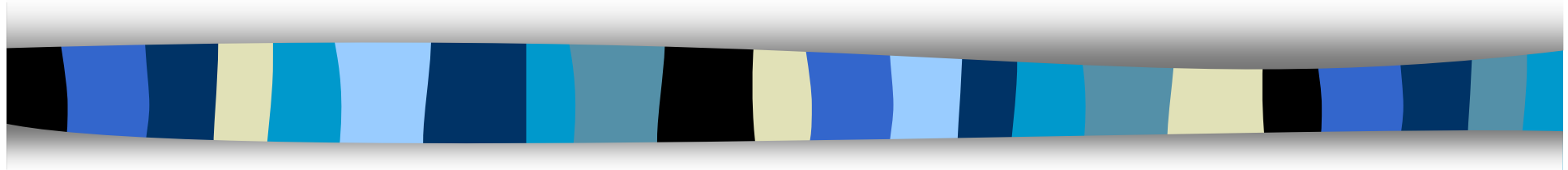


Information Security

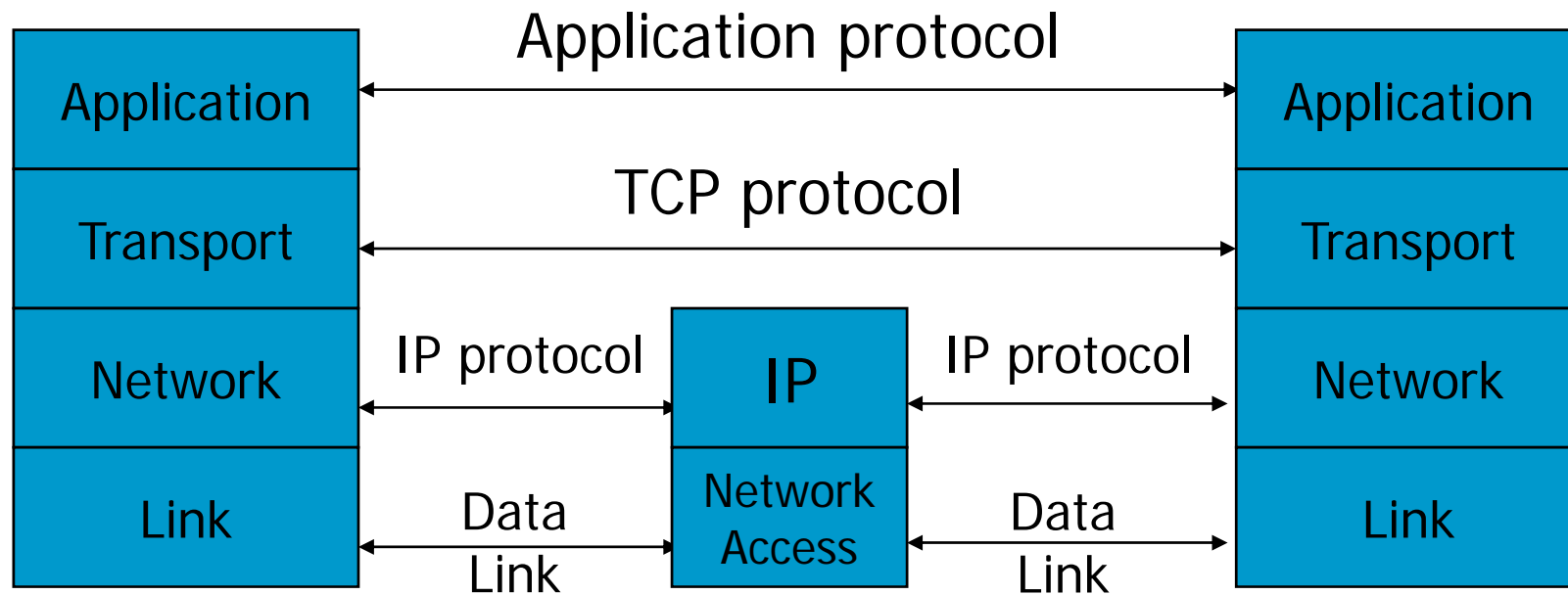
CS 526

Lecture 24



Network Security (1)

