Computer Security DD2395

http://www.csc.kth.se/utbildning/kth/kurser/DD2395/dasak10/

Spring 2010

Sonja Buchegger
buc@kth.se

Lecture 8, Feb. 10, 2010

Malicious Software, Denial of Service

Announcements

- Lab reports
- Bonus points
- Presentation topics, tasks
- Guest lecture on audits, Feb. 17

Virus Countermeasures

- prevention - ideal solution but difficult
- realistically need:
  - detection
  - identification
  - removal
- if detect but can't identify or remove, must discard and replace infected program

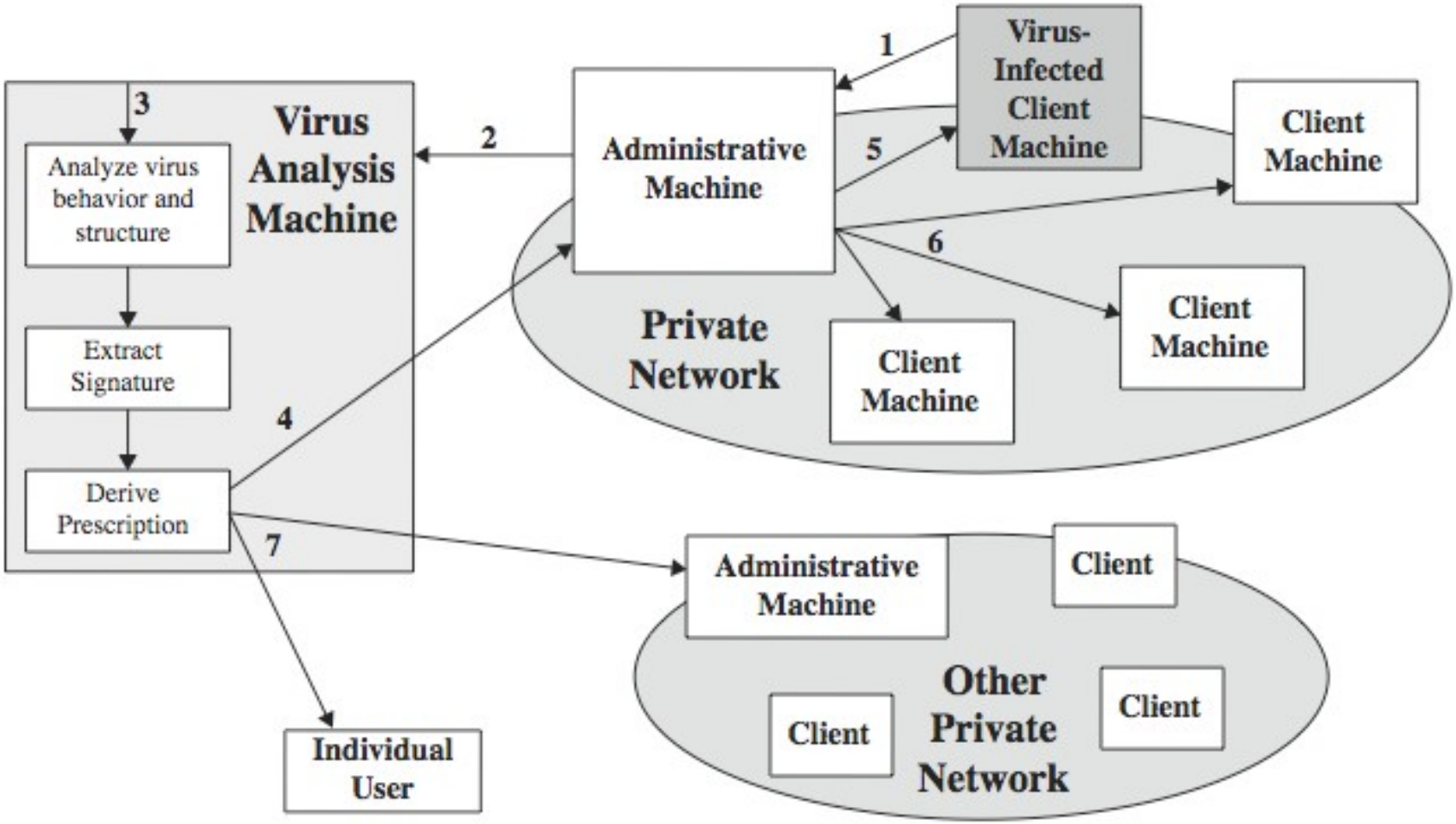
Anti-Virus Evolution

- virus & antivirus tech have both evolved
- early viruses simple code, easily removed
- as become more complex, so must the countermeasures
- generations
  - first - signature scanners
  - second - heuristics
  - third - identify actions
  - fourth - combination packages

