Denial-of-Service Attacks

Dr. Demetrios Glinos
University of Central Florida

CIS3360 - Security in Computing

Readings

- "Computer Security: Principles and Practice", 3rd Edition, by William Stallings and Lawrie Brown
  - Chapter 7

Outline

- Brief Introduction to the Internet
- Denial of Service
- Flooding Attacks
- Distributed Denial-of-Service Attacks
- Application-Based Bandwidth Attacks
- Reflection and Amplification Attacks
- Defenses Against Denial-of-Service Attacks
- Responding to a Denial-of-Service Attack

The Internet

- Designed during Cold War to be a survivable communications system

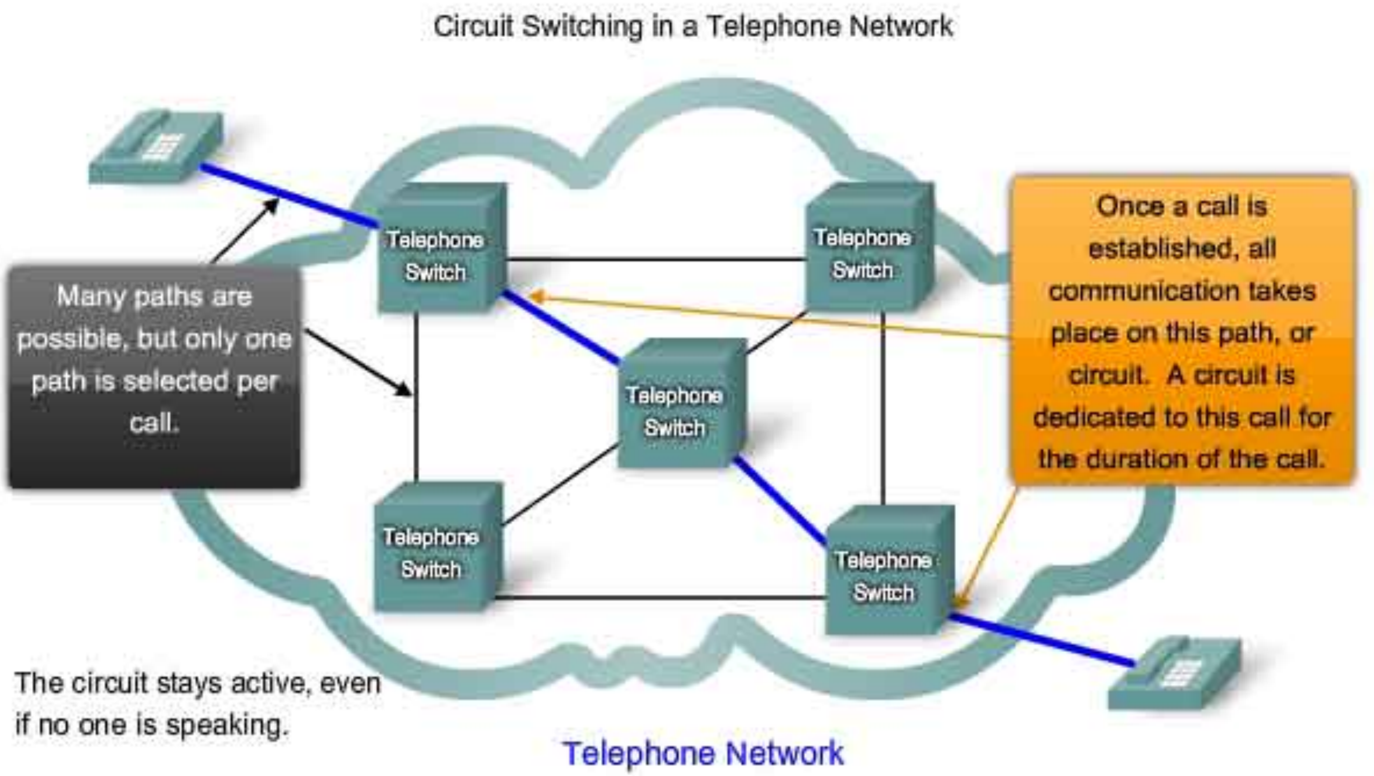
- Uses **packet switching** concept (as opposed to *circuit switching*)
  - Data divided into chunks called “packets”
  - Packets routed individually to their destination
  - Many intermediate communication nodes, so many possible paths
  - Dynamic rerouting if a node should become unavailable