Network Protocols Stack



Protocols

Application	DNS, TFTP, TLS/SSL, FTP, Gopher, HTTP, IMAP, IRC, NNTP, POP3, SIP, SMTP, SNMP, SSH, TELNET, ECHO, BitTorrent, RTP, PNRP, rlogin, ENRP
	BGP
Transport	TCP, UDP, DCCP, SCTP, IL, RUDP
Internet	OSPF, ICMP and IGMP
	IP (IPv4, IPv6)
	ARP and RARP
Network access	Ethernet, Wi-Fi, token ring, PPP, SLIP, FDDI, ATM, Frame Relay, SMDS

Types of Addresses in Internet

- MAC addresses in the network access layer
 - 48 bits or 64 bits
- IP addresses for the network layer
 - 32 bits for IPv4, and 128 bits for IPv8
 - E.g., 128.3.23.3
- IP addresses + ports for the transport layer
 - E.g., 128.3.23.3:80
- Domain names for the application/human layer
 - E.g., www.purdue.edu

Routing and Translation of Addresses

- Translation between IP addresses and MAC addresses
 - Address Resolution Protocol (ARP) for IPv4
 - Neighbor Discovery Protocol (NDP) for IPv6
- Routing with IP addresses
 - TCP, UDP, IP for routing packets, connections
 - Border Gateway Protocol for routing table updates
- Translation between IP addresses and domain names
 - Domain Name System (DNS)

Threats in Networking

- Confidentiality
 - Packet sniffing
- Integrity
 - Session hijacking
- Availability
 - Denial of service attacks
- Common
 - Address translation poisoning attacks
 - Routing attacks

Concrete Security Problems

- ARP is not authenticated
 - APR spoofing (or ARP poisoning)
- Network packets pass by untrusted hosts
 - Packet sniffing
- TCP state can be easy to guess
 - TCP spoofing attack
- DNS is not authenticated
 - DNS poisoning attacks

Address Resolution Protocol (ARP)

- Primarily used to translate IP addresses to Ethernet MAC addresses
- Also used for IP over other LAN technologies, e.g., FDDI, or IEEE 802.11
- Each host maintains a table of IP to MAC addresses
- Message types:
 - ARP request
 - ARP reply
 - ARP announcement

ARP Spoofing (ARP Poisoning)

- Send fake or 'spoofed', ARP messages to an Ethernet LAN.
 - To have other machines associate IP addresses with the attacker's MAC
- Defenses
 - static ARP table
 - detection: Arpwatch, DHCP snooping
- Legitimate use
 - redirect a user to a registration page before allow usage of the network

IP Internet Protocol

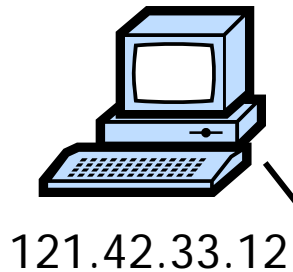
- Connectionless
 - Unreliable
 - Best effort
- Transfer datagram
 - Header
 - Data

Version	Header Length
Type of Service	
Total Length	
Identification	
Flags	Fragment Offset
Time to Live	
Protocol	
Header Checksum	
Source Address of Originating Host	
Destination Address of Target Host	
Options	
Padding	
IP Data	

IP Routing

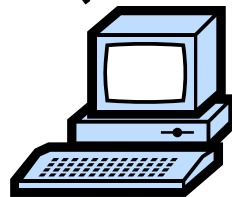


Meg



121.42.33.12

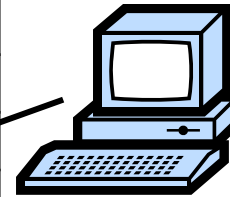
Packet	
Source	121.42.33.12
Destination	132.14.11.51
Sequence	5