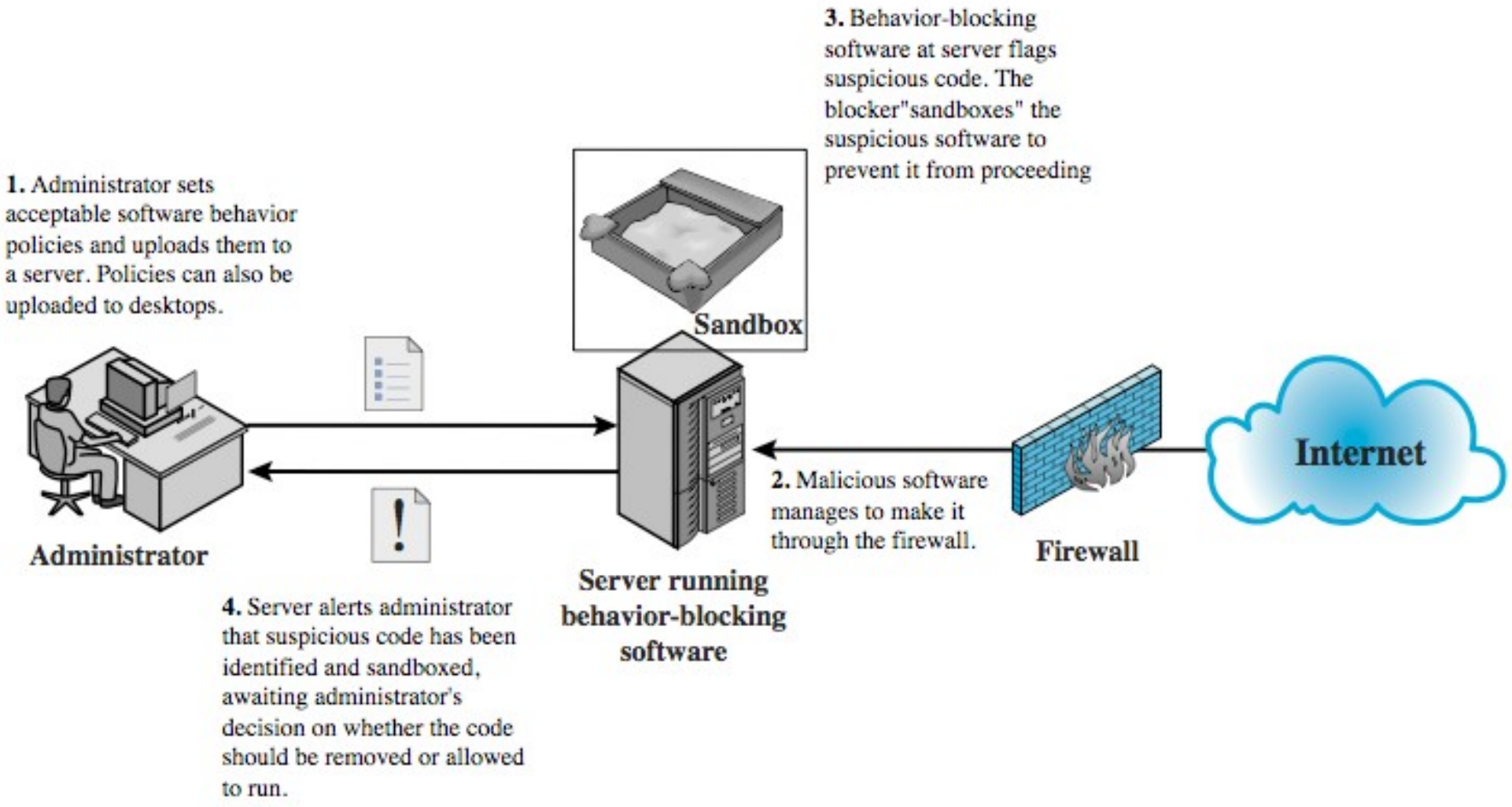
Generic Decryption

- runs executable files through GD scanner:
  - CPU emulator to interpret instructions
  - virus scanner to check known virus signatures
  - emulation control module to manage process
- lets virus decrypt itself in interpreter
- periodically scan for virus signatures
- issue is long to interpret and scan
  - tradeoff chance of detection vs time delay