- Packet switching presents unique issues to be addressed:
  - Lost, duplicate, and out-of-order packet receipt

- Protocols are used for different types of messages
  - Examples: HTTP, FTP, PPP

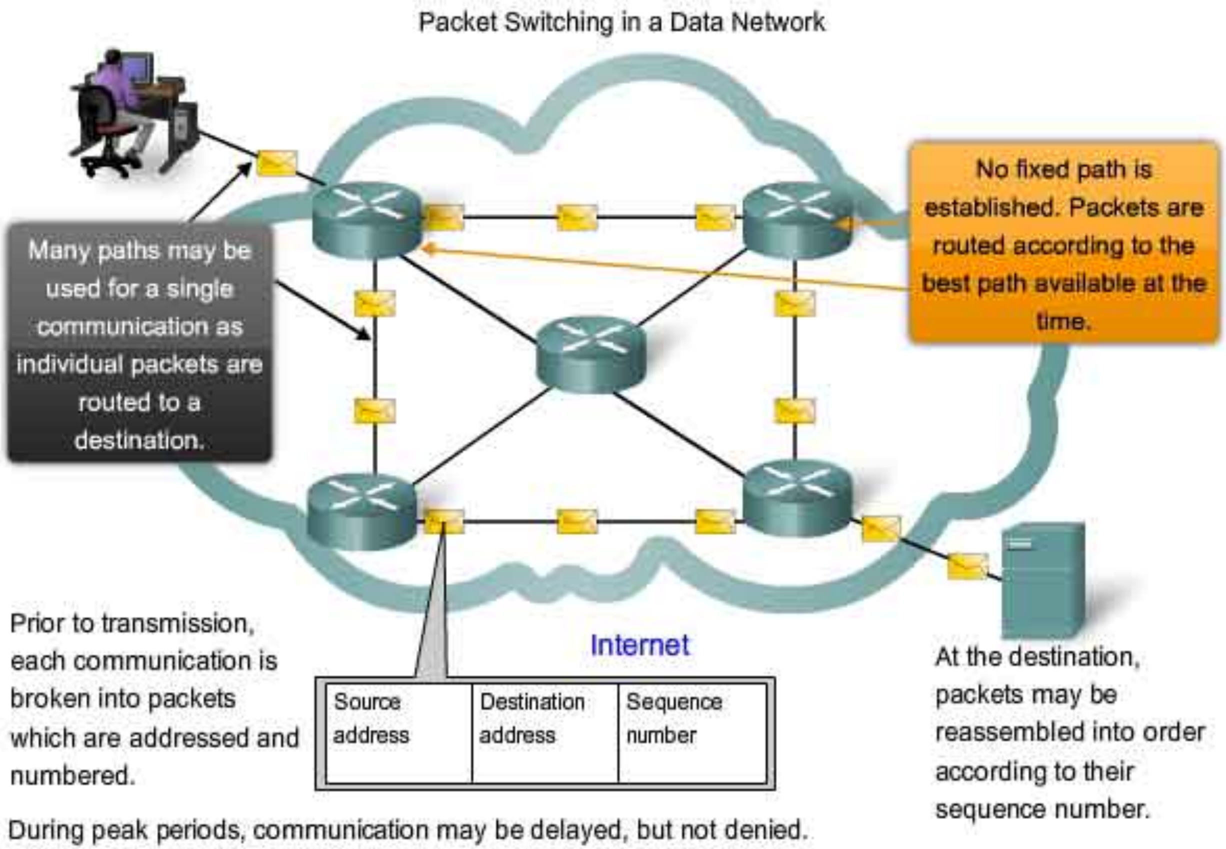
Circuit Switching

- Traditional land line telephone system is best example
- Same operation whether live switchboard operator or automated switch