ISP

121.42.33.1

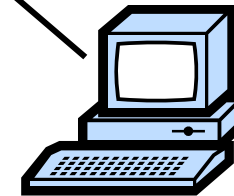
Office gateway



132.14.11.1



Tom



132.14.11.51

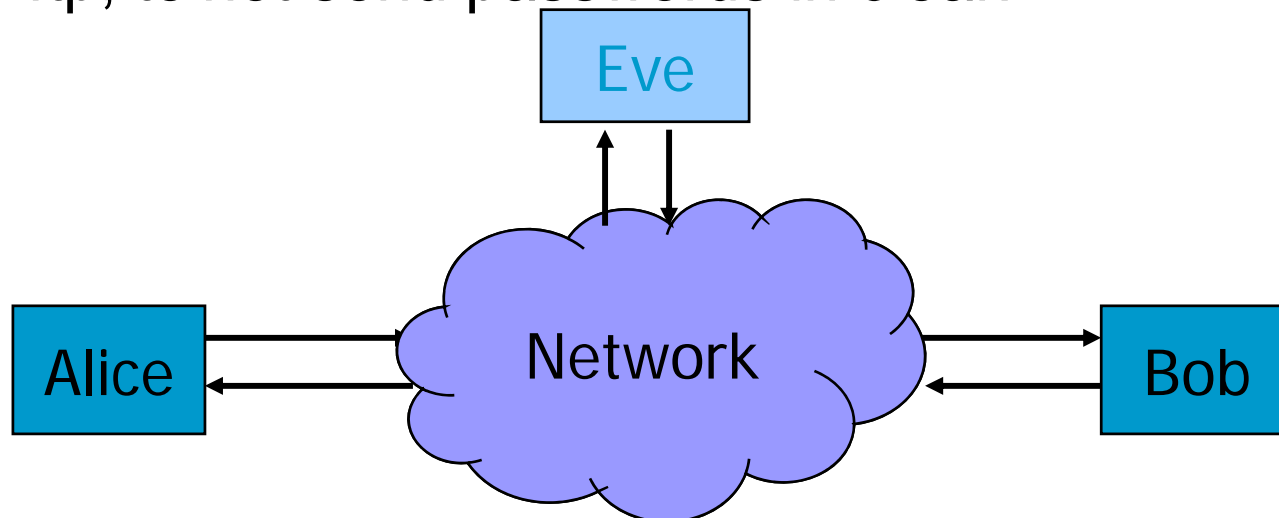
- Internet routing uses numeric IP address
- Typical route uses several hops

IP Protocol Functions (Summary)

- Routing
 - IP host knows location of router (gateway)
 - IP gateway must know routes to other networks
- Fragmentation and reassembly
 - If max-packet-size less than the user-data-size
- Error reporting
 - ICMP packet to source if packet is dropped

Packet Sniffing

- Promiscuous Network Interface Card reads all packets
 - Read all unencrypted data (e.g., “ngrep”)
 - ftp, telnet send passwords in clear!



Prevention: Encryption (IPSEC, TLS)

Tools for Network Sniffing

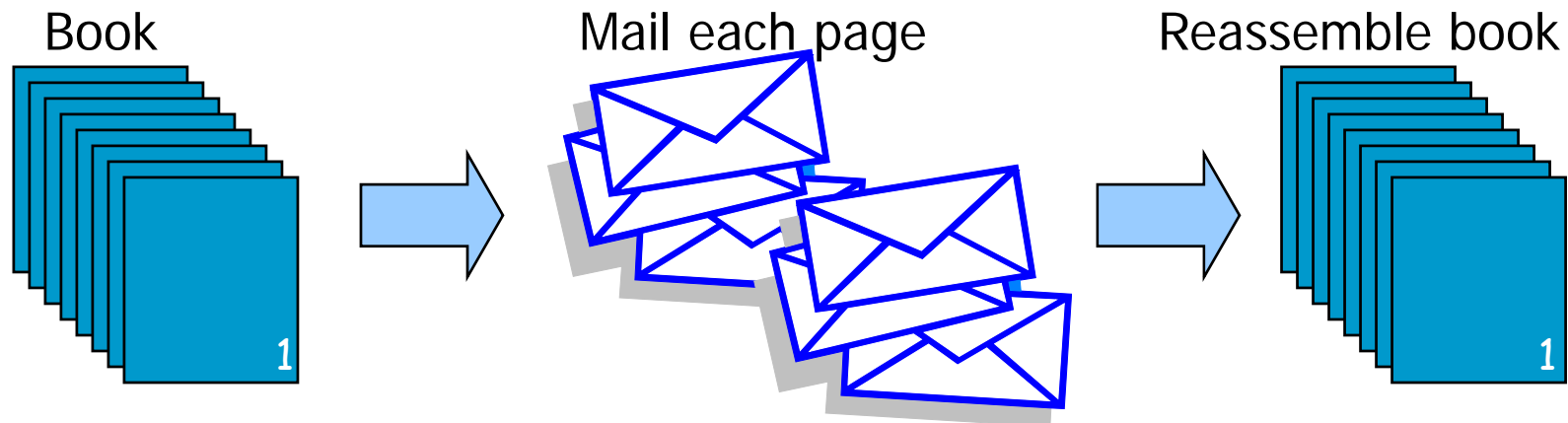
- tcpdump
- Windump
- Snort (network sniffer and network intrusion detection system)
- Wireshark (formerly Ethereal)
 - history of lot of buffer overflow vulnerabilities
- Sniffiy
- Dsniff