Digital Immune System



Behavior-Blocking Software



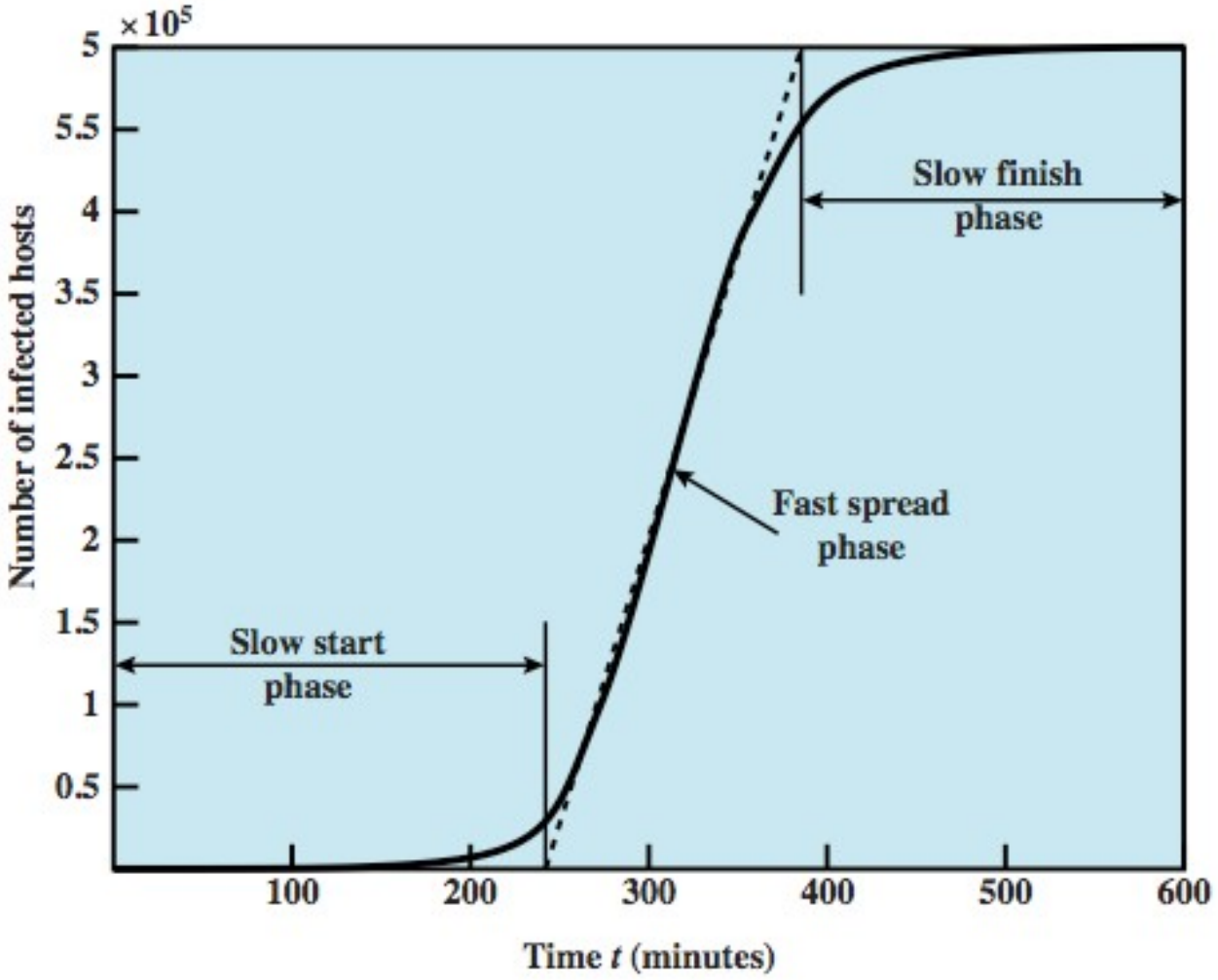
Worms

- replicating program that propagates over net
  - using email, remote exec, remote login
- has phases like a virus:
  - dormant, propagation, triggering, execution
  - propagation phase: searches for other systems, connects to it, copies self to it and runs
- may disguise itself as a system process
- concept seen in Brunner's "Shockwave Rider"
- implemented by Xerox Palo Alto labs in 1980's

Morris Worm

- one of best know worms
- released by Robert Morris in 1988
- various attacks on UNIX systems
  - cracking password file to use login/password to logon to other systems
  - exploiting a bug in the finger protocol
  - exploiting a bug in sendmail
- if succeed have remote shell access
  - sent bootstrap program to copy worm over

Worm Propagation Model



Recent Worm Attacks

- Code Red
  - July 2001 exploiting MS IIS bug
  - probes random IP address, does DDoS attack
  - consumes significant net capacity when active
- Code Red II variant includes backdoor
- SQL Slammer
  - early 2003, attacks MS SQL Server
  - compact and very rapid spread
- Mydoom
  - mass-mailing e-mail worm that appeared in 2004
  - installed remote access backdoor in infected systems

Worm Technology

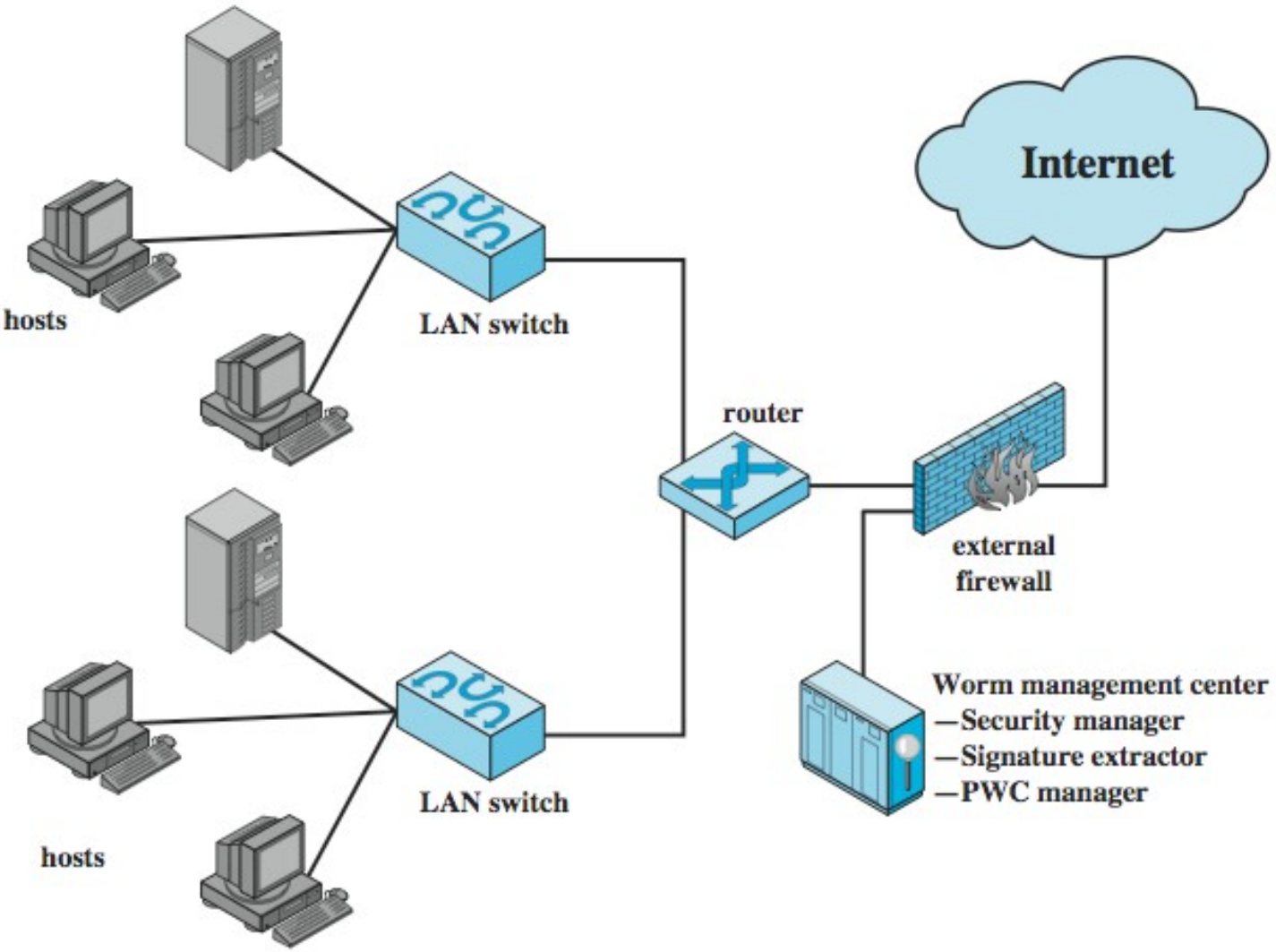
- multiplatform
- multi-exploit
- ultrafast spreading
- polymorphic
- metamorphic
- transport vehicles
- zero-day exploit

