Introduction to Computer Security

Chapter 7: Denial-of-Service Attacks

Chi-Yu Li (2019 Spring)
Computer Science Department
National Chiao Tung University

Denial of Service (DoS) Attack

- NIST [CICH12] defines DoS attack:

> "A **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."

[CICH12] Chichonski, P., et al. *Computer Security Incident Handling Guide*. NIST Special Publication 800-61, August 2012.

Outline

- Denial-of-Service Attacks
- Flooding Attacks
- Distributed Denial-of-Service Attacks
- Application-Based Bandwidth Attacks
- Reflector and Amplifier Attacks
- Defenses Against DoS Attacks
- Responding to a DoS Attack

Denial-of-Service Attacks

- Attempts to compromise availability by hindering or blocking completely the provision of some services
  - e.g., flooding a Web server with so many spurious requests
- Nowadays: distributed denial-of-service (DDoS) attacks
  - Due to Internet bandwidth growth
  - 400 MBps (2002) → 100 GBps (2010) → 300 GBps (Spamhaus in 2013)
  - 50 GBps: power enough to exceed the bandwidth capacity of any target

DoS Attacks: Categories of Resources

Network bandwidth

Relates to the capacity of the network links connecting a server to the Internet

For most organizations this is their connection to their Internet Service Provider (ISP)

System resources

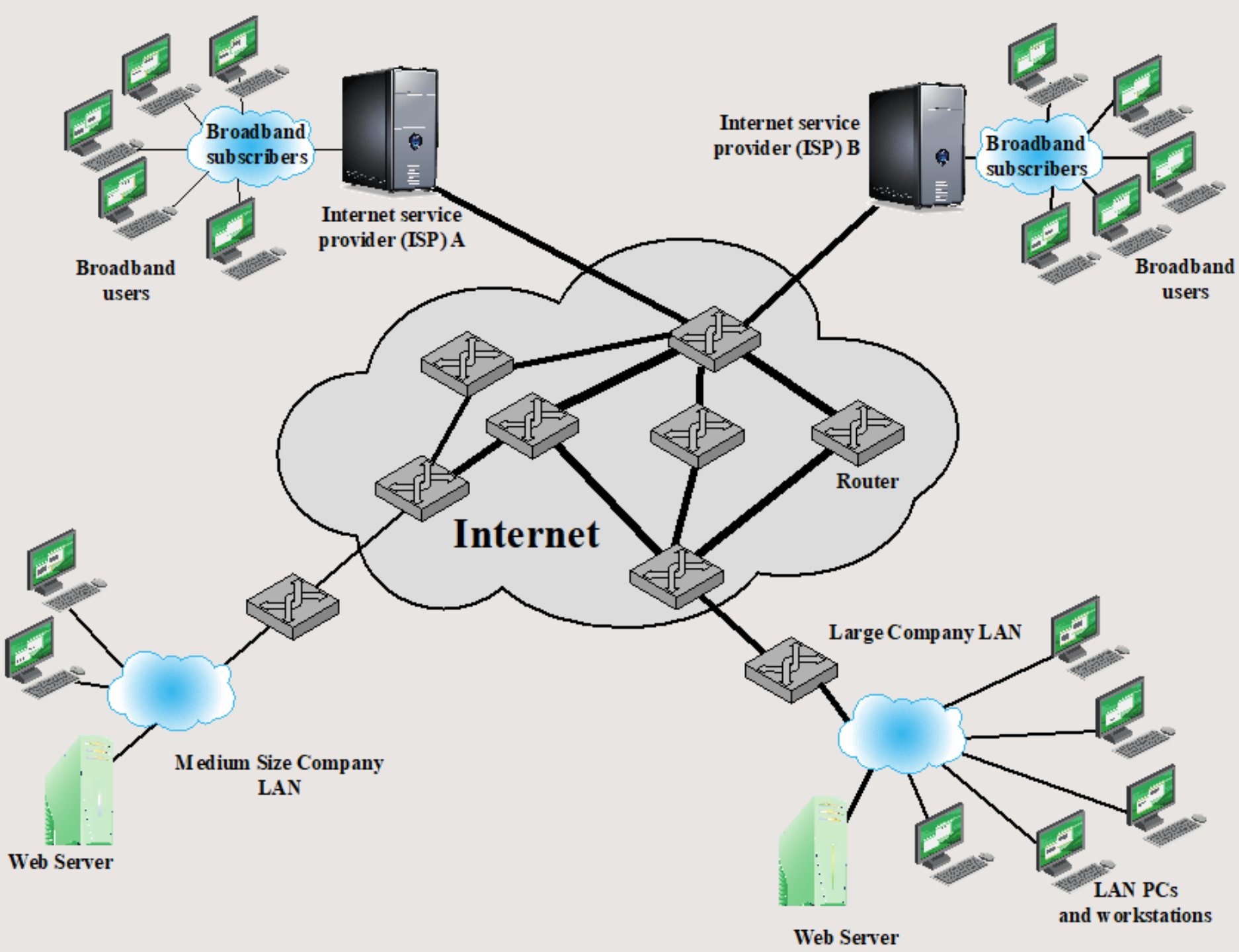
Aims to overload or crash the network handling software