Packet Switching

- Example: File transfer using the Internet



Layered Network Model

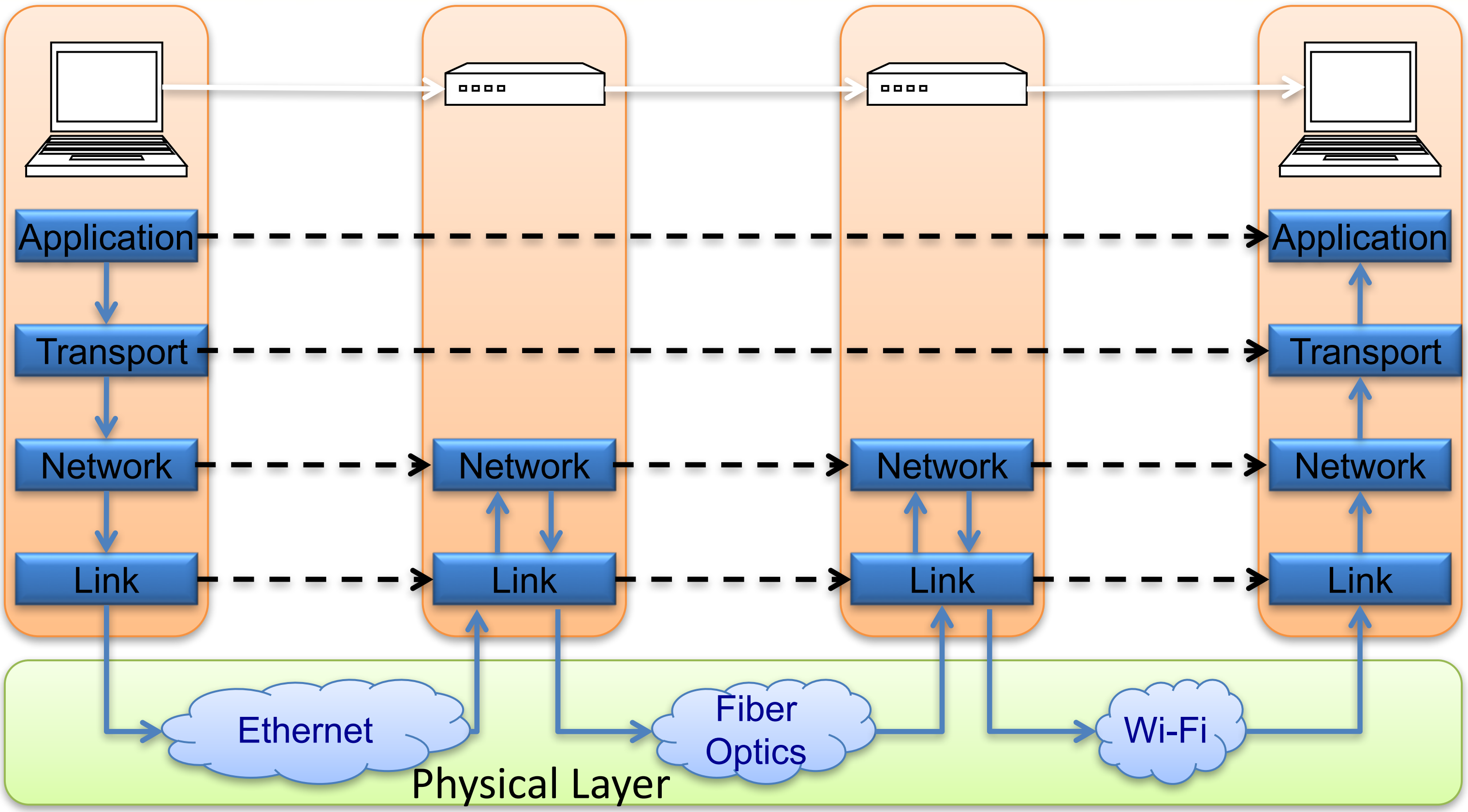
- The Internet implements a layered network architecture called the **"Internet Protocol Suite"**, or more commonly, the **"Internet Protocol Stack"**
  - Examples: TCP and IP protocols

- Network functionality implements a model that consists of a **stack** of **layers**
  - Higher layers use the *services* of lower layers via encapsulation
  - A layer can be implemented in hardware or software
  - The bottommost layer must be in hardware

- A particular network device may implement several layers

- A communication channel between two nodes is established for each layer
  - Actual (physical) channel at the bottom layer
  - Virtual channel at higher layers

