Denial of Service

Lecture 9
Computer Security DD2395

Roberto Guanciale
robertog@kth.se

2015-11-26

Lab W

- Instructions on-line
- Booking for help sessions on-line
- All sessions can also be used to report your result
- Start tomorrow, go to the lab and start your assignment
- Session Dec 17 is only to report you work

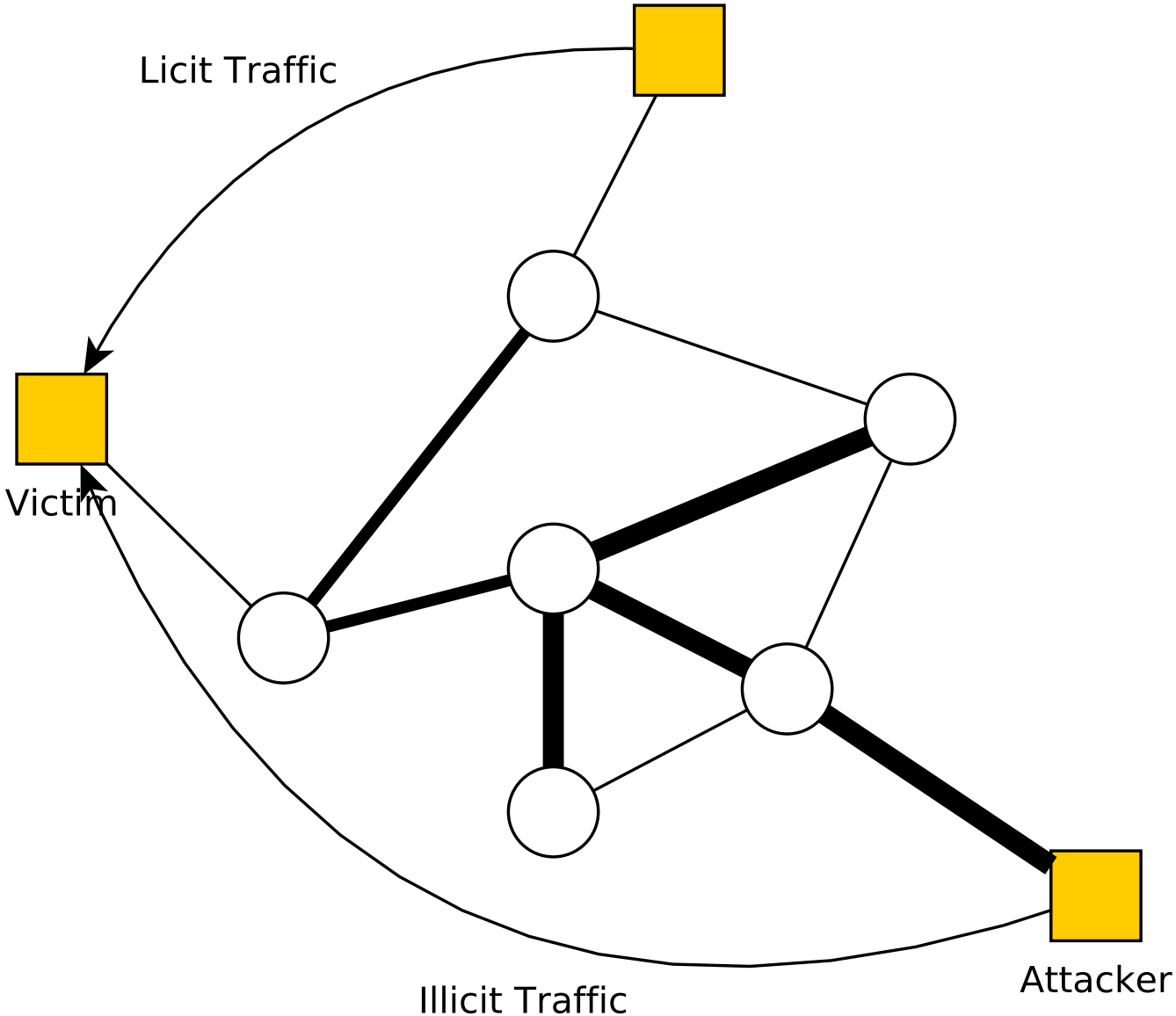
Denial of Service

- 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

Source Address Spoofing

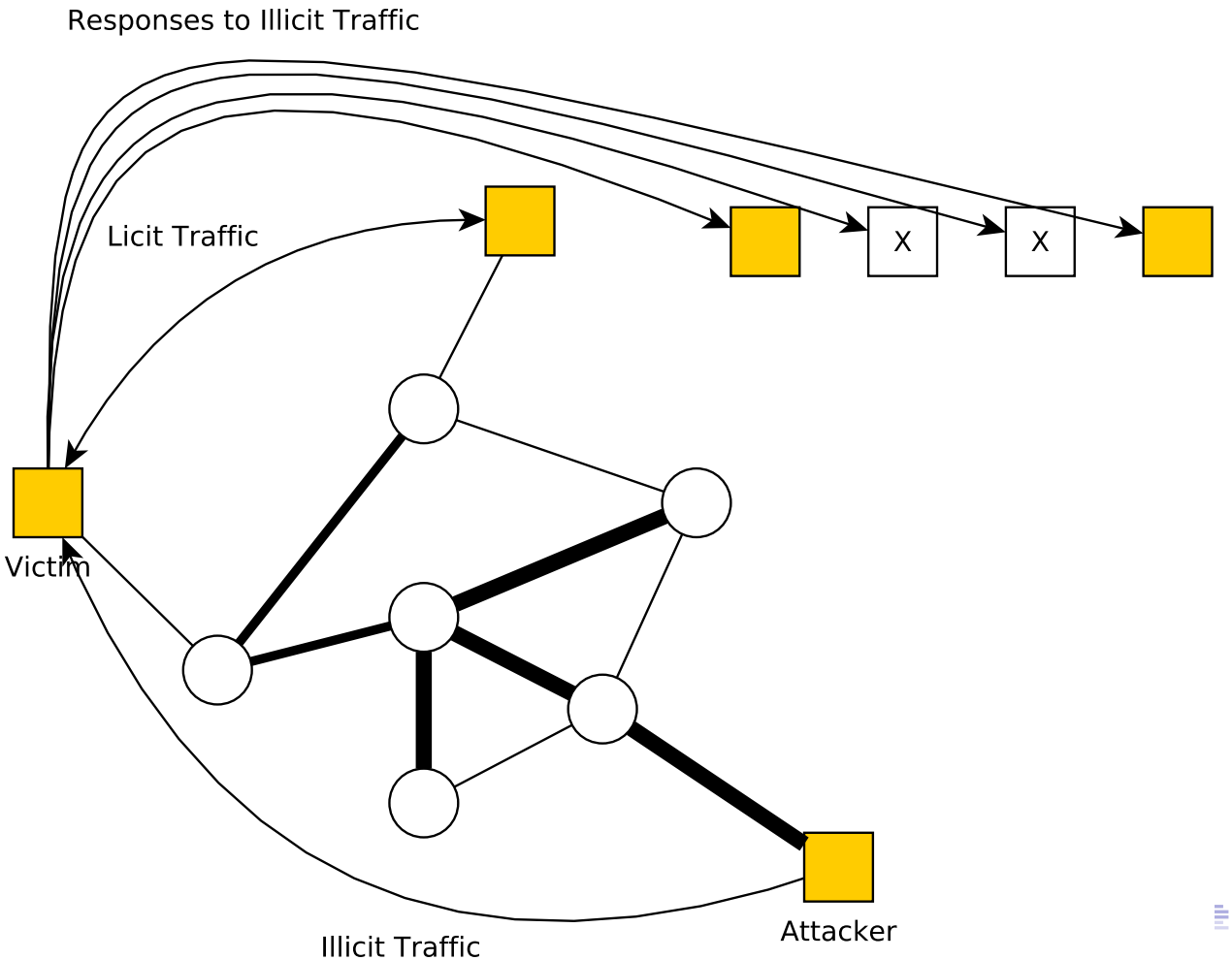
- Why would an attacker bother doing that?