Countermeasures

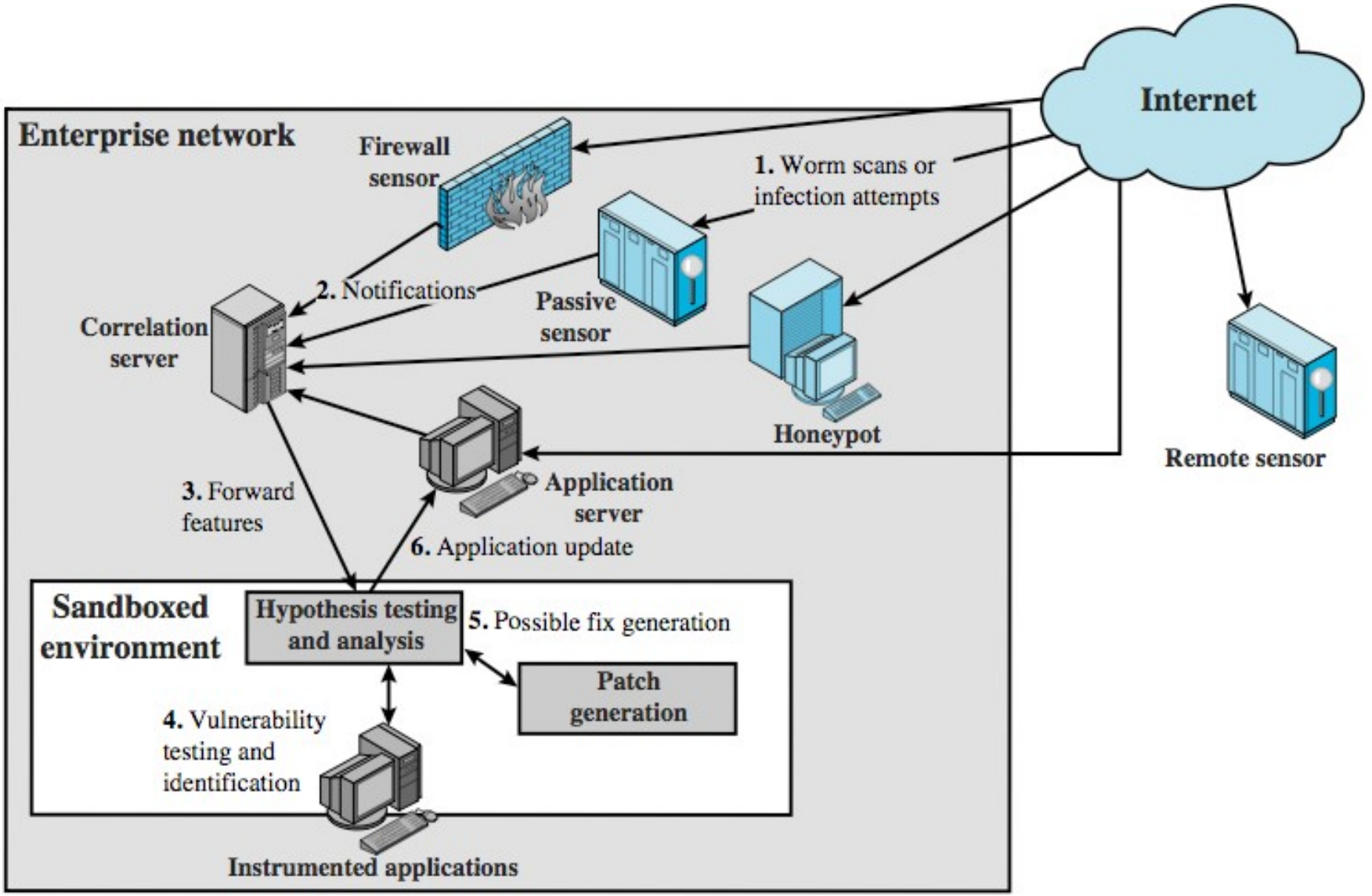
Worm Countermeasures

- overlaps with anti-virus techniques
- once worm on system A/V can detect
- worms also cause significant net activity
- worm defense approaches include:
  - signature-based worm scan filtering
  - filter-based worm containment
  - payload-classification-based worm containment
  - threshold random walk scan detection
  - rate limiting and rate halting

Proactive Worm Containment



Network Based Worm Defense



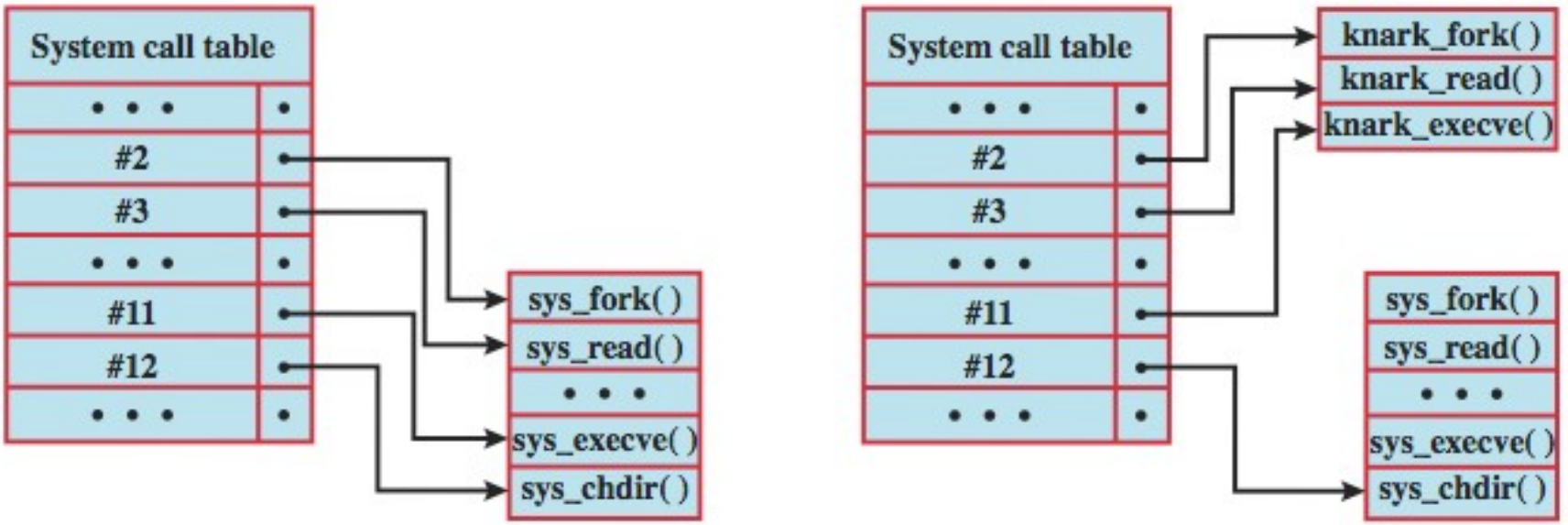
Bots

- program taking over other computers
- to launch hard to trace attacks
- if coordinated form a botnet
- characteristics:
  - remote control facility
    - via IRC/HTTP etc
  - spreading mechanism
    - attack software, vulnerability, scanning strategy
- various counter-measures applicable