User Datagram Protocol

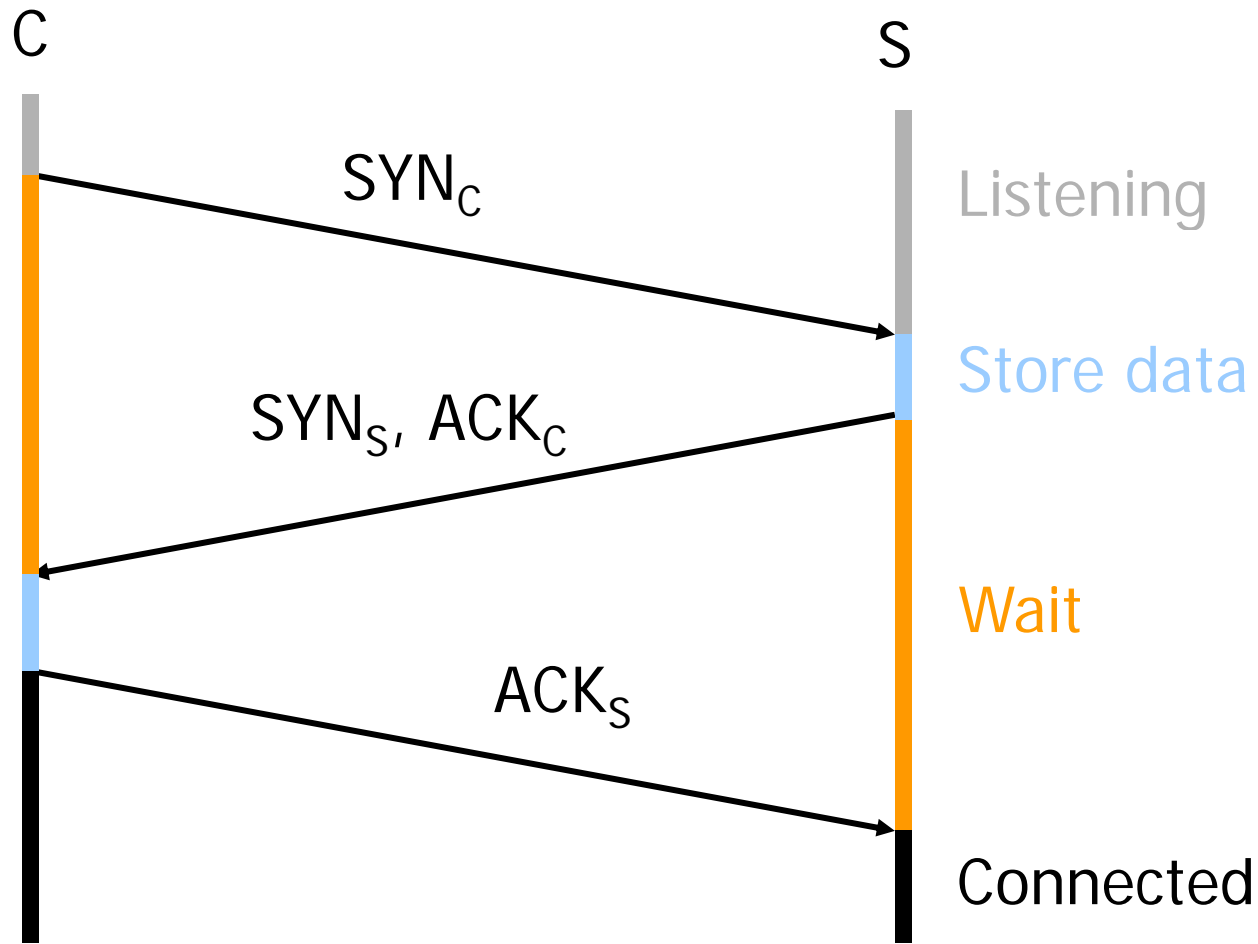
- IP provides routing
 - IP address gets datagram to a specific machine
- UDP separates traffic by port (16-bit number)
 - Destination port number gets UDP datagram to particular application process, e.g., 128.3.23.3:53
 - Source port number provides return address
- Minimal guarantees
 - No acknowledgment
 - No flow control
 - No message continuation