Internet Protocol Stack

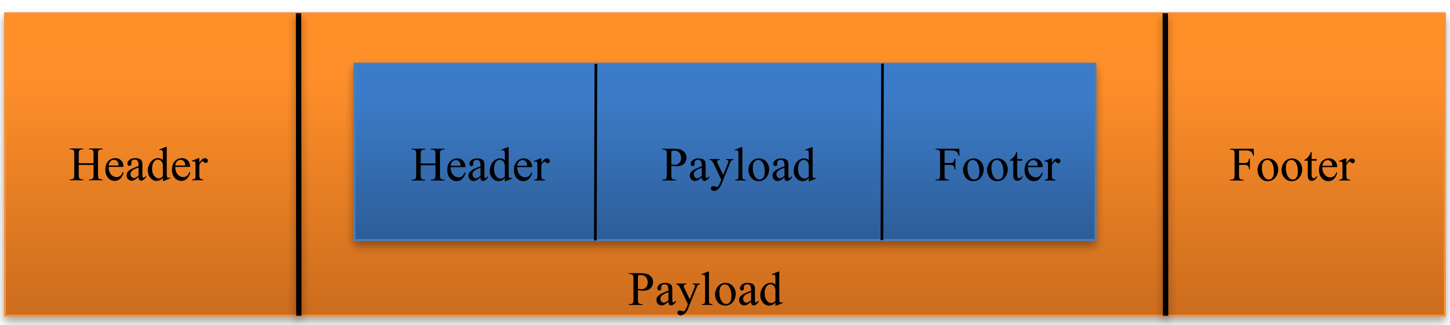


Link layer: ARP, Ethernet, WiFi, use "Media Access Control" (MAC) addresses
Network layer: IPv4 and IPv6 addressing, ICMP Ping, Traceroute, etc.
Transport layer: TCP, UDP protocols
Application layer: DNS, HTTP, FTP, SSL, VoIP, SMTP, IMAP, POP, SOAP

Packet Structure

- A packet typically consists of
  - Control information for addressing the packet: **header** and **footer**
  - Data: **payload**

- A network protocol N2 can use the services of another network protocol N1
  - A packet p1 of N1 is **encapsulated** in a packet p2 of N2
  - The payload of p2 **is** p1
  - The control information of p2 is derived from that of p1
  - The payload of the lower layer packet is the entire higher layer packet



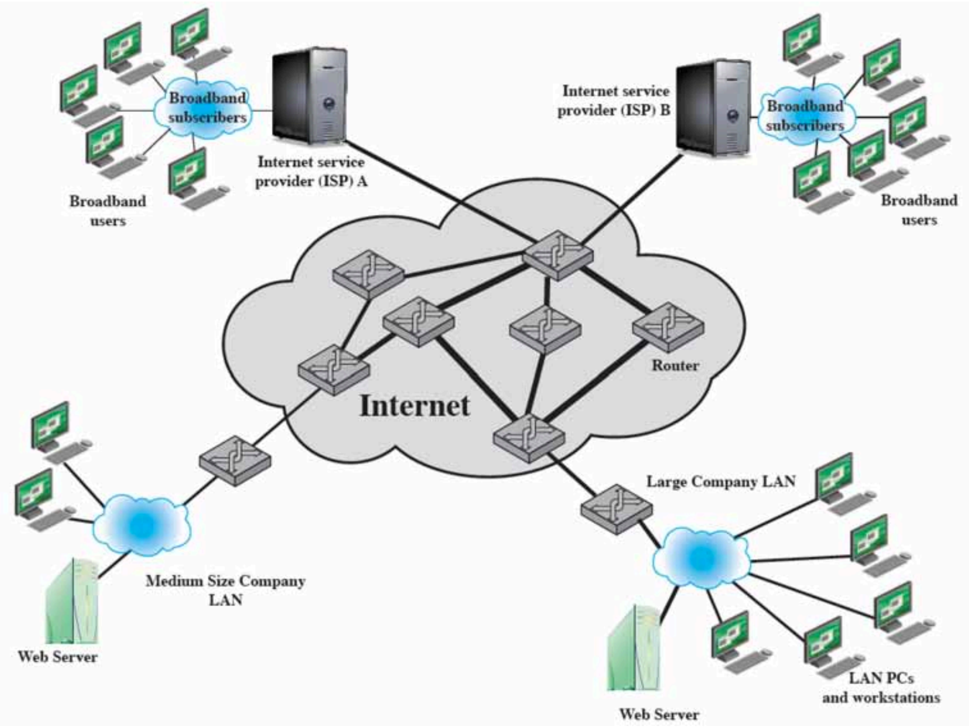
Denial of Service Defined

- **Denial of Service (DoS) is:**
  - 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

- in other words: DoS is an attempt to compromise *availability* by hindering or blocking completely the provision of some service