Rootkits

- set of programs installed for admin access
- malicious and stealthy changes to host O/S
- may hide its existence
  - subverting report mechanisms on processes, files, registry entries etc
- may be:
  - persistent or memory-based
  - user or kernel mode
- installed by user via trojan or intruder on system
- range of countermeasures needed

Rootkit System Table Mods



(a) Normal kernel memory layout

(b) After nkark install

Summary

- introduced types of malicous software
  - incl backdoor, logic bomb, trojan horse, mobile
- virus types and countermeasures
- worm types and countermeasures
- bots
- rootkits

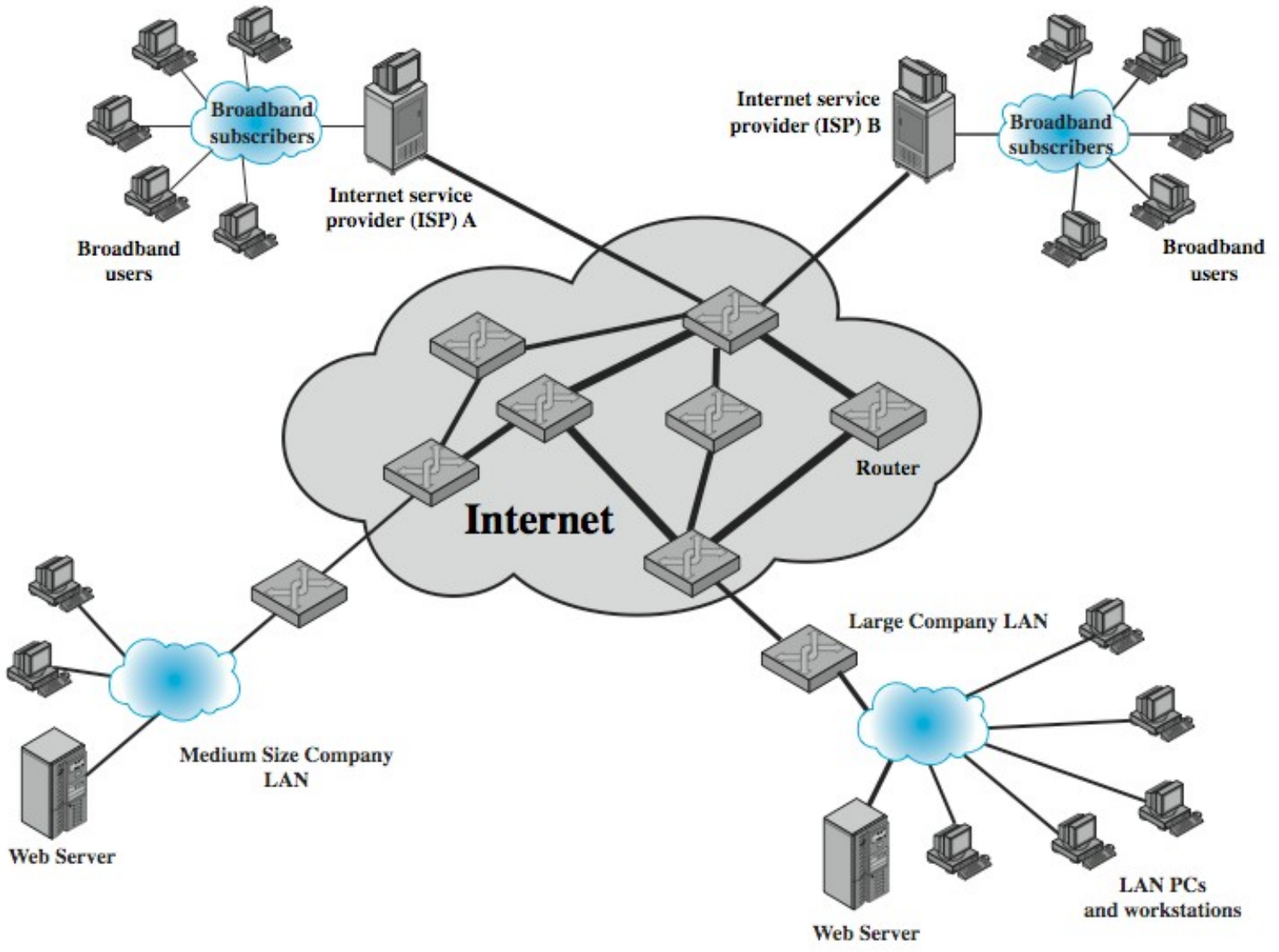
Denial of Service

- **denial of service** (DoS) an action that prevents or impairs the authorized use of networks, systems, or applications by exhausting resources such as central processing units (CPU), memory, bandwidth, and disk space
- attacks
  - network bandwidth
  - system resources
  - application resources
- have been an issue for some time

Classic Denial of Service Attacks

- can use simple flooding ping
- from higher capacity link to lower
- causing loss of traffic
- source of flood traffic easily identified