Application resources

Typically involves a number of valid requests, each of which consumes significant resources

Thus limiting the ability of the server to respond to requests from other users

Example: DoS Attacks on Network Bandwidth



Classic DoS Attack I

- Ping flooding attack
  - Traffic: ICMP echo request and response packets
  - Goal: overwhelm the capacity of network connection to the target organization
  - Traffic can be handled by higher capacity links on the path, but packets are discarded as capacity decreases
- Two disadvantages from the attacker's perspective
  - Source of the attack is clearly identified
  - Attack reflection at the source system
    - Network performance will be noticeably affected

Source Address Spoofing

- The use of forged source addresses
  - Usually via the *raw socket interface* on OS
- Consider the Ping flooding attack
  - Same congestion would result in the router connected to the final, lower capacity link
  - But, ICMP echo response packets would no longer be reflected back
  - Randomly spoofed source addresses: backscatter traffic
- Make attacking systems harder to be identified
  - Just inspecting each packet's header is not sufficient to identify source
  - Requiring network engineers to specifically query flow info from routers

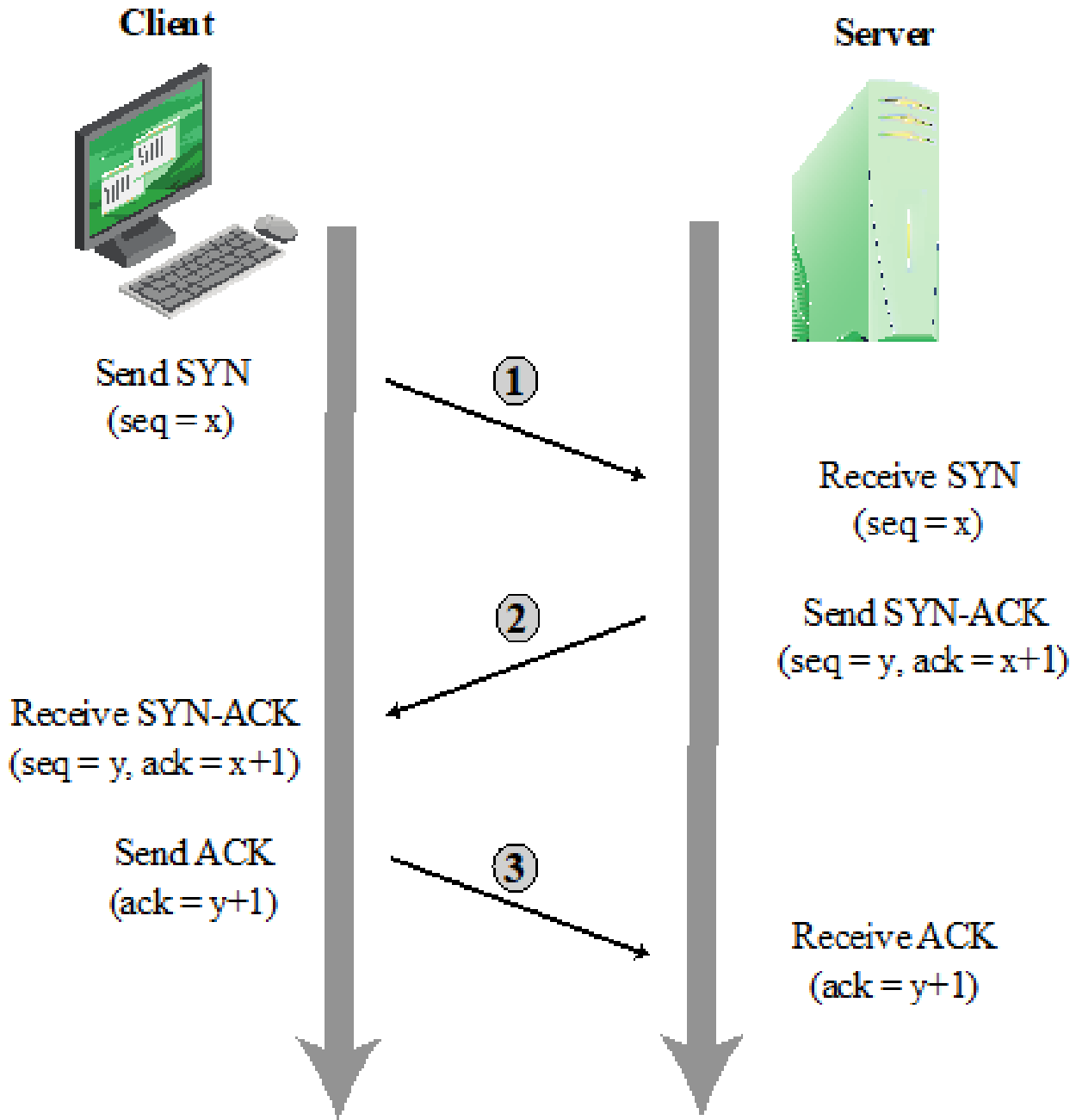
Why Such Easy Forgery of Source Addresses is Allowed?