Targets of DoS Attacks

- Network bandwidth
  - attack attempts to overload connection to ISP
  - ISP can also be targeted
- System resources
  - target is the network handling software
  - overloading buffers
  - open connection tables
  - similar memory data structures



source: Fig. 7.1

- Application resources
  - attempt to overload application with bogus (but still valid) requests
  - effect is to crowd out legitimate users

Flooding Attacks

- Basic idea for all flooding attacks
  - overload network connection with requests that require replies
  - effect is to cause target server to drop traffic from legitimate users
- Almost any type of packet can be used
  - attacker tries to use a packet that is of a type that is permitted to get through to the targeted server
  - common packet types used
    - ICMP
    - UDP
    - TCP SYN

Flooding Attack: Ping Flood

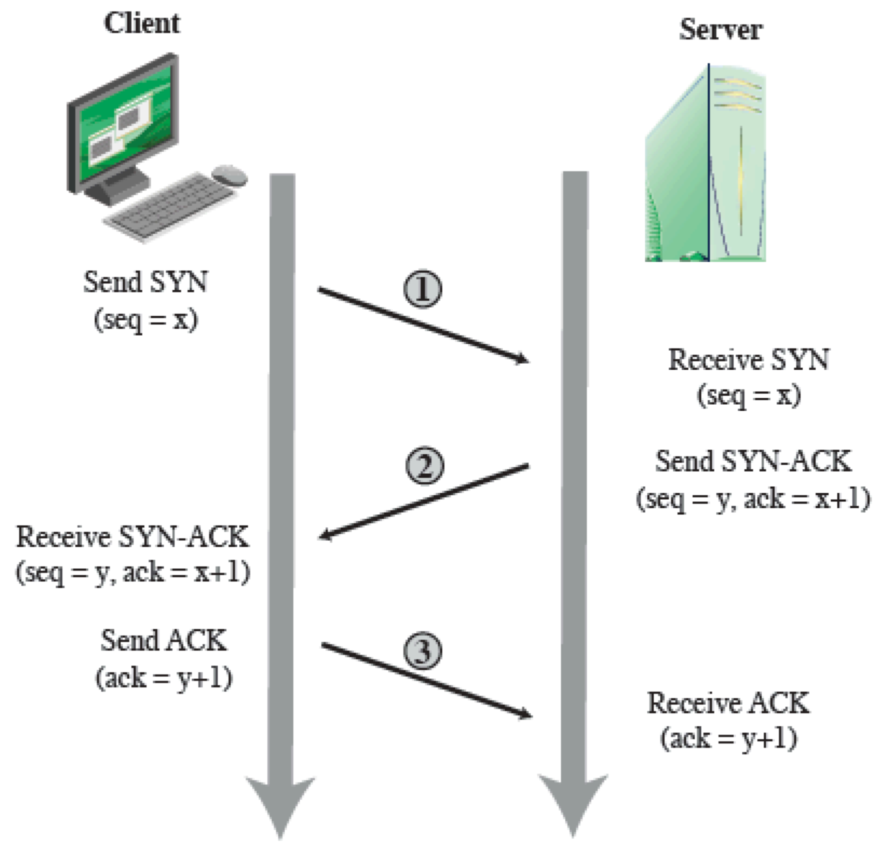
- **ping flood** attack
  - ping is an ICMP packet designed to find out if an IP is reachable
  - Ping uses echo requests to determine if a server can respond
  - Echo requests, if received, ***require*** echo replies

- **source address spoofing** is often used
  - background: source and destination IP addresses are in the IP packet header
  - sender inserts a different (often random) source IP address in packet
  - serves to hide the source of attack
  - also serves to avoid reply traffic returning to source and slowing down the attack

- Ping flood can be executed by
  - A more powerful machine on a less powerful server
  - A distributed DoS attack