Transmission Control Protocol

- Connection-oriented, preserves order
 - Sender
 - Break data into packets
 - Attach packet numbers
 - Receiver
 - Acknowledge receipt; lost packets are resent
 - Reassemble packets in correct order



TCP Handshake



TCP Sequence Numbers

- Need high degree of unpredictability
 - If attacker knows initial seq # and amount of traffic sent, can estimate likely current values
 - Send a flood of packets with likely seq numbers
 - Attacker can inject packets into existing connection
- Some implementations are vulnerable

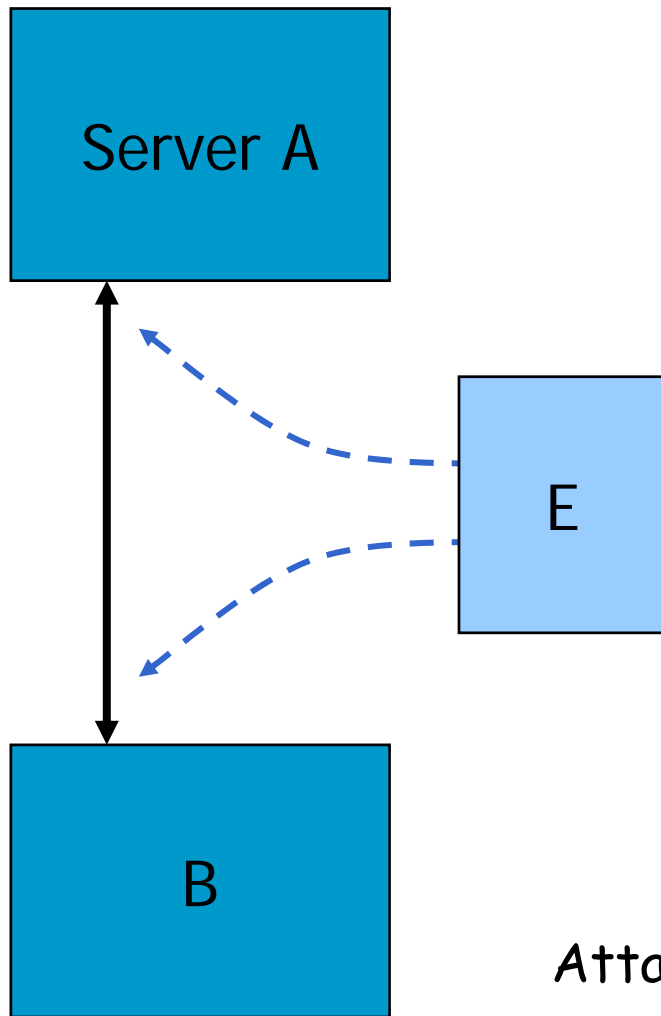
TCP Session Hijacking

- Each TCP connection has an associated state
 - Client IP and port number; same for server
 - Sequence numbers for client, server flows
- Problem
 - Easy to guess state
 - Port numbers are standard
 - Sequence numbers often chosen in predictable way

Risks from Session Hijacking

- Inject data into an unencrypted server-to-server traffic, such as an e-mail exchange, DNS zone transfers, etc.
- Inject data into an unencrypted client-to-server traffic, such as ftp file downloads, http responses.
- IP addresses often used for preliminary checks on firewalls or at the service level.
- Hide origin of malicious attacks.
- Carry out MITM attacks on weak cryptographic protocols.
 - often result in warnings to users that get ignored
- Denial of service attacks, such as resetting the connection.

Blind TCP Session Hijacking



- A, B trusted connection
 - Send packets with predictable seq numbers
- E impersonates B to A
 - Opens connection to A to get initial seq number
 - DoS B's queue
 - Sends packets to A that resemble B's transmission
 - E cannot receive, but may execute commands on A

Attack can be blocked if E is outside firewall.