- Development of TCP/IP: generally cooperative, trusting environment
  - Simply does not include the ability to ensure the valid source address
- How to address it?
  - Impose filtering on routers to ensure it
  - As close to the originating system as possible (e.g., borders of the ISP's connection)
- This has been a long-standing security recommendation (RFC 2827); however, many ISPs DO NOT implement such filtering

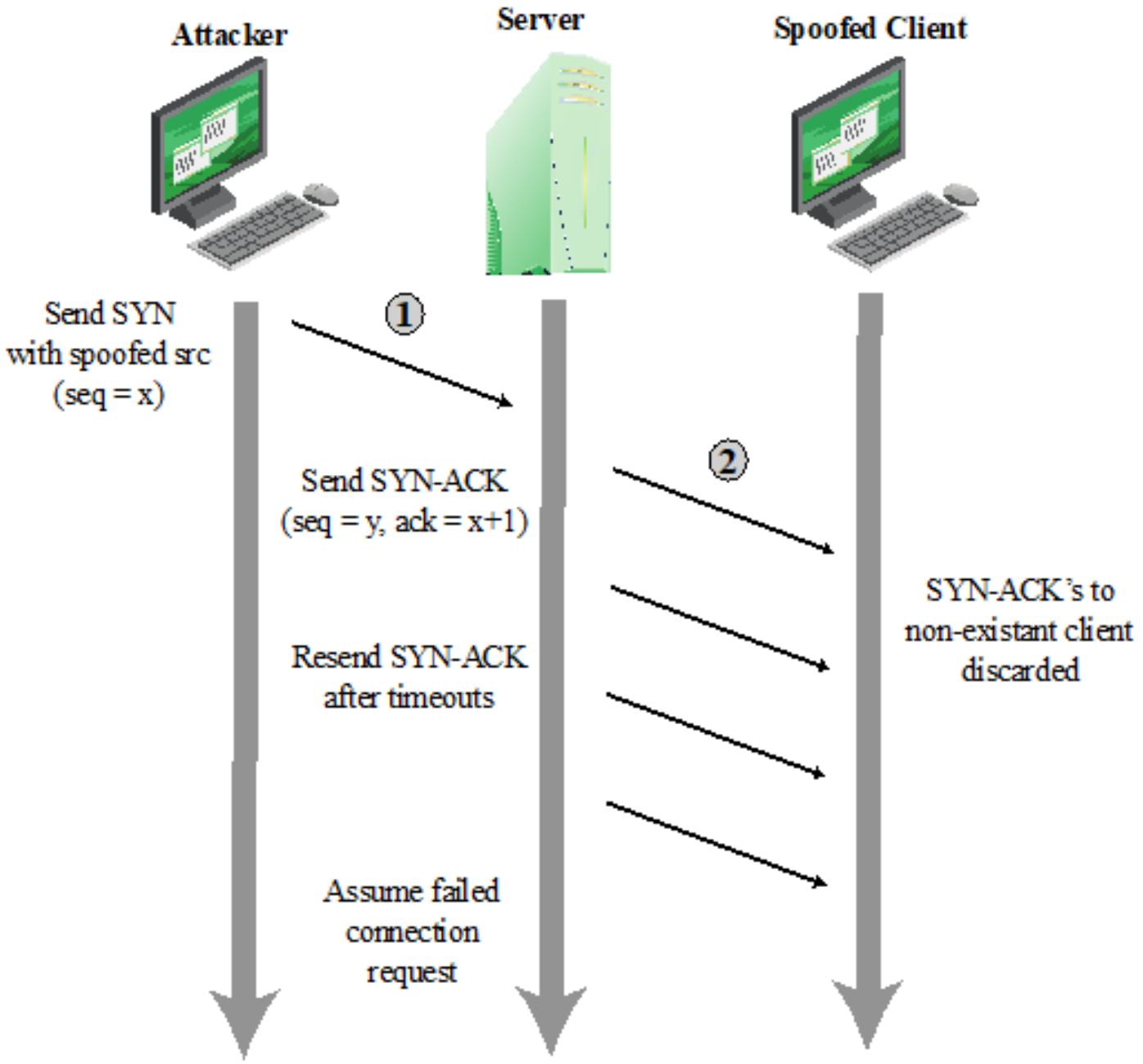
Classic DoS Attack II

- SYN spoofing
  - Goal: attacking the ability of a network server to respond to TCP connection requests
  - How? Overflowing the tables used to manage such connections
  - Generating a very large number of forged connection requests to the server
  - Legitimate users are denied access to the server
- Different from the basic flooding attack
  - Actual volume of SYN traffic can be comparatively low
  - High enough to keep the known TCP connections table filled