TCP Three-Way Handshake

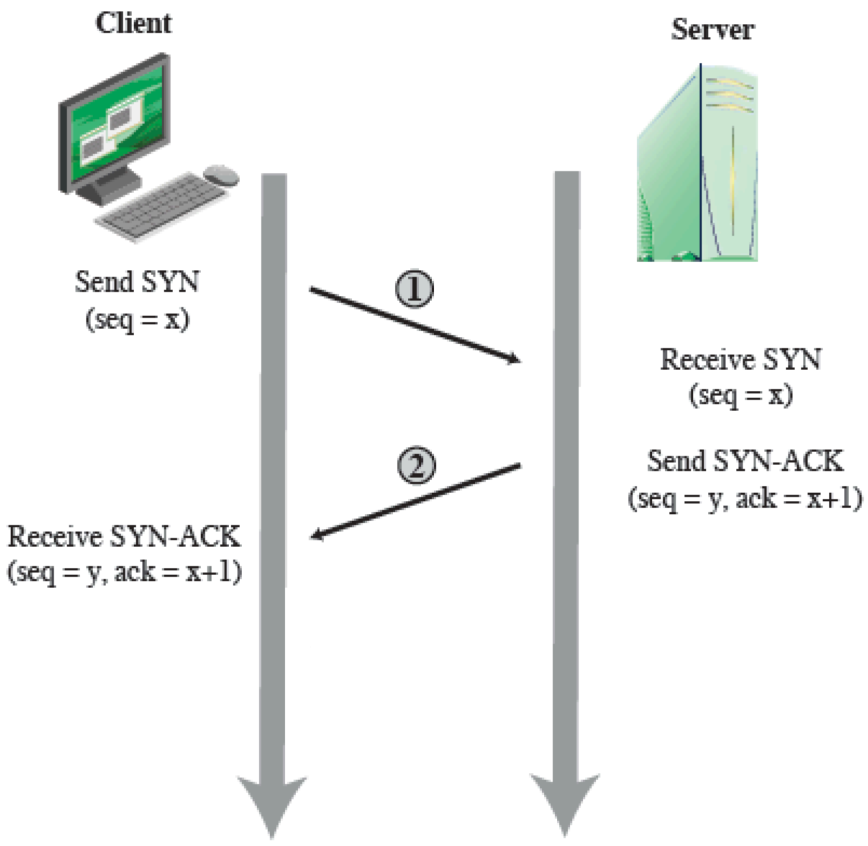
- TCP protocol is used to establish a virtual connection betwen source and destination hosts
  - uses a so-called "three-way handshake" to establish the connection
    - client sends SYN packet
    - server sends SYN-ACK
    - client sends ACK
  - these packets contain sequence and acknowledgement numbers
    - used to assure each host that it is connecting to the correct other host



source: Fig. 7.2

Flooding Attack: TCP SYN Flood

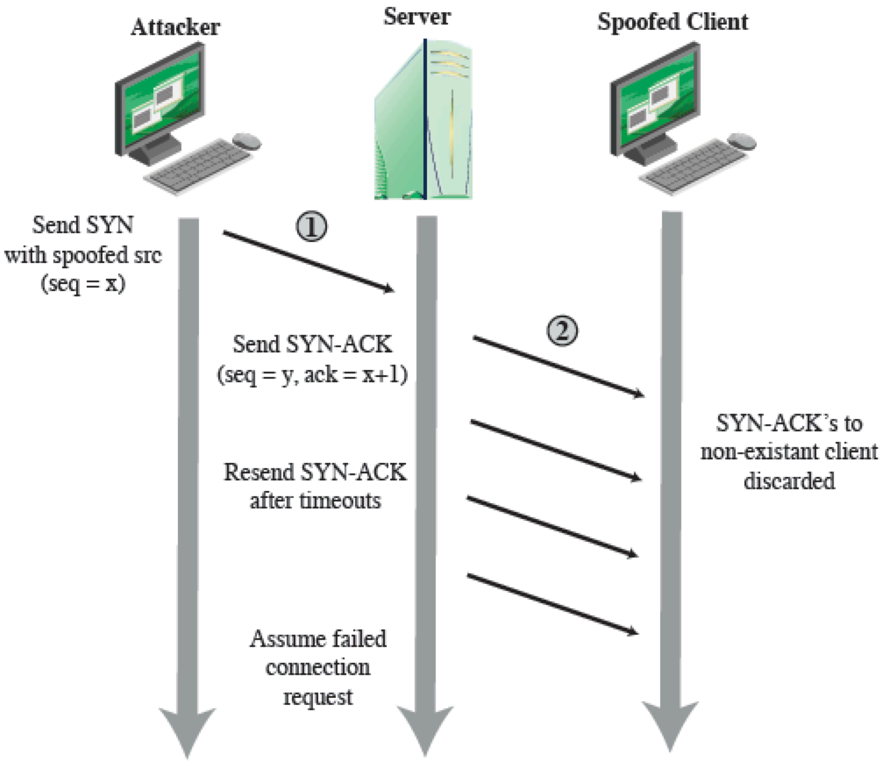
- **SYN flood** attack
  - target is the server table that is used to manage TCP connections
  - attacker sends a flood of TCP connection requests to the target server
  - attacker ignores SYN-ACKs received from server and does not send any ACKs
  - connection table is filled with bogus connections
    - causes legitimate connection requests to be dropped



source: Fig. 7.2 (modified)