Source Address Spoofing

- Why would an attacker bother doing that?
- To hide his identity

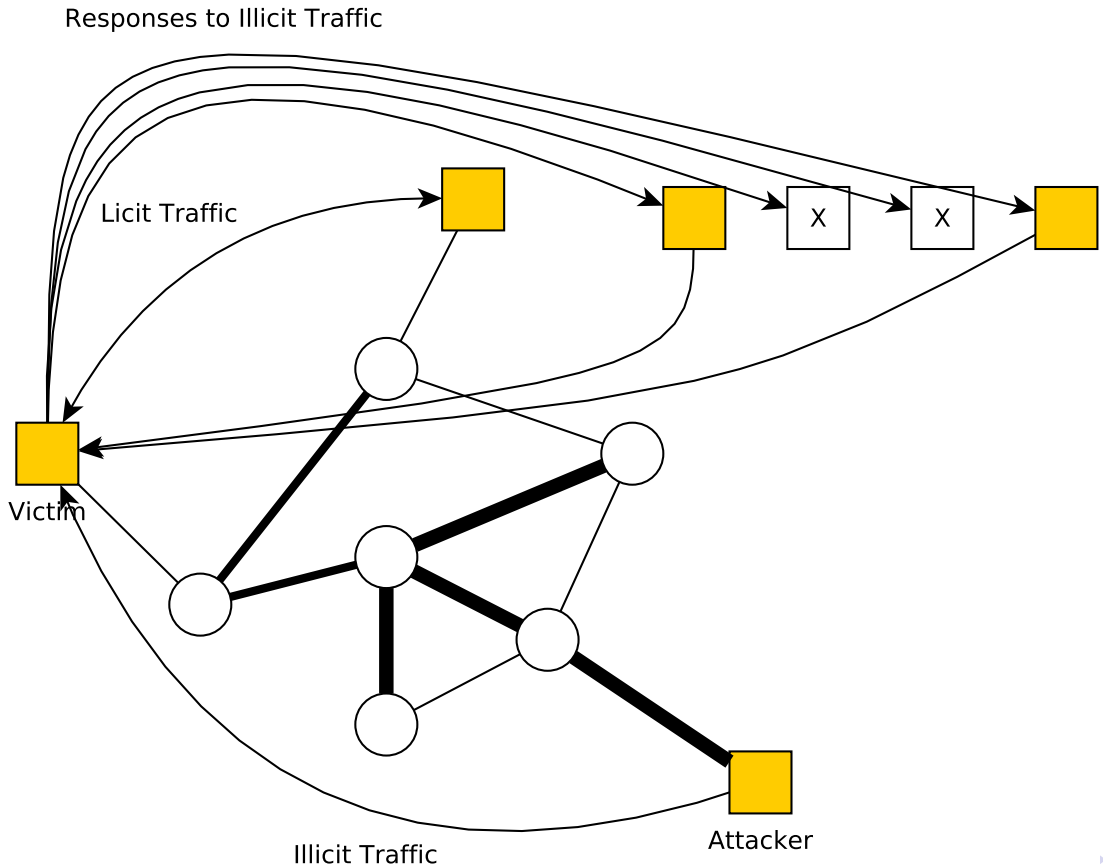
Source Address Spoofing

- Why would an attacker bother doing that?
- To hide his identity
- To avoid back traffic



