CSE 484 / CSE M 584 (Autumn 2011)

Security and Networks

Daniel Halperin
Tadayoshi Kohno

Thanks to Dan Boneh, Dieter Gollmann, John Manferdelli, John Mitchell, Vitaly Shmatikov, Bennet Yee, and many others for sample slides and materials ...

Class updates

- (Short) Homework 3
  - Due Wednesday
  - Individual assignment
- My office hours this week:
  - **CSE 210:** M,W,F after class. T,Th afternoons
  - others by appointment
  - come pick up graded Homework #2

Lab 3

- **Posted on website and on Catalyst.**
    - https://catalyst.uw.edu/collectit/assignment/dhalperi/17513/72548
    - Hack my privacy!

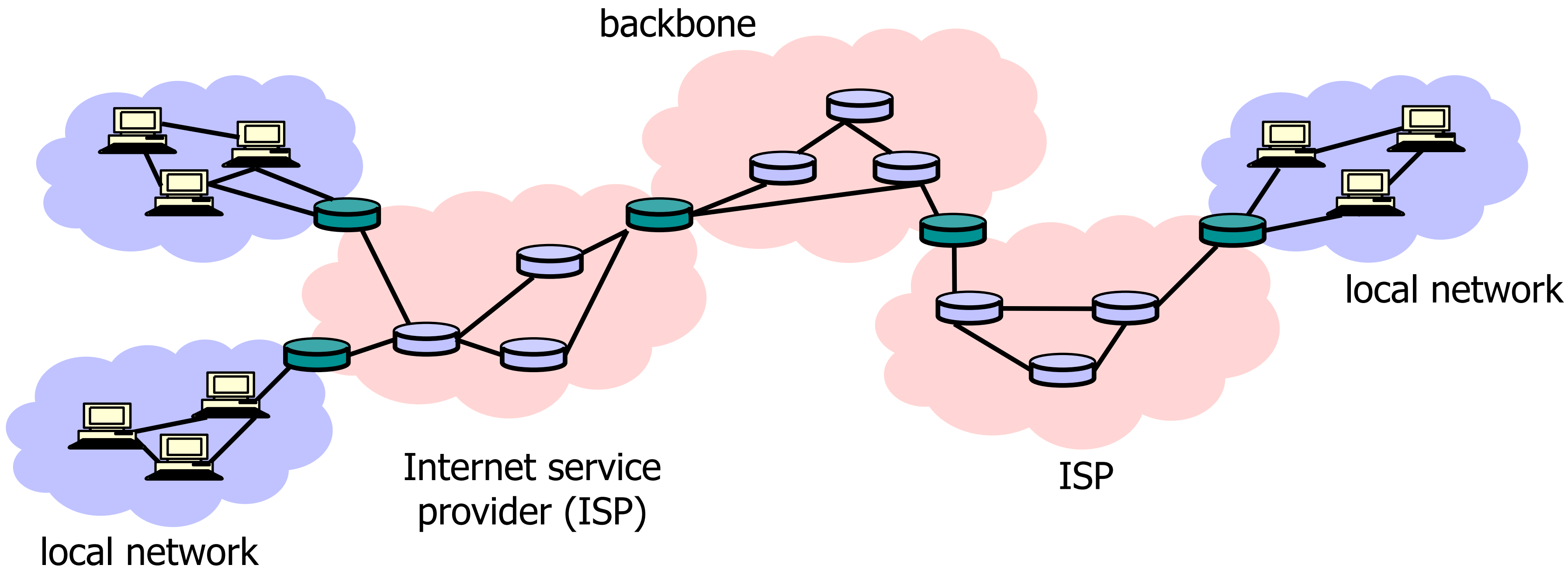
Lab 3

- **Posted on website and on Catalyst.**
  - https://catalyst.uw.edu/collectit/assignment/dhalperi/17513/72548
  - Hack my privacy!
- ***This lab is optional***
  - Can only help your grade.
  - Lots of opportunity for extra credit.
  - I really think this lab is fun, and encourage you to do it, but we're not going to require it.