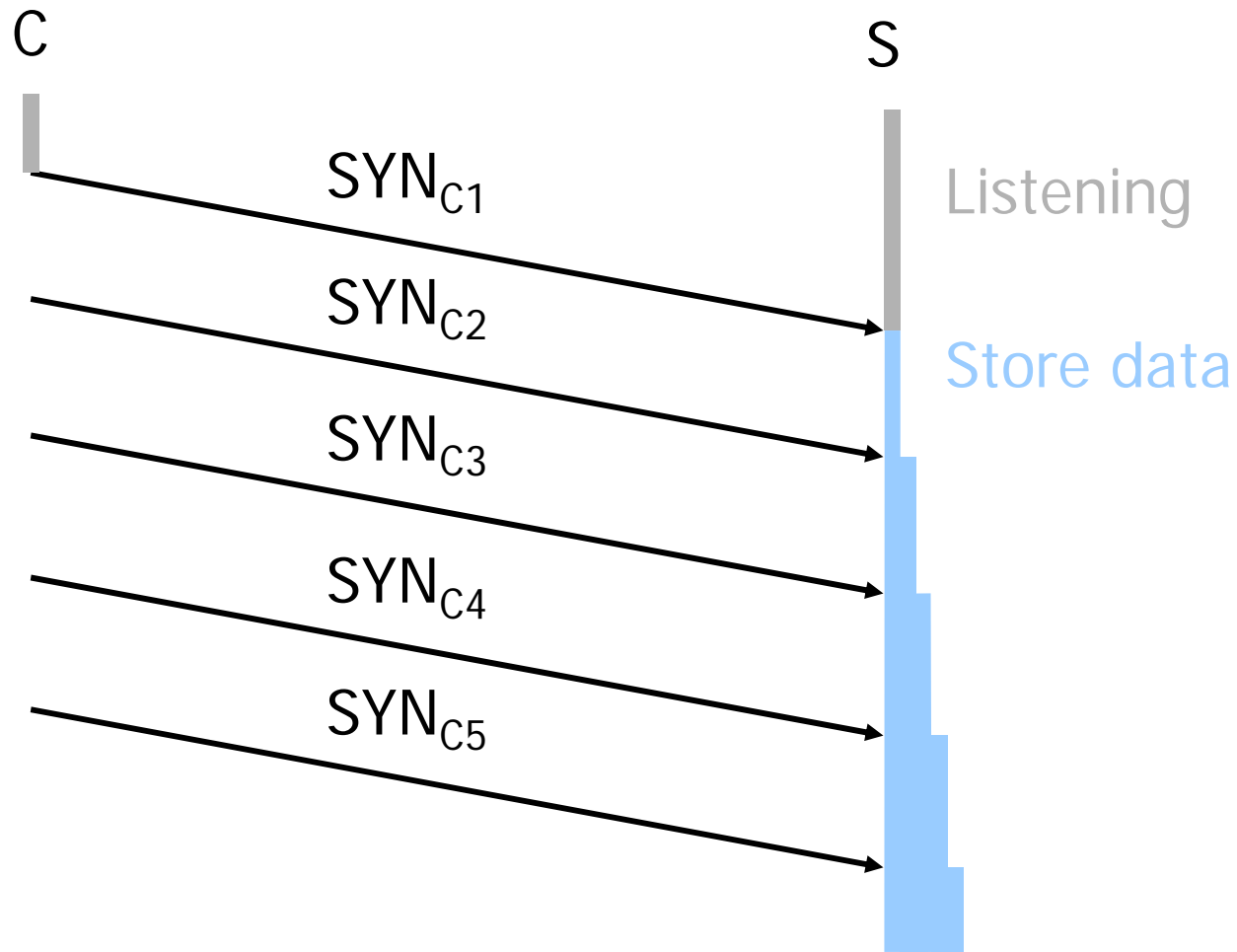
DoS vulnerability caused by session hijacking

- Suppose attacker can guess seq. number for an existing connection:
 - Attacker can send Reset packet to close connection. Results in DoS.
 - Naively, success prob. is $1/2^{32}$ (32-bit seq. #'s).
 - Most systems allow for a large window of acceptable seq. #'s
 - Much higher success probability.
- Attack is most effective against long lived connections, e.g. BGP.