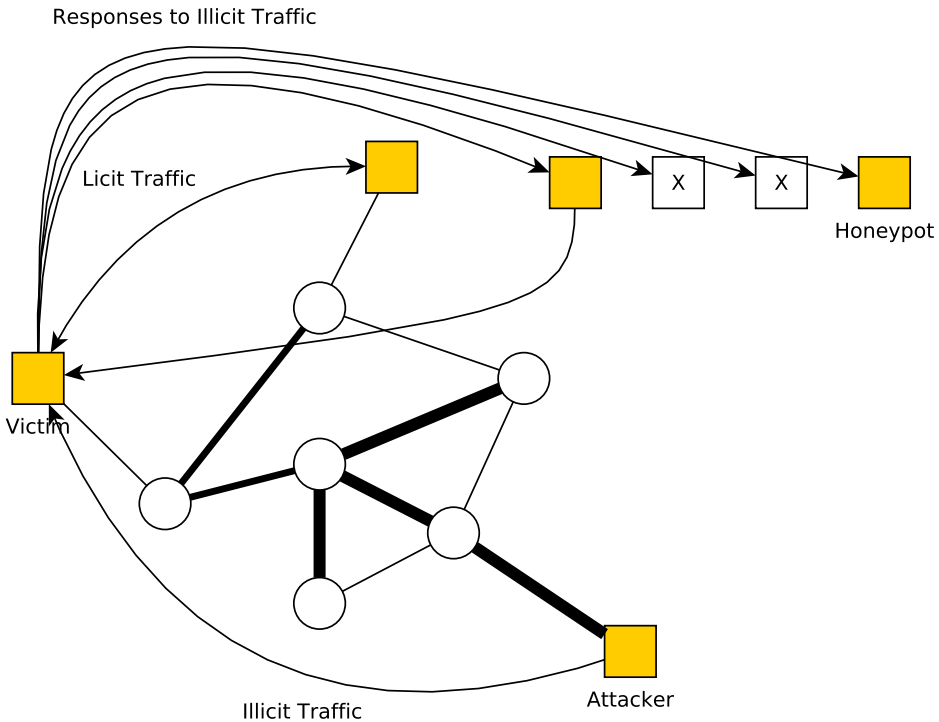
Source Address Spoofing

- Why would an attacker bother doing that?
- To hide his identity
- To avoid back traffic
- IMCP Answers increase the victim's congestion



Source Address Spoofing

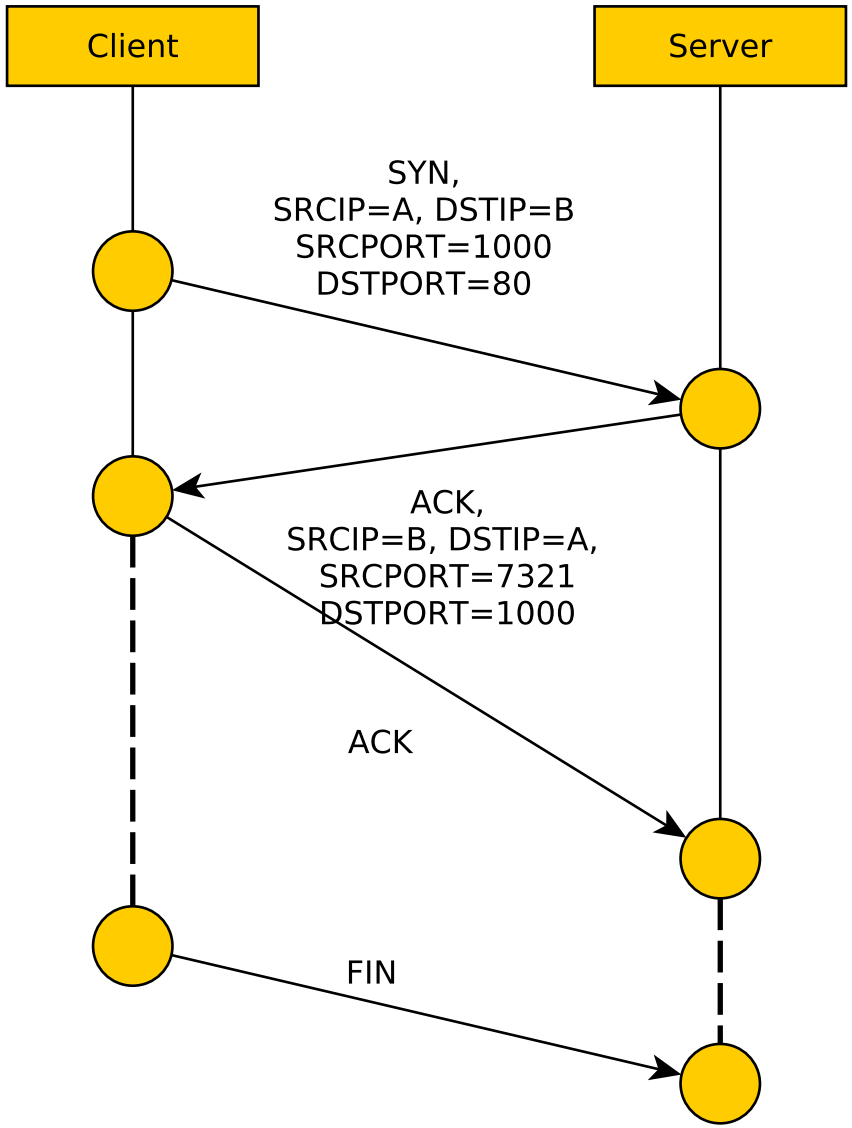
- Why would an attacker bother doing that?
- To hide his identity
- To avoid back traffic
- IMCP Answers increase the victim's congestion
- IMCP Answers can be used to identify attacks



