CSE 410/565 Computer Security
Spring 2021

Lecture 18: Network Attacks

Department of Computer Science and Engineering
University at Buffalo
Lecture Overview

• Network attacks
  – denial-of-service (DoS) attacks
    • SYN floods, ICMP floods
    • source address spoofing
    • distributed DoS
  – DNS attacks
  – other types of spoofing
  – session hijacking
DoS Attacks

- **Denial of service attacks** target at denying availability of some service or resource, including
  - network bandwidth
  - system resources
  - application resources

- **Types of DoS attacks**

<table>
<thead>
<tr>
<th></th>
<th>stopping services</th>
<th>exhausting resources</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>local</strong></td>
<td>process crashing</td>
<td>spawning processes to fill process table</td>
</tr>
<tr>
<td></td>
<td>process killing</td>
<td>filling up file system</td>
</tr>
<tr>
<td></td>
<td>system reconfiguration</td>
<td>saturating bandwidth</td>
</tr>
<tr>
<td><strong>remote</strong></td>
<td>malformed packets to crash</td>
<td>packet floods</td>
</tr>
<tr>
<td></td>
<td>buggy services</td>
<td></td>
</tr>
</tbody>
</table>

©Marina Blanton
Overview of Network Protocols

- **IP: Internet Protocol**
  - the main protocol used for routing
  - each IP packet includes the source and destination addresses
  - the protocol is connectionless and unreliable (best effort)
  - TCP and UDP run on top of IP
  - IP is used for routing, data fragmentation and reassembly and error reporting (via ICMP)

- **ICMP: Internet Control Message Protocol**
  - it is used for network reachability testing and to report errors
  - examples: echo request/reply, destination unreachable and time-to-live exceeded messages
Overview of Network Protocols

- **UDP: User Datagram Protocol**
  - transport protocol with minimal guarantees
    - no acknowledgment, no flow control, no message continuation
  - traffic is separated by port number

- **TCP: Transmission Control Protocol**
  - connection-oriented transport protocol
  - partitions data into packets and reassembles them in correct order at the destination
  - transmission is reliable
    - packets are acknowledged and retransmitted if necessary
  - port numbers are used for different services as well
DoS Attacks

• Basic form of DoS
  – attacker sends a large number of packets through a link or to a particular service
  – the goal is to saturate the network or overload the server
  – most requests from legitimate users will be dropped
  – example
    • attacker sends many ICMP echo request packets to a server
    • the server replies with ICMP echo reply packet

• From attacker’s point of view this is unsatisfactory
  – attacker can be easily traced
  – packets sent in response use attacker’s resources
DoS Attacks

- **Solution: source address spoofing**
  - with sufficient privileges to a machine, the source address in IP packets can be set to anything
  - the source address is set to a randomly chosen address
  - replies from the victim machine are scattered across the internet

```
111.11.111.11

attacker
IP 123.45.67.89

echo request from 111.11.111.11

echo request from 22.222.22.222

victim
IP 98.76.54.3

22.222.22.222
```
• Another way to mount a DoS attack is by **TCP SYN flooding**
  – uses the fact that a machine has a limit on the number of open connections
  – allows attacker to deny availability with much less traffic

• **TCP handshake**

```
Client C
SYN_C
SYN_S, ACK_C
ACK_S
```

```
Server S
listening
storing data
waiting
connected
```
DoS Attacks

- TCP SYN flooding attack exploits the fact that server waits for ACKs
  - attacker sends many SYN requests with spoofed source addresses
  - victim allocates resources for each request
    - connection requests exist until timeout
    - there is a fixed bound on half-open connections

```
Client C

SYN_{C_1}
SYN_{C_2}
SYN_{C_3}
SYN_{C_4}
SYN_{C_5}

Server S
listening

storing data and waiting
```
DoS Attacks

