls
    - Reject directed broadcast ICMP packets
    - If an IP packet arrives from outside with the source address that is from inside your network, reject it
    - Restrict access to machines on your network to the extent possible
    - Reject unknown protocols
- Use the maxim:
  - Do not give a person or a program access or privileges that are not necessary

Encryption

Protection

- Using cryptographic protocols is better than not using them
  - Performance considerations
  - Encryption is only as secure as the host that it is originating from
  - Encryption is only as secure as the password that is used to generate, store, or access keys
  - Be careful of protocol failures

Monitor and Log Communications

- Track all packets entering and leaving the network
  - You may detect anomalies
  - You may be able to trace Oscar
  - You may be able to save yourself from a bigger attack
- Employ an intrusion detection system and auditing process
- (Some may consider it an invasion of privacy)

Detection