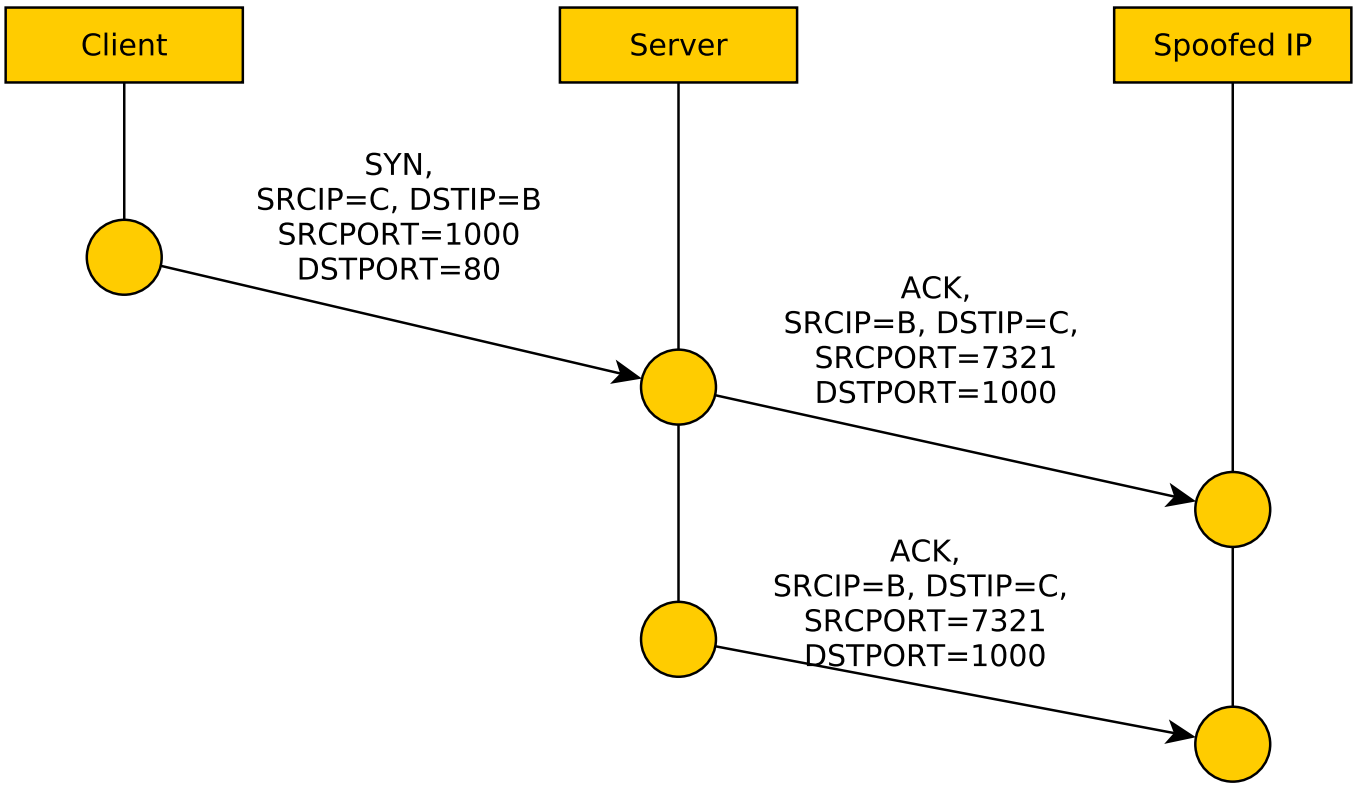
SYN Spoofing

- 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



- Use non-existent C
- Use C behind a dropping FW
- Overload C

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