Categories of Denial-of-service Attacks

	Stopping services	Exhausting resources
Locally	<ul style="list-style-type: none">• Process killing• Process crashing• System reconfiguration	<ul style="list-style-type: none">• Spawning processes to fill the process table• Filling up the whole file system• Saturate comm bandwidth
Remotely	<ul style="list-style-type: none">• Malformed packets to crash buggy services	<ul style="list-style-type: none">• Packet floods (Smurf, SYN flood, DDoS, etc)

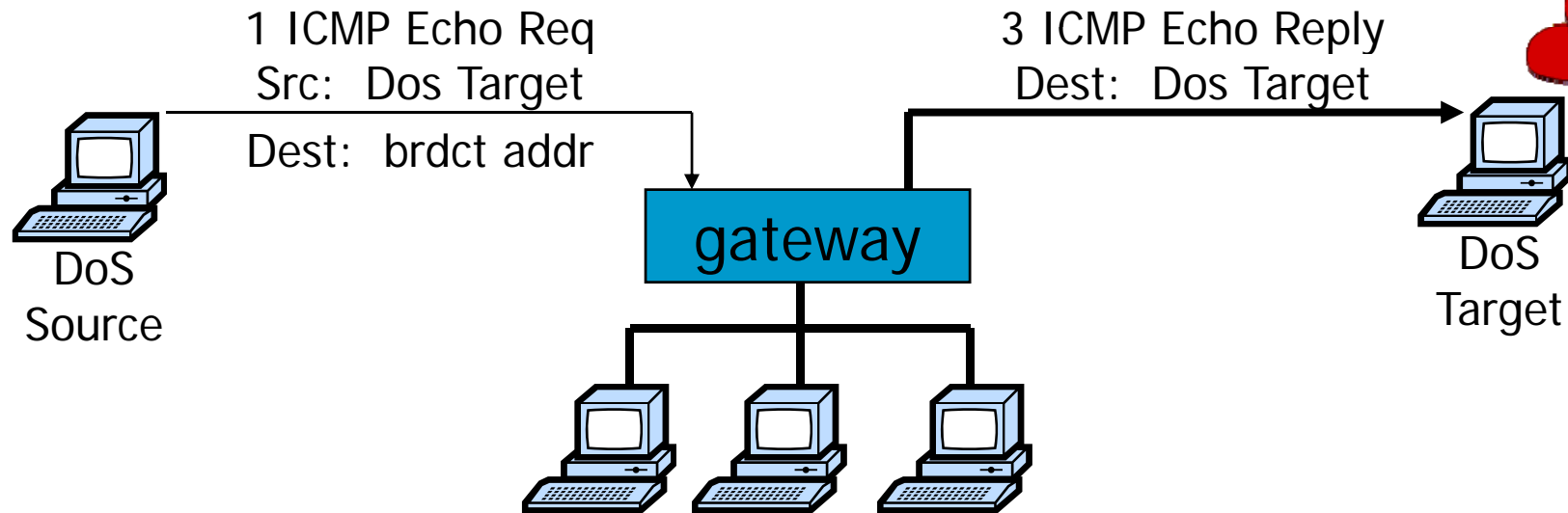
SYN Flooding



SYN Flooding

- Attacker sends many connection requests
 - Spoofed source addresses
- Victim allocates resources for each request
 - Connection requests exist until timeout
 - Old implementations have a small and fixed bound on half-open connections
- Resources exhausted \Rightarrow requests rejected
- No more effective than other channel capacity-based attack today

Smurf DoS Attack



- Send ping request to broadcast addr (ICMP Echo Req)
- Lots of responses:
 - Every host on target network generates a ping reply (ICMP Echo Reply) to victim
 - Ping reply stream can overload victim