Classic Denial of Service Attacks



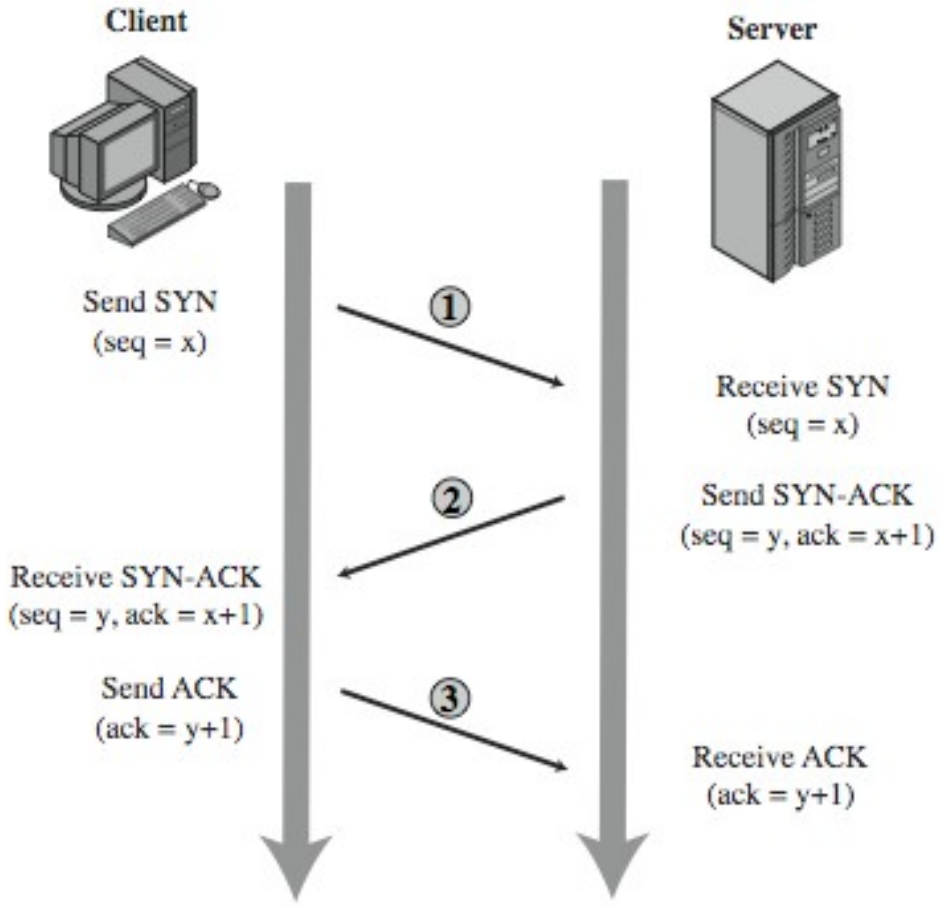
Source Address Spoofing

- use forged source addresses
  - given sufficient privilege to “raw sockets”
  - easy to create
- generate large volumes of packets
- directed at target
- with different, random, source addresses
- cause same congestion
- responses are scattered across Internet
- real source is much harder to identify

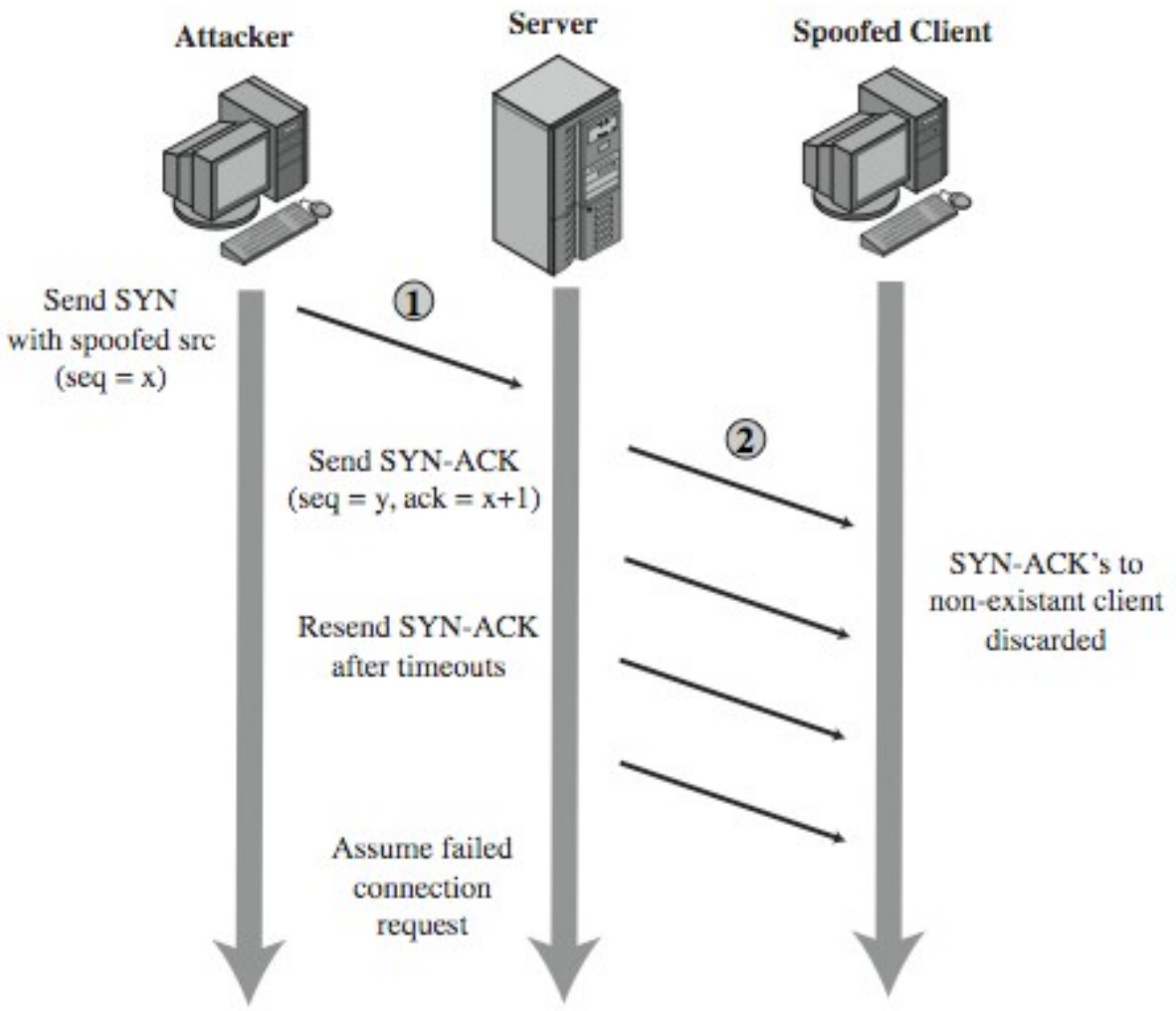
SYN Spoofing

- other common attack
- attacks ability of a server to respond to future connection requests
- overflowing tables used to manage them
- hence an attack on system resource

TCP Connection Handshake



SYN Spoofing Attack



Countermeasure

- One way of preventing such DoS: Make it costly to connect. Whoever wants to connect has to do solve a computation-heavy problem.
  - Is this effective?
  - When? Why?
  - When not? Why?