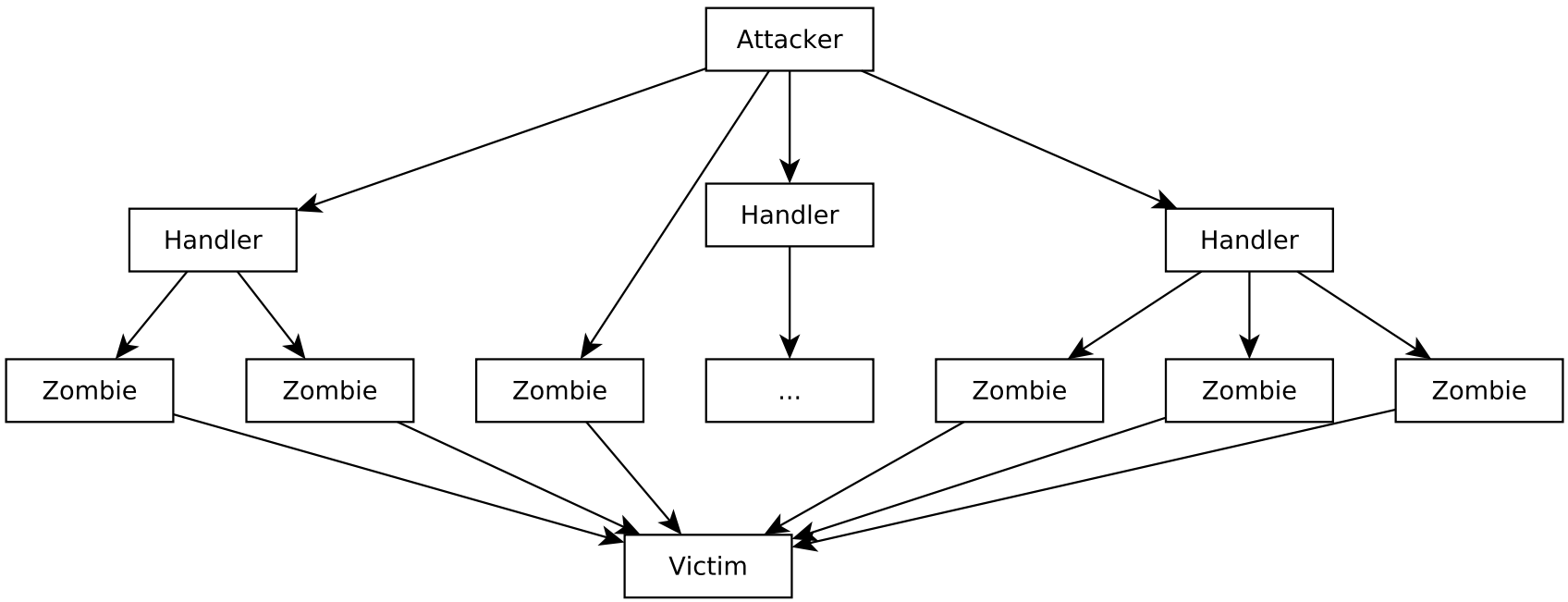
Types of Flooding Attacks

- 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 PCs / workstations
  - zombies with backdoor programs installed
  - forming a botnet
- e.g. Tribe Flood Network (TFN), TFN2K used a trojan