TCP Three-way Connection Handshake



TCP SYN Spoofing Attack



Flooding Attacks

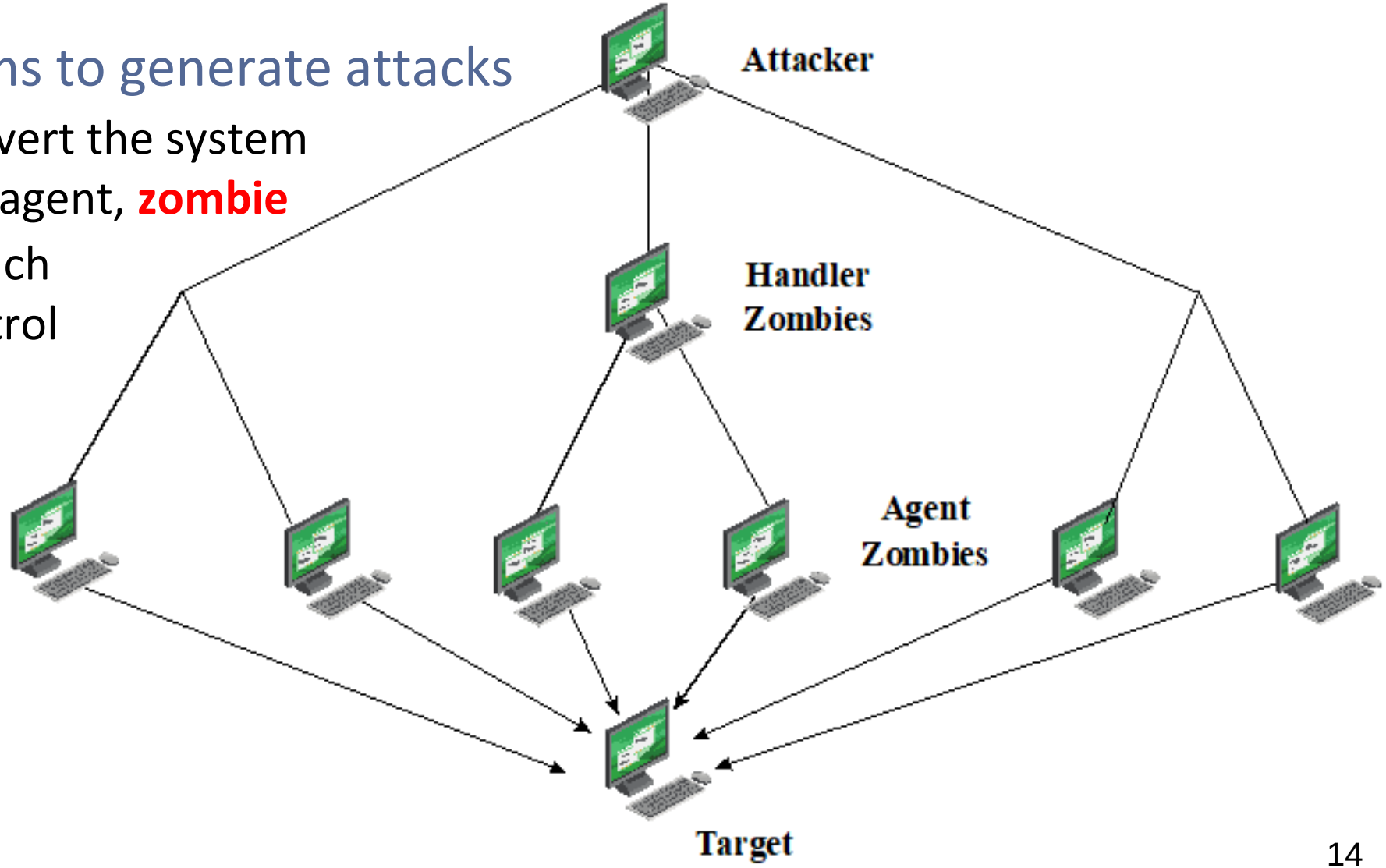
- Classified based on network protocols
- Intent: overload the network capacity on some link to a server
  - Virtually any type of network packets can be used, e.g., HTTP

ICMP flood	• Ping flood using ICMP echo request packets • Traditionally network administrators allow such packets into their networks because ping is a useful network diagnostic tool
UDP flood	• Uses UDP packets directed to some port number on the target system, e.g., DNS
TCP SYN flood	• Sends TCP packets to the target system • Total volume of packets is the aim of the attack rather than the system code

However, the flooding attacks are limited by a single system!!