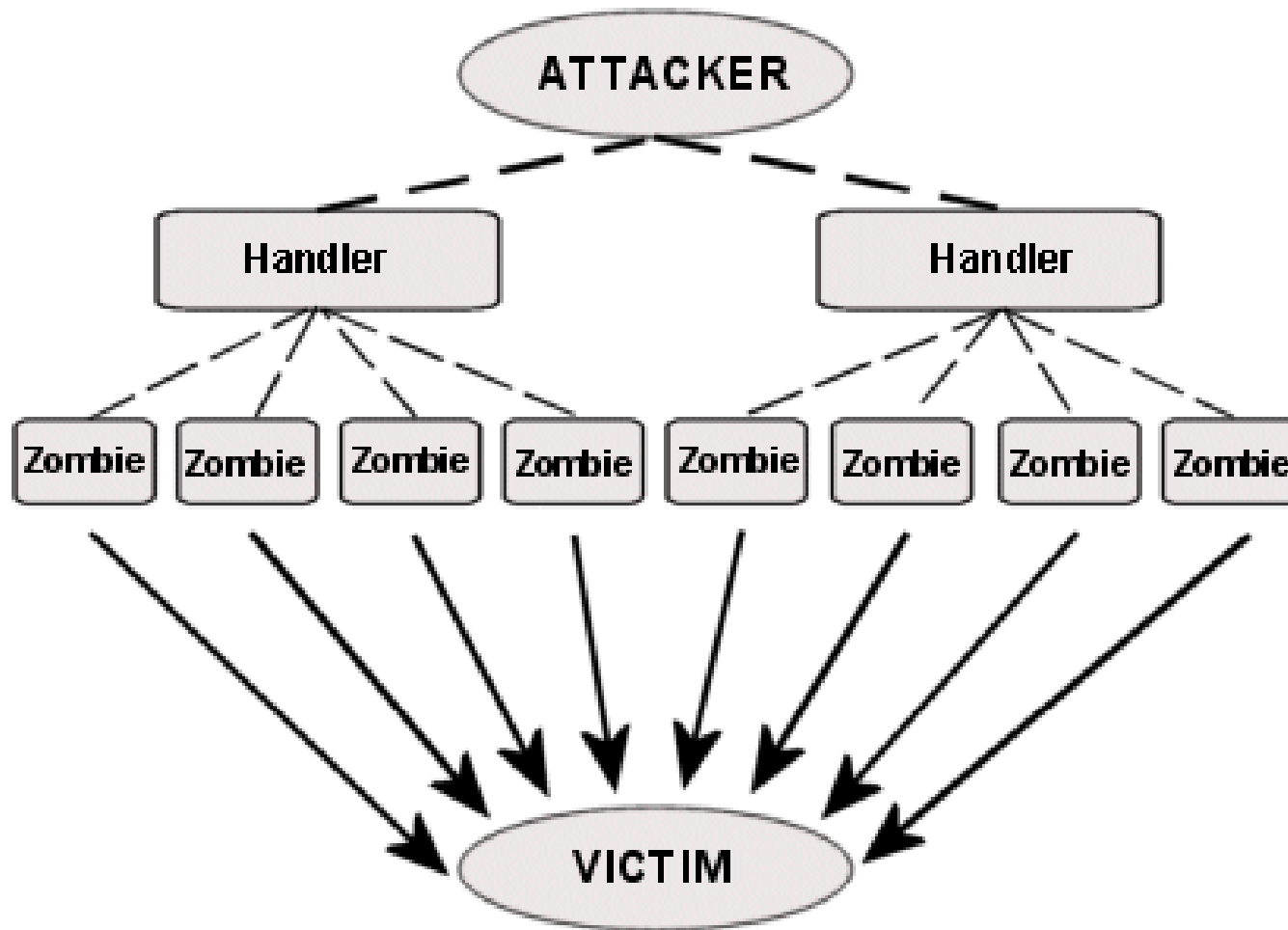
Prevention: reject external packets to broadcast address

Internet Control Message Protocol

- Provides feedback about network operation
 - Error reporting
 - Reachability testing
 - Congestion Control
- Example message types
 - Destination unreachable
 - Time-to-live exceeded
 - Parameter problem
 - Redirect to better gateway
 - Echo/echo reply - reachability test
 - Timestamp request/reply - measure transit delay

Distributed DoS (DDoS)

Architecture of a DDoS Attack



Hiding DDoS Attacks

- Reflection
 - Find big sites with lots of resources, send packets with spoofed source address, response to victim
 - PING => PING response
 - SYN => SYN-ACK
- Pulsing zombie floods
 - each zombie active briefly, then goes dormant;
 - zombies taking turns attacking
 - making tracing difficult

Cryptographic network protection

- Solutions above the transport layer
 - Examples: SSL and SSH
 - Protect against session hijacking and injected data
 - Do not protect against denial-of-service attacks caused by spoofed packets
- Solutions at network layer
 - Use cryptographically random ISNs [RFC 1948]
 - More generally: IPsec
 - Can protect against
 - session hijacking and injection of data
 - denial-of-service attacks using session resets

Coming Attractions ...

- DNS Security
- Network Security Tools