DDoS Control Hierarchy



Connections

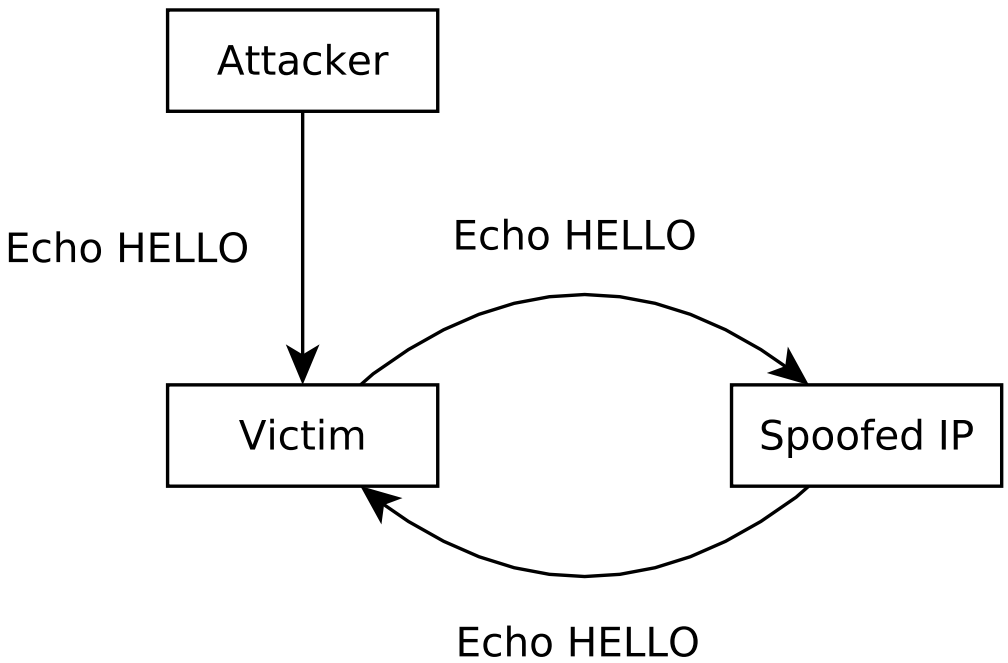
- (D)DoS Malware Countermeasures (intrusion detection, firewalls, authentication)
- Risk to become a zombie
- Example: MyDoom
  - Worm
  - Backdoor
  - DoS
  - Blocking Antivirus

Reflection Attacks

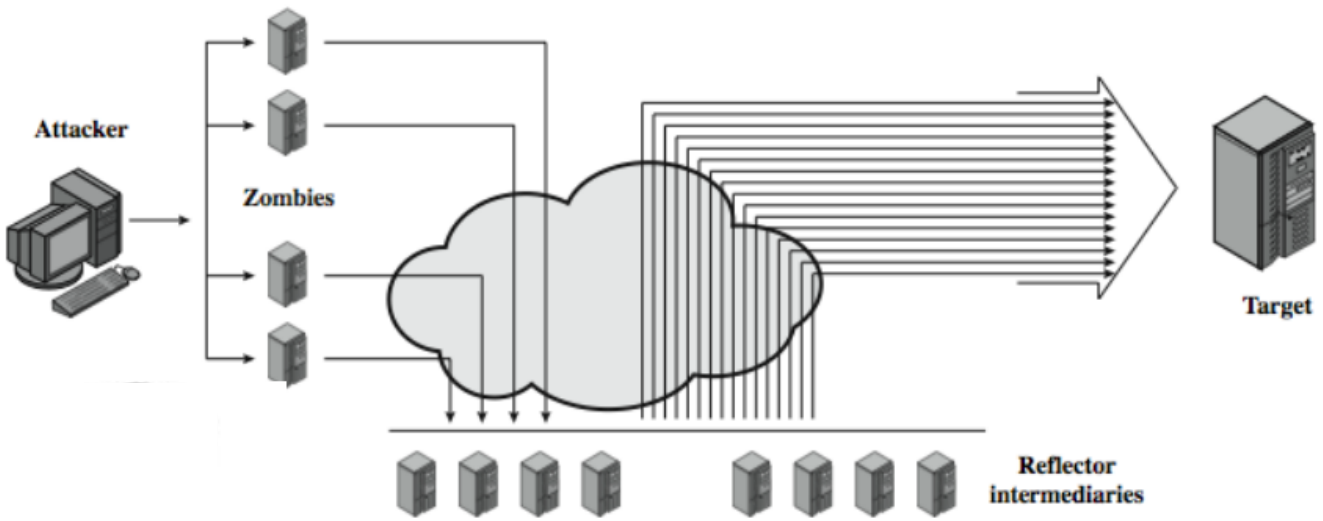
- use normal behavior of network
- attacker sends packet with spoofed source address being that of target to a server
- server response is directed at target
- if send many requests to multiple servers, response can flood target
- various protocols e.g. ICMP, UDP or TCP/SYN
- ideally want response larger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