This week

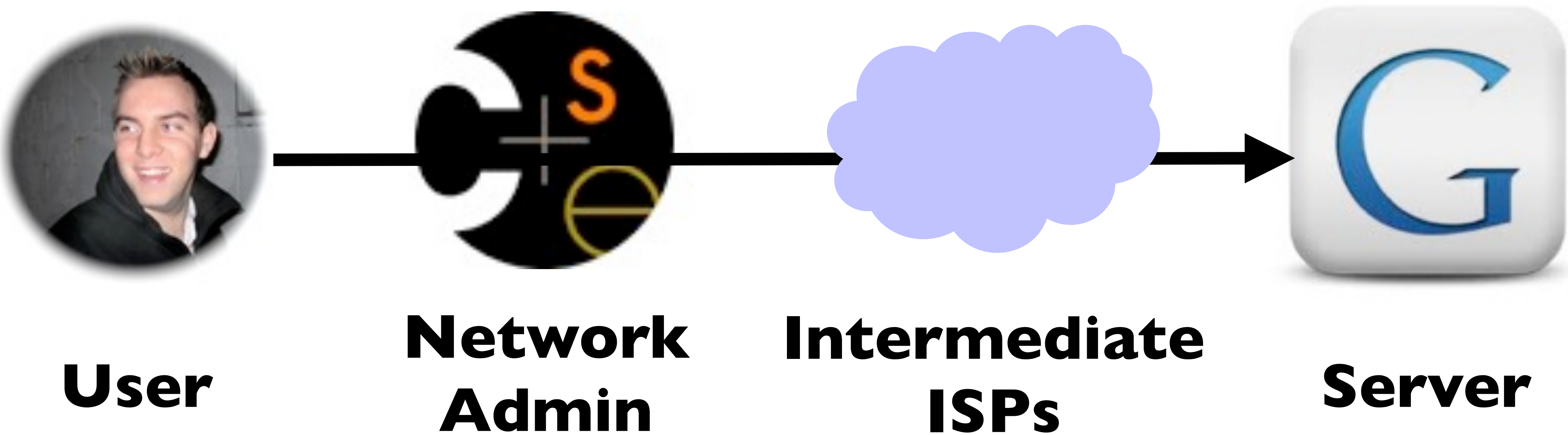
- **Today:** Network security
- **Wednesday:** Potpourri
- **Friday:** Any questions you have
  - Submit to my email, cse484-tas
  - Submit anonymously via the feedback form on the website

Internet Infrastructure



- TCP/IP for packet routing and connections
- Border Gateway Protocol (BGP) for route discovery
- Domain Name System (DNS) for IP address discovery

(Some) Entities



(Some) Goals

User

(Some) Goals

User

- Service (can get to Internet)
- Privacy (middle-entities shouldn't know what communicating or with whom)
- Fairness (e.g., get service I paid for)
- Integrity (can't impersonate me)
- Safety (network shouldn't attack me)

(Some) Goals

Network Admin

(Some) Goals

Network Admin

- Service (clients can get to Internet)
- Performance (network works well)
- Identity (know what's on network)
- Safety (no one launching attacks)
- Accountability (can find bad users)

(Some) Goals

Intermediate ISPs

(Some) Goals

Intermediate ISPs

- Service (deliver traffic -> earn $$)
- Reliability & Performance (network works well)
- Integrity of delivered traffic (can bill customers properly, you're not over-charged by providers)

(Some) Goals

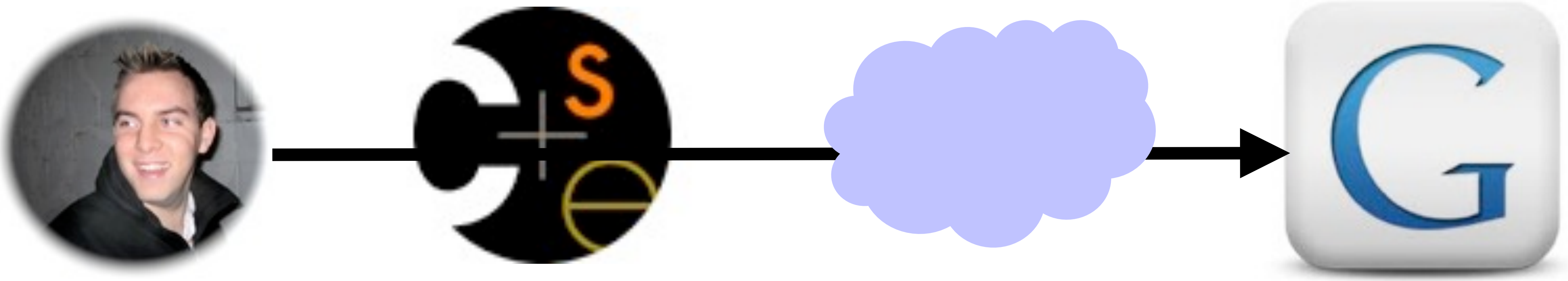
Server

(Some) Goals

Server

- Service (deliver traffic -> earn $$)
- Reliability & Performance (network works well)
- Analytics (better delivery)
- Accounting (can bill customers properly)
- Safety (not being attacked)