SYN Spoofing Attack

- attacker often uses either
  - random source addresses
  - or that of an overloaded server
  - to block return of (most) reset packets
- has much lower traffic volume
  - attacker can be on a much lower capacity link

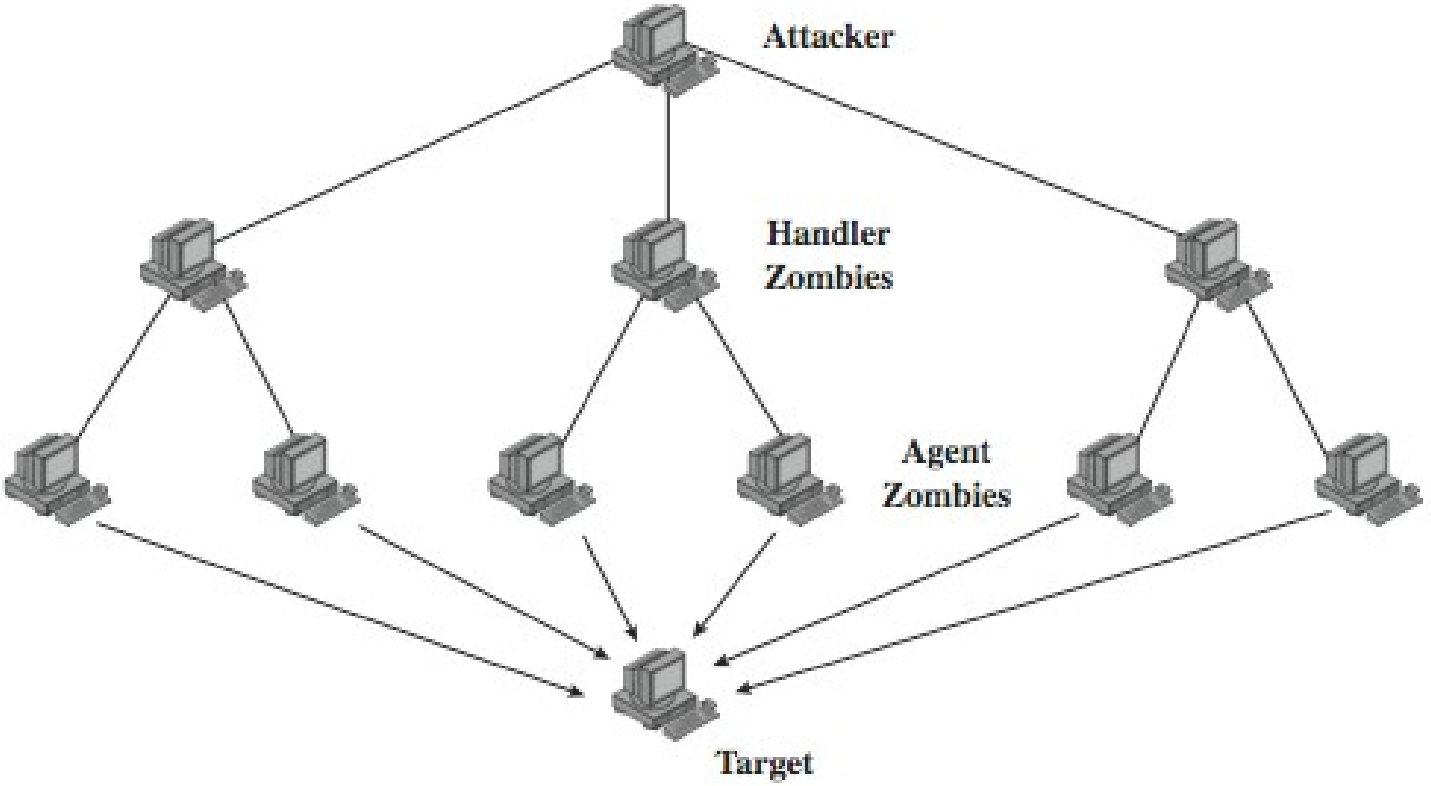
Types of Flooding Attacks

- classified based on network protocol used
- ICMP Flood
  - uses ICMP packets, eg echo request
  - typically allowed through, some required
- UDP Flood
  - alternative uses UDP packets to some port
- TCP SYN Flood
  - use TCP SYN (connection request) packets
  - but for volume attack

Distributed Denial of Service Attacks

- have limited volume if single source used
- multiple systems allow much higher traffic volumes to form a Distributed Denial of Service (DDoS) Attack
- often compromised PC's / workstations
  - zombies with backdoor programs installed
  - forming a botnet
- e.g. Tribe Flood Network (TFN), TFN2K