Distributed Denial of Service (DDoS) Attacks

- Using multiple systems to generate attacks
  - Using malware to subvert the system and to install an attack agent, **zombie**
  - Large collections of such systems under the control of one attacker: **botnet**



Distributed Denial of Service (DDoS) Attacks

- Earlies and best-known DDoS tool: Tribe Flood Network (TFN) – Mixter
  - Original variant from the 1990s: only exploited Sun Solaris systems
  - Rewritten as (TFN2K): could run on UIX, Solaris, and Windows NT
  - Two-layer command hierarchy
  - Capable: ICMP flood, SYN flood, UDP flood, and ICMP amplification
  - Didn’t spoof source; rather, relied on many compromised systems
  - Communication between the handler and its agent was encrypted
- Current DDoS tools
  - Using IRC or HTTP servers to communicate with agents, instead of the handler
  - Hindering analysis of command traffic: agents authentication

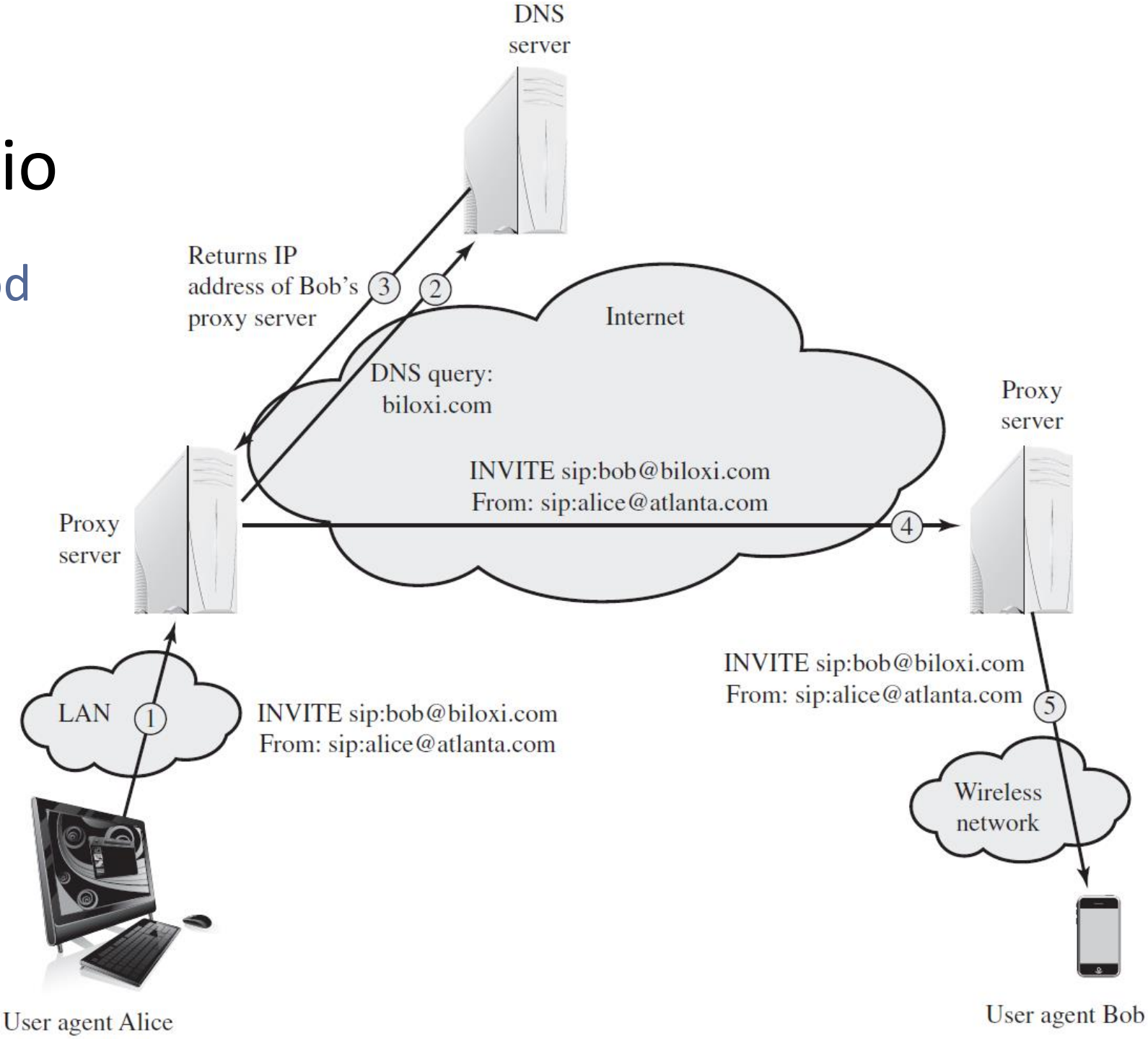
- Best defense: prevent your systems from being compromised

Application-based Bandwidth Attacks

- Force the target to execute resource-consuming operations
- SIP (Session Initiation Protocol) Flood
  - SIP: a protocol for call setup in Voice over IP (VoIP)
    - A text-based protocol with a syntax similar to HTTP
    - Two types: requests and responses
- HTTP-based attacks
  - HTTP Flood
  - Slowloris

SIP INVITE Scenario

- What does the SIP flood attack exploit?
  - A single IMVITE request triggers considerable resource consumption
  - Attacks in two ways
    - Server resources for processing the INVITE requests
    - Network capacity is consumed
  - Legitimate incoming calls are not allowed



HTTP-based Attacks: HTTP Flood

- An attack that bombards Web servers with HTTP requests
- Designed to consume considerable resources
  - How?
  - e.g., an HTTP request to download a large file: consuming memory, processing, and transmission resources
  - Causing the Web server to
    - Read the file from hard disk
    - Store it in memory
    - Convert it into a packet stream
    - Transmit the packets
  - e.g., Spidering: bots starting from a given HTTP link and following all links on the provided Web site in a recursive way