- TCP SYN flooding attack (cont.)
  - resources exhausted ⇒ legitimate requests rejected
  - the attack relies on the fact that many SYN-ACK packets will be unanswered
    - an existing host replies to a SYN-ACK packet with RST
    - many IP addresses are not in use
  - the attacker needs to keep sending new SYN packets to keep the table full

- Flooding attacks in general can use any type of packets
  - e.g., ICMP flood, UDP flood, TCP SYN flood

- In any attack with spoofed addresses it is hard to find attacker
DDoS Attacks

- In all of the above attacks, attacker needs to have substantial resources
  - thus attacks are more effective if carried out from many sources
  - they are called distributed DoS (DDoS) attacks

- DDoS attacks often use compromised computers (zombies)
  - attacker compromised machines and builds a botnet
  - attacker instructs the bots to attack the target machine
  - all communication is often encrypted, can be authenticated
  - zombie machines flood the victim
  - spoofing IP addresses is not necessary since it is hard to trace the attacker from the zombie machines
DDoS Attacks

- DDoS attack illustrated

```
Attacker
  |    |
  V    V
Handler Handler
  |    |
  V    V
Zombie Zombie Zombie Zombie Zombie Zombie
  |    |
  V    V
Victim
```
DoS Attacks

- **Other variants of DoS attacks** that use additional machines
  - **reflection**
    - find sites with lots of resources
    - send packets to them with (spoofed) source address of the victim
    - responses flood the victim
    - e.g., echo request $\Rightarrow$ echo response, SYN $\Rightarrow$ SYN–ACK
    - no spurious packets can be observed by other sites
    - attack is harder to detected and defend against
  - **amplification**
    - also sends packets with spoofed addresses to intermediaries
    - now one original packet generates many response packets
• Variants of DoS attacks (cont.)

  – amplification
    • amplification is accomplished by sending a request packet to a broadcast address
    • examples are ICMP echo request packets (smurf program) and UDP packets
    • only connectionless protocols can be used (i.e., not TCP)

  – pulsing zombie floods
    • each zombie is active briefly and then goes dormant
    • zombies take turns in attacking
    • this makes tracing difficult
Defenses Against DoS Attacks

- A significant challenge in defending against DoS attacks is that spoofed addresses are used

- What can be done
  - ingress filtering
    - basic recommendation to check that packets coming from a network have source address within the network’s range
    - ISPs are best suited to perform such filtering
    - despite its simplicity and effectiveness, this recommendation is not implemented by many ISPs
Defenses Against DoS Attacks

- **DoS defenses** (cont.)
  - **SYN cookies**
    - this technique is used to defend against TCP SYN floods
    - after receiving a SYN, information about it is not stored on the server
    - instead it is encoded in the SYN-ACK packet
    - upon receiving ACK, server can reconstruct all information
    - disadvantages: increased server computation
  - **blocking certain packets**
    - many systems block ICMP echo requests from outside of network
    - often IP broadcasts are also blocked from outside
Defenses Against DoS Attacks

- **DoS defenses (cont.)**
  - **limiting packet rates**
    - certain types of packets such as ICMP are rather rare in normal network operation
    - limiting their rate can help mitigate attacks
  - **packet marking**
    - a router marks a small number of packets with its ID
    - for high volume traffic, packets will be marked by most servers on their path to the victim
    - path to the attacker can be reconstructed
    - effectiveness of this technique depends on its wide usage
  - **general good security practices**
• Domain Name System (DNS) allows to map symbolic names to IP addresses
  – the name space is hierarchical
• Hierarchical service

  – root name servers are for top-level domains
  – authoritative name servers are for sub-domains
  – local name resolvers contact authoritative servers when they don’t know a name
• **DNS resource records**
  - “A” record supplies host IP address
  - “NS” record supplies name server for domain

• **DNS caching**
  - DNS responses are cached
    • quick response for repeated translations
    • useful for finding servers as well as addresses
  - negative results are cached
    • save time for nonexistent sites, e.g., misspelling
  - cashed data periodically time out
• DNS lookup using cache
DNS