DDoS Control Hierarchy

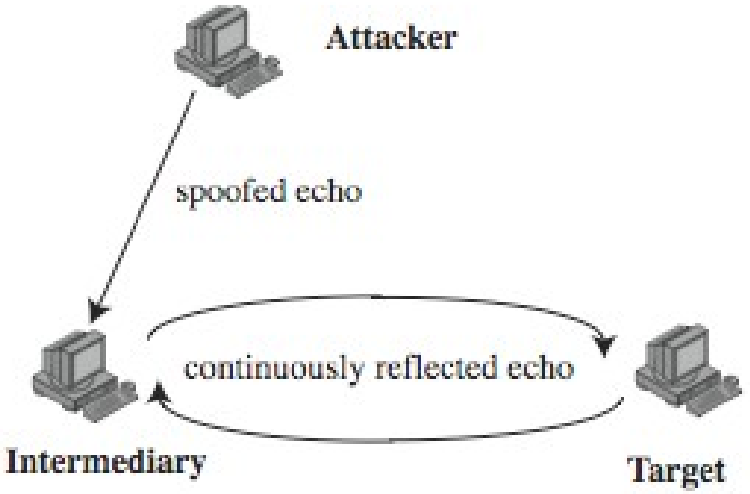


Reflection Attacks

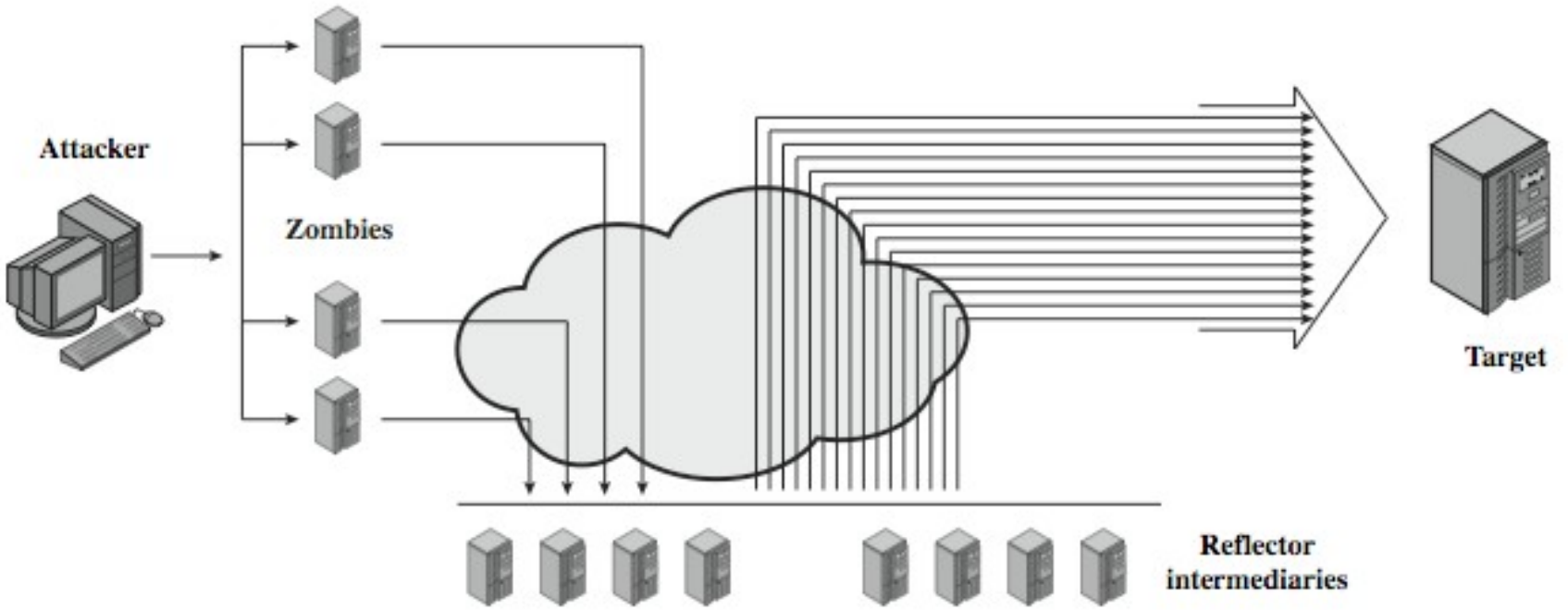
- use normal behavior of network
- attacker sends packet with spoofed source address being that of target to a server
- server response is directed at target
- if send many requests to multiple servers, response can flood target
- various protocols e.g. UDP or TCP/SYN
- ideally want response larger than request
- prevent if block source spoofed packets

Reflection Attacks

- further variation creates a self-contained loop between intermediary and target
- fairly easy to filter and block



Amplification Attacks



DNS Amplification Attacks

- use DNS requests with spoofed source address being the target
- exploit DNS behavior to convert a small request to a much larger response
  - 60 byte request to 512 - 4000 byte response
- attacker sends requests to multiple well connected servers, which flood target
  - need only moderate flow of request packets
  - DNS servers will also be loaded

DoS Attack Defenses

- high traffic volumes may be legitimate
  - result of high publicity, e.g. “slash-dotted”
  - or to a very popular site, e.g. Olympics etc
- or legitimate traffic created by an attacker
- three lines of defense against (D)DoS:
  - attack prevention and preemption
  - attack detection and filtering
  - attack source traceback and identification

Attack Prevention

- block spoofed source addresses
  - on routers as close to source as possible
  - still far too rarely implemented
- rate controls in upstream distribution nets
  - on specific packets types
  - e.g. some ICMP, some UDP, TCP/SYN
- use modified TCP connection handling
  - use SYN cookies when table full
  - or selective or random drop when table full

Attack Prevention

- block IP directed broadcasts
- block suspicious services & combinations
- manage application attacks with “puzzles” to distinguish legitimate human requests
- good general system security practices
- use mirrored and replicated servers when high-performance and reliability required

Responding to Attacks

- need good incident response plan
  - with contacts for ISP
  - needed to impose traffic filtering upstream
  - details of response process
- have standard filters
- ideally have network monitors and IDS
  - to detect and notify abnormal traffic patterns

Responding to Attacks

- identify type of attack
  - capture and analyze packets
  - design filters to block attack traffic upstream
  - or identify and correct system/application bug
- have ISP trace packet flow back to source
  - may be difficult and time consuming
  - necessary if legal action desired
- implement contingency plan
- update incident response plan

Summary

- introduced denial of service (DoS) attacks
- classic flooding and SYN spoofing attacks
- ICMP, UDP, TCP SYN floods
- distributed denial of service (DDoS) attacks
- reflection and amplification attacks
- defenses against DoS attacks
- responding to DoS attacks