- See SMS-o-Death by Collin Mulliner
- http://www.mulliner.org/security/sms

SMTP

- Mail reflection
  - A sends: from: c@C to: b@B hello
  - B sends: from: b@B to: c@C vacation message
  - C sends: from: c@C to: b@B vacation message

SMTP

- Mail reflection
  - A sends: from: c@C to: b@B hello
  - B sends: from: b@B to: c@C vacation message
  - C sends: from: c@C to: b@B vacation message
- Quota full

SMTP

- Mail reflection
  - A sends: from: c@C to: b@B hello
  - B sends: from: b@B to: c@C vacation message
  - C sends: from: c@C to: b@B vacation message
- Quota full
- User not allowed e.g. SMTP fax proxy

SMTP

- Mail reflection
  - A sends: from: c@C to: b@B hello
  - B sends: from: b@B to: c@C vacation message
  - C sends: from: c@C to: b@B vacation message
- Quota full
- User not allowed e.g. SMTP fax proxy
- Non existing user

ZIP

- Infinite recursive zip: e.g. r.zip

ZIP

- Infinite recursive zip: e.g. r.zip
- Bombs (e.g. non recursive files)
  - gzip: 100 GB, 97 MB compressed, 1000:1 ration
  - bzip2: 100 GB, 69 KB, $1.6 * 10^6$:1
  - PNG image: 19000 x 19000, 1-bit (45 MB) expand in 24-bit color to 1 GB, 44 KB compressed, 1000:1

XML

```
<?xml version="1.0"?>
<!DOCTYPE lolz [
 <!ENTITY lol "lol">
 <!ELEMENT lolz (#PCDATA)>
 <!ENTITY lol1 "&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;">
 <!ENTITY lol2 "&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&
 <!ENTITY lol3 "&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&
 <!ENTITY lol4 "&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&
 <!ENTITY lol5 "&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&
 <!ENTITY lol6 "&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&
 <!ENTITY lol7 "&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&
 <!ENTITY lol8 "&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&
 <!ENTITY lol9 "&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&
]>
<lolz>&lol9;</lolz>
```

- 1 KB file
- 3GB XML (DOM worst)

Fork Bomb

- PostgreSQL 7.2
- Uses elapsed milliseconds to compute number of thread to spawn
- No limit check

Fork Bomb

- PostgreSQL 7.2
- Uses elapsed milliseconds to compute number of thread to spawn
- No limit check
- What happen if you advance the date of one year?

Fork Bomb

- PostgreSQL 7.2
- Uses elapsed milliseconds to compute number of thread to spawn
- No limit check
- What happen if you advance the date of one year?
- Network Time Protocol attack

Old SMS attack

- Limited number of SMSs (i.e. 9) stored into the phone
- No limit in the reception from the provider
- Usage of zombie and Internet free SMS servers
- Fill and block the SMS reception of a user

HTTP

- Distributed flood (spiders)
- Slashdotted
- Slowloris
  - HTTP server: one thread/one request
  - Open connection
  - Infinitely (and slowly) send HTTP headers
  - Consume the thread pool

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 (ingress filtering)
  - still far too rarely implemented
  - alternatively enforce “unicast severse path”
- 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 (consumes computational resources)
  - or selective or random drop when table full

Attack Prevention

- block IP directed broadcasts
- block suspicious services (e.g. echo) & combinations (DNS to echo port)
- 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