• DNS is susceptible to cache poisoning attacks
  – change IP address in cache to redirect URLs to fraudulent sites
    • this attack is called pharming
  – example
    • www.yahoo.com NS ns.evil.org (delegate to evil.org)
    • ns.evil.org A 1.2.3.4 (address for evil.org)
  – if resolver looks up www.yahoo.com, the address 1.2.3.4 will be returned
  – the attack is more dangerous than phishing attacks
    • in phishing, users receive email with link to fraudulent website
    • pharming requires no email solicitation, all users go to a wrong address
• DNS cache poisoning
  – the problem is DNS messages are not authenticated
  – some DNS poisoning attacks in the past
    • in January 2005, the address of a large ISP Panix was redirected to a site in Australia
    • in November 2004, Google and Amazon users were sent to Med Network Inc., an online pharmacy

• There are also attacks on DNS reverse address lookup and DNS implementations
  – example: reverse query buffer overrun in BIND releases 4.9 and 8
    • could gain root access, abort DNS service
Domain Name System Security Extensions (DNSSEC) was developed to protect integrity of DNS records

- all DNS responses are authenticated
  - a server signs all answers it provides
  - this prevents forgery such as DNS cache poisoning
- DNSSEC is specified in IETF RFCs 4033, 4034, 4035, and others
- DNSSEC is being deployed slowly due to its perceived overhead
- see dnssec.net and other resources for more information
Other Attacks

- **Address resolution protocol (ARP)**
  - primarily used to translate IP addresses to Ethernet MAC addresses
  - each host maintains a table of IP to MAC addresses

- **ARP spoofing** (or **ARP poisoning**)
  - send fake ARP messages to an Ethernet LAN (no authentication)
    - this causes other machines to associate IP addresses with attacker’s MAC
  - defenses
    - static ARP table
    - DHCP snooping (access control based on IP, MAC, and port)
    - detection: Arpwatch, reverse ARP
Other Attacks

• Session hijacking attacks
  – host-based session hijacking
    • with root privileges can read and write to local terminal devices
  – network-based session hijacking
    • often performed against TCP

• What harm can be done
  – data injection into unencrypted server-to-server traffic such as email exchange, DNS zone transfers, etc.
  – data injection into unencrypted client-to-server traffic such as ftp file downloads and http responses
  – denial of service attacks such as resetting a connection
Other Attacks

- TCP session hijacking
  - each TCP connection has an associated state
    - client and server IP and port numbers, sequence numbers
  - the problem is that it is not difficult to guess state
    - port numbers can be standard
    - sequence numbers are often chosen in a predictable way

- TCP sequence numbers
  - need high degree of unpredictability
    - attacker who knows initial sequence numbers and amount of traffic sent can estimate likely current values
    - send a flood of packets with likely sequence numbers
Other Attacks

- **TCP sequence numbers** (cont.)
  - packets can be injected into existing connection
  - some implementations are vulnerable

- **DoS vulnerability**
  - if attacker can guess sequence numbers for an existing connection, it can send a RST packet to close connection (DoS)
  - naively, success probability is $1/2^{32}$ (32-bit numbers)
  - most systems allow for a large window of acceptable sequence numbers resulting in much higher success probability
  - attack is most effective against long lived connections such as BGP
Defenses

• Cryptographic network protection
  – protocol level solutions
    • adding authentication to protocols would solve many problems
      (various types of spoofing and poisoning)
    • perceived as too expensive for current internet speeds/volumes
  – solutions at network layer
    • use cryptographically random initial sequence numbers, IPsec
    • can protect against session hijacking/data injection and DoS using
      session resets
  – solutions above transport layer
    • tools such as TLS and SSH
    • protect against session hijacking, but not against RST-based DoS
Conclusions

- DoS attacks are common and result in substantial losses
  - a number of defenses are effective, but no perfect solution exists
- DNS attacks can also have a large impact
- Manipulating other protocols and information transmitted on the network can result in various types of other attacks