HTTP-based Attacks: Slowloris

- Monopolizing all of the available request handling threads on the Web sever
  - Common server technique: using multiple threats to support multiple requests
  - Sending HTTP requests that never complete
- How? Need to understand the HTTP protocol
  - A blank line must be used to indicate the end of the request headers and the beginning of the payload (RFC2616)
  - Step 1: sending an incomplete request that does not have that blank line
  - Step 2: sending additional header lines periodically to keep the connection alive
- Not detected by signature-based solutions: legitimate HTTP traffic

Outline

- Denial-of-Service Attacks
- Flooding Attacks
- Distributed Denial-of-Service Attacks
- Application-Based Bandwidth Attacks
- Reflector and Amplifier Attacks
- Defenses Against DoS Attacks
- Responding to a DoS Attack

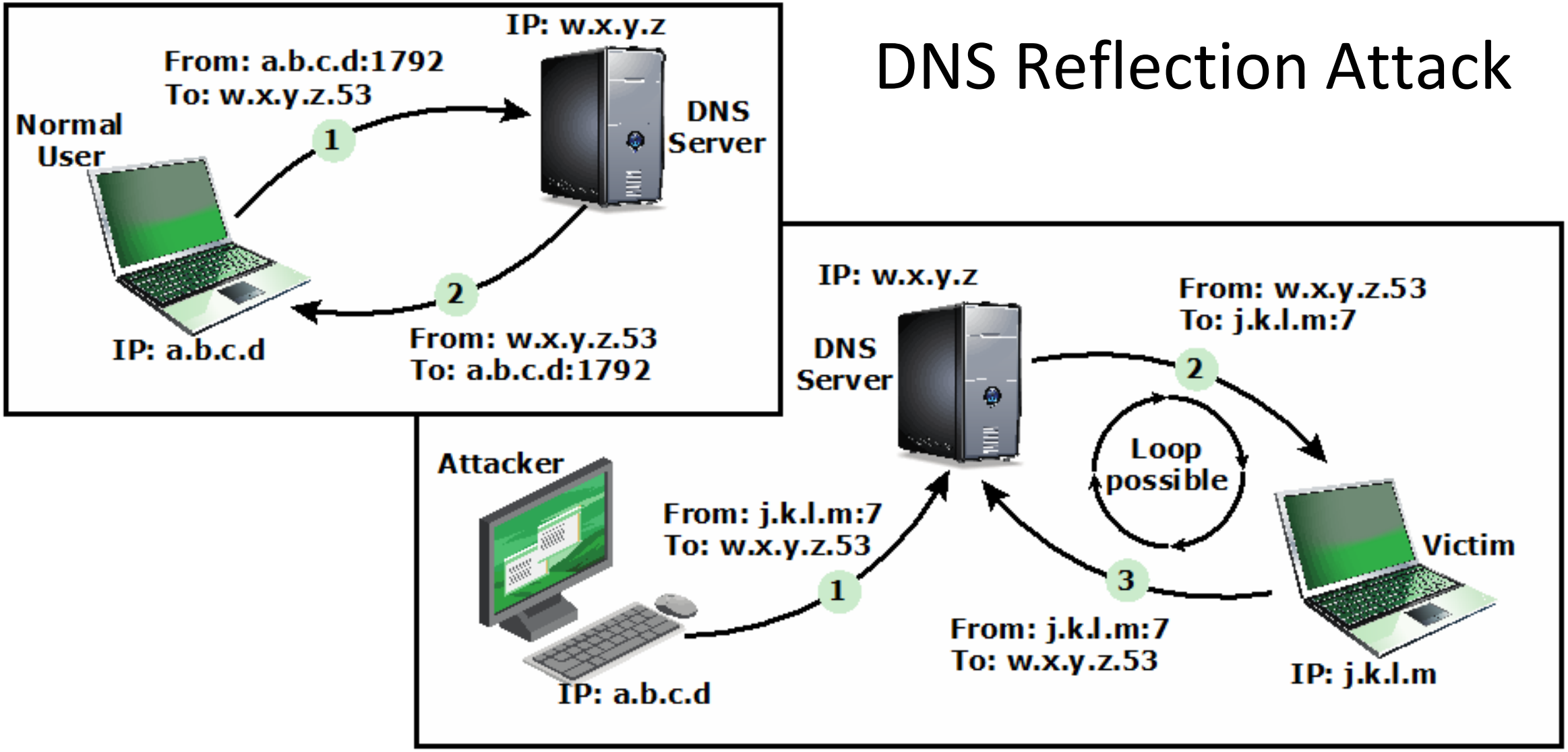
Reflector and Amplifier Attacks

- Normal server systems are used as intermediaries for attacks
  - Easier to deploy: their handling of the packets is entirely conventional
  - Harder to trace back to the actual attacker
- Reflection attacks
- Amplification attacks