Compare: SYN Spoofing

- **SYN spoofing** attack
  - target is the server table that is used to manage TCP connections
  - attacker sends a large number of SYN packets with spoofed source addresses
  - server sends SYN-ACKs to the spoofed address
  - spoofed address doesn't respond (because it is either nonexistent or itself under attack)
  - as a result, connection table fills up and legitimate requests are dropped



source: Fig. 7.3

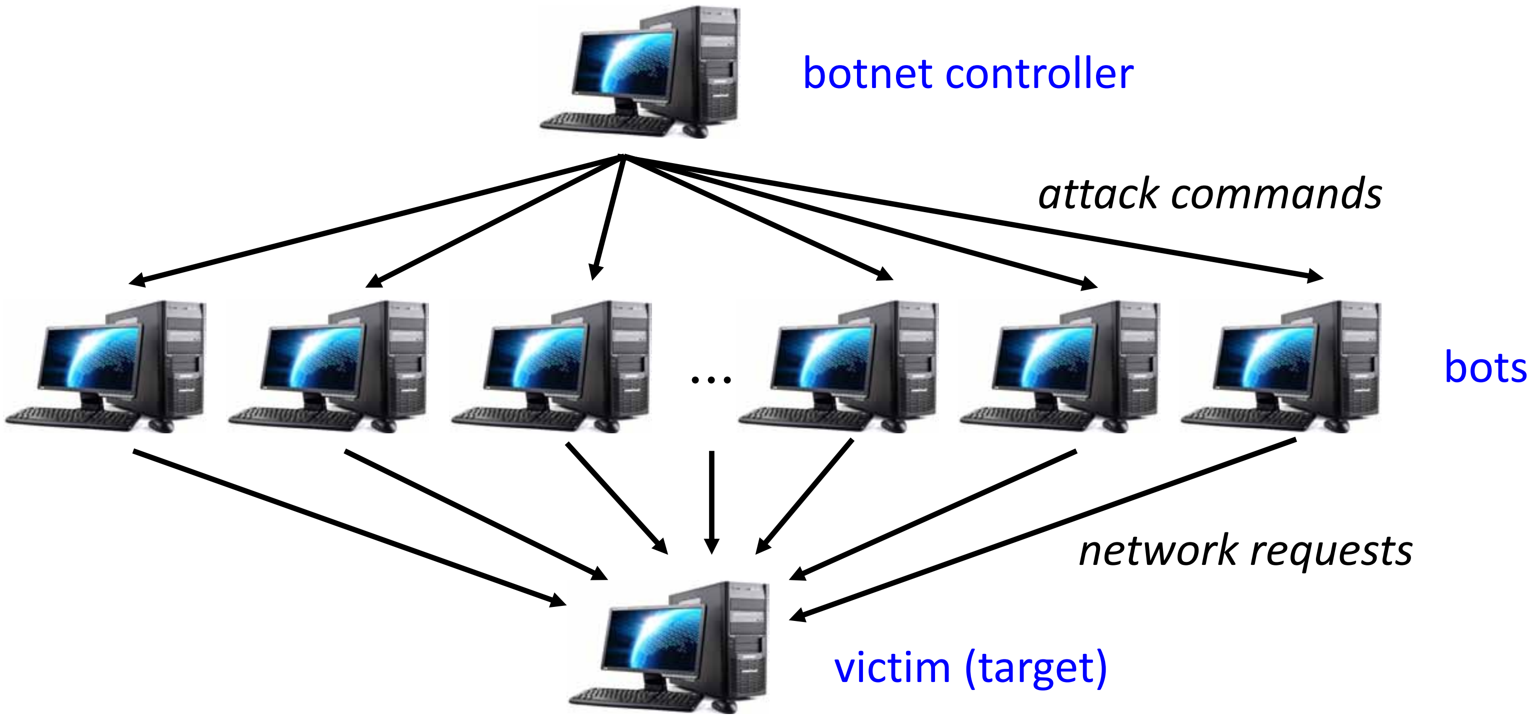
Flooding Attack: UDP Flood

- **User Datagram Protocol**
  - at transport layer, manages connections between processes
  - but is ***connectionless***
    - No guarantee of delivery, order or correctness of delivery
      - *delivery is on a "best effort" basis*
    - Sequence numbers are not used
    - Uses a 16-bit checksum
  - suitable for applications where speed is more important than data integrity
    - Example: streaming media, VoIP (Voice over IP)