(Some) Malicious Goals



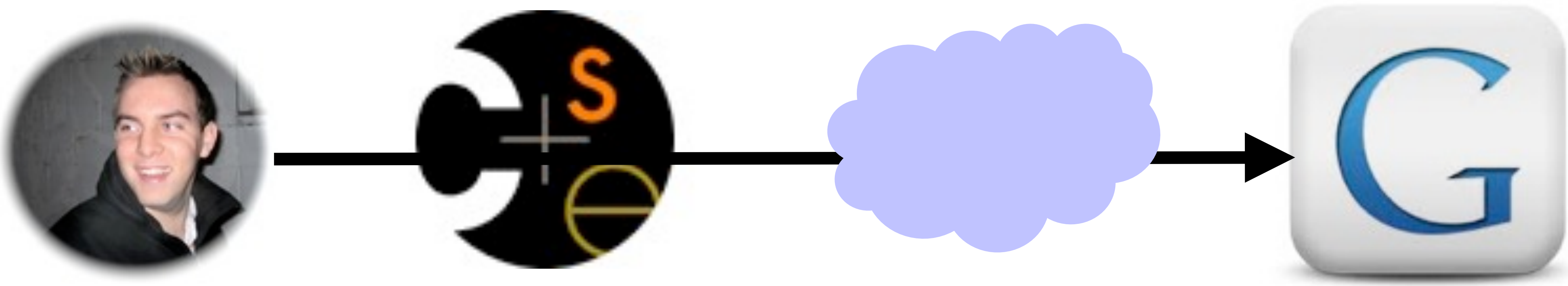
User

Network Admin

Intermediate ISPs

Server

(Some) Malicious Goals

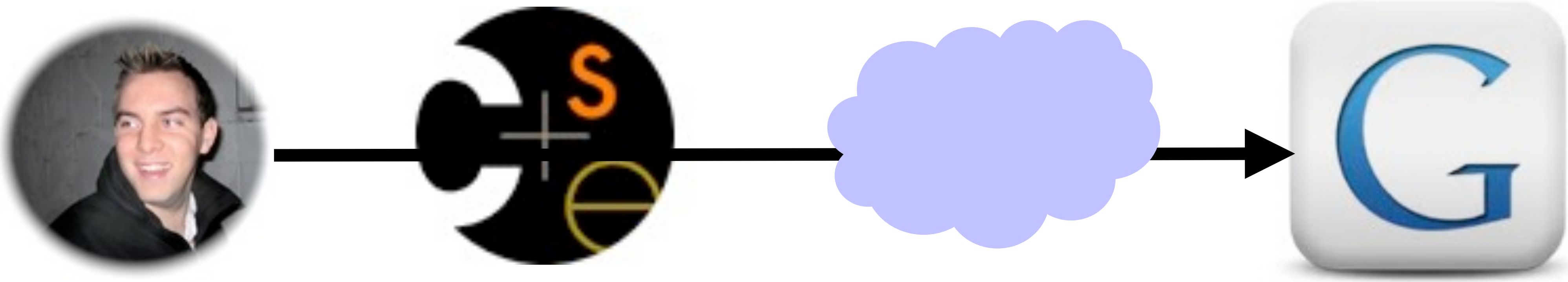


User | **Network Admin** | **Intermediate ISPs** | **Server**

Launch undetectable attacks

Probe for vulnerabilities

(Some) Malicious Goals



User | **Network Admin** | **Intermediate ISPs** | **Server**

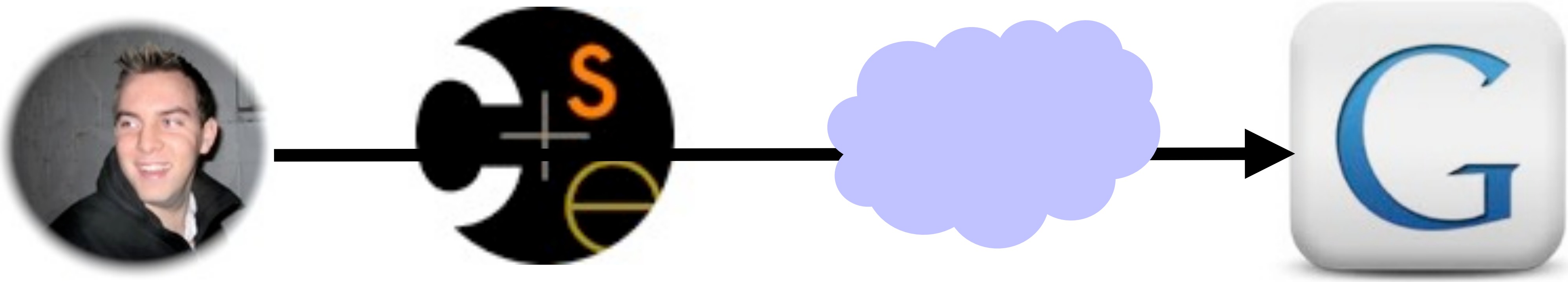
Launch undetectable attacks

Probe for vulnerabilities

Spy on/tamper with traffic

Impersonate servers

(Some) Malicious Goals



User

Launch undetectable attacks

Probe for vulnerabilities

Network Admin / **Intermediate ISPs**

Spy on/tamper with traffic