Reflection Attacks

- Sending packets to a known service on the intermediary with a spoofed source address of the actual target system (victim)
  - The intermediary sends response to the target
  - Intermediary systems: high-capacity network servers or routers
- Moreover, attempting to create a larger response packet
  - Any generally accessible UDP service could be used
    - e.g., DNS, SNMP, or ISAKMP services
  - Also for TCP services
    - e.g., TCP SYN: SYN-ACK and RST packets
- Fundamental issue: spoofed-source packets

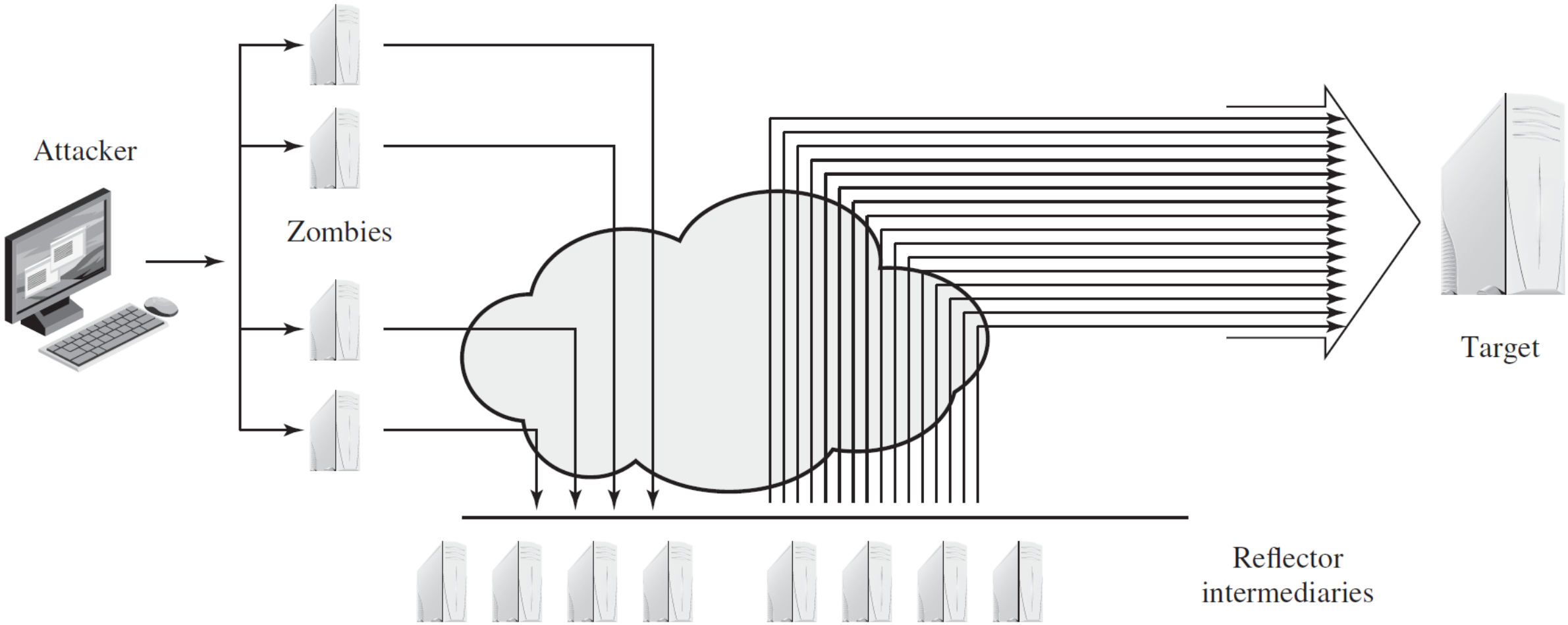
DNS Reflection Attack



Amplification Attacks

- Directing the request to the broadcast address for some network
  - Using a service handled by large numbers of hosts on the intermediate network
    - e.g., ping (ICMP echo request), suitable UDP services (echo)
  - But, TCP services cannot be used for this attack. Why?
- Defense: not allow directed broadcasts to be routed into a network from outside
- Other defenses:
  - Blocking spoofed source addresses
  - limiting network services (e.g., echo and ping) from being accessed from outside