- **UDP flooding** attack
  - attacker's goal is to use up connection bandwidth
  - flood directed to a particular port (hence, service) on server, whether or not it is running on the server
  - flood of traffic to this port (and replies, if any) uses up bandwidth
    - available bandwidth for legitimate traffic is degraded

Distributed Denial-of-Service (DDoS)

- Modern servers are now generally too powerful to be subject to DoS from a single machine.
- A **DDoS attack** is a DoS attack that is executed by a **botnet**
- The botnet amplifies the effectiveness of an attack by a single server, but generally on a much larger scale.



DDoS Case Study

- DDoS attack on 10/21/2016
- Target DNS servers operated by DYN Corporation
- Effect: temporary outages at many popular sites
  - Twitter, Netflix, Spotify, Reddit
- Source was a botnet of IoT ("Internet of Things") devices
  - CCTVs, security cameras, etc., mostly outside US
  - most were traced to a particular Chinese company's products that had default passwords

Application-Based Bandwidth Attacks

- **HTTP flood**
  - attack is usually in the form of a DDoS by a botnet
  - attacker floods a Web server with valid HTTP requests
    - e.g., to download a large file
  - effect is to consume memory, processing, and transmission resources
  - attacker can also use **spidering**, to follow links in a recursive way and pursue the attack further.
- **Slowloris attack**
  - attacker sends many legitimate HTTP requests to server, but the sessions are kept open
  - each request keeps one processing thread on server busy waiting for the session to end
  - eventually, the Web servers's connection capacity is fully occupied by such requests, crowding out requests from legitimate users
  - since requests are legitimate, not generally detected by intrusion detection systems that rely on signatures

Reflection and Amplification

- Reflection and amplification attacks
  - use network systems functioning normally
  - unlike DDoS, which uses compromised systems

- **Reflection attack**
  - uses an intermediary to "reflect" an attack in the direction of the target
  - attacker sends packets to a known service on intermediary
  - packets spoof source address
  - intermediary responds to the spoofed address
  - the spoofed server is the target

  - can use many different types of packets
    - e.g., TCP SYN flood to high-capacity server, with source address spoofed to be the IP address of the intended target

Amplification: Smurf Attack

- **Amplification attack**
  - uses some means to generate multiple response packets for each attack packet
  - classic case: **Smurf attack**
    - Takes advantage of a broadcast capability which some LANs are are still configured with
      - IP address ending in n.n.n.255 was traditionally a broadcast address
    - Attacker sends ICMP packets (e.g., echo requests) to the *broadcast IP address*, but with the source IP spoofed to be that of the target server.
    - All recipients of the broadcast send replies to the server, overwhelming it
    - Defense is to configure hosts and routers to ignore broadcast requests

Defenses Against Denial-of-Service Attacks

- DoS attacks cannot be prevented entirely
  - high traffic volumes may be legitimate
    - high publicity, or current popularity
      - National Hurricane Center site during hurricane
      - sports sites during major events like Olympics or World Cup

- Lines of defense
  1. prevention
  2. detection and filtering
  3. source traceback
  4. attack reaction

DoS Attack Prevention

- An incomplete list of prevention measures
  - Block spoofed source addresses
    - use info from ISPs to detect bogus addresses in their address space, and block them at the enterprise firewall
  - Modify TCP connection handling code
    - drop half-open connections when connection table overflows
    - SYN Cookies (which contain encrypted sequence numbers in the SYN/ACKs themselves instead of in memory)
    - Not allocating resources for a TCP connection until ACK packet received
  - Disable LAN broadcast addresses
  - Manage application access with a graphical puzzle
    - CAPTCHA - "**C**ompletely **A**utomated **P**ublic **T**uring test to tell **C**omputers and **H**umans **A**part")
  - Use mirrored and replicated servers for high performance and availability

Responding to DoS Attacks

- Identify the 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 planning legal action
- Implement a contingency plan
  - switch to alternate backup servers
  - commission new servers at a new site with new addresses
- Update incident response plan
  - analyze the attack and response for better handling in the future