Example: A Flood of Responses



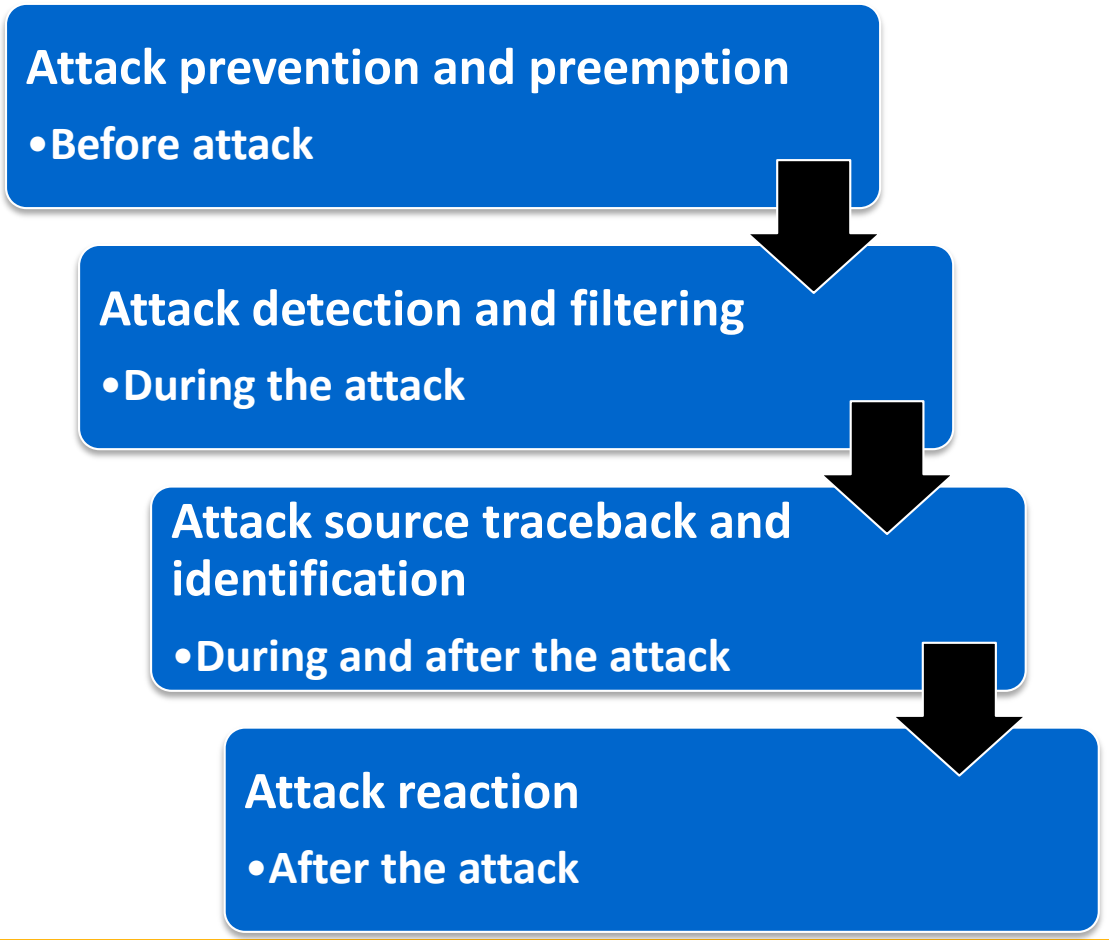
DNS Amplification Attacks

- Using packets directed at a legitimate DNS server as the intermediary
- Amplification: converting a small request to a much larger response
  - Contrasting with the original amplifier attacks: responses from multiple systems
  - e.g., 60-byte UDP request → a 512-byte UDP response
- Targeting servers that support the extended DNS features (e.g., IPv6)
  - Much larger responses of over 4000 bytes are allowed

DoS Attack Defenses

- These attacks cannot be prevented entirely
- High traffic volumes may be legitimate
  - Activity on a very popular site
  - Described as slashdotted or flash crowd
- Usual response: provision of significant excess network bandwidth and replicated distributed servers
  - But, high implementation cost

Four lines of defense against DDoS attacks



DoS Attack Prevention: Flooding Attacks

- Blocking spoofed source addresses
  - On routers as close to source as possible
- Filters may be used to ensure path back to the claimed source address
  - Filters must be applied to traffic, before it leaves the ISP's network or at the point of entry to their network
- Imposing limits on the rate of some specific types of packets

DoS Attack Prevention: SYN Spoofing Attacks

- Modified versions of TCP connection handling code
  - Cryptographically encode critical information in a cookie that is sent as the server's initial sequence number
    - Legitimate client responds with an ACK packet containing the incremented sequence number cookie
    - Reconstructing the info about the connection based on the cookie, and then creating an entry in the TCP connection table
    - Pros and cons?
  - Selective drop: dropping an entry for an incomplete connection from the TCP connections table when it overflows
  - Modifying parameters: table size and timeout period

DoS Attack Prevention: Others

- Block IP directed broadcasts
- Block suspicious services and combinations
- Manage application attacks with a form of graphical puzzle (captcha) to distinguish legitimate human requests
- Good general system security practices
- Use mirrored and replicated servers when high-performance and reliability is required

Responding to DoS Attacks

Good Incident Response Plan

- Details on how to contact technical personnel for ISP
- Needed to impose traffic filtering upstream (target attack sources)
- Details of how to respond to the attack

- Anti-spoofing, directed broadcast, and rate limiting filters should have been implemented
- Ideally have network monitors and IDS to detect and notify abnormal traffic patterns

Responding to DoS Attacks

- Identify type of attack
  - Capture and analyze packets
  - Design filters to block attack traffic upstream
  - Or identify and correct system/app bug
- Ask ISP to trace packet flow back to source
  - May be difficult and time consuming
  - Necessary if planning legal action
- Implement 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 the response for future handling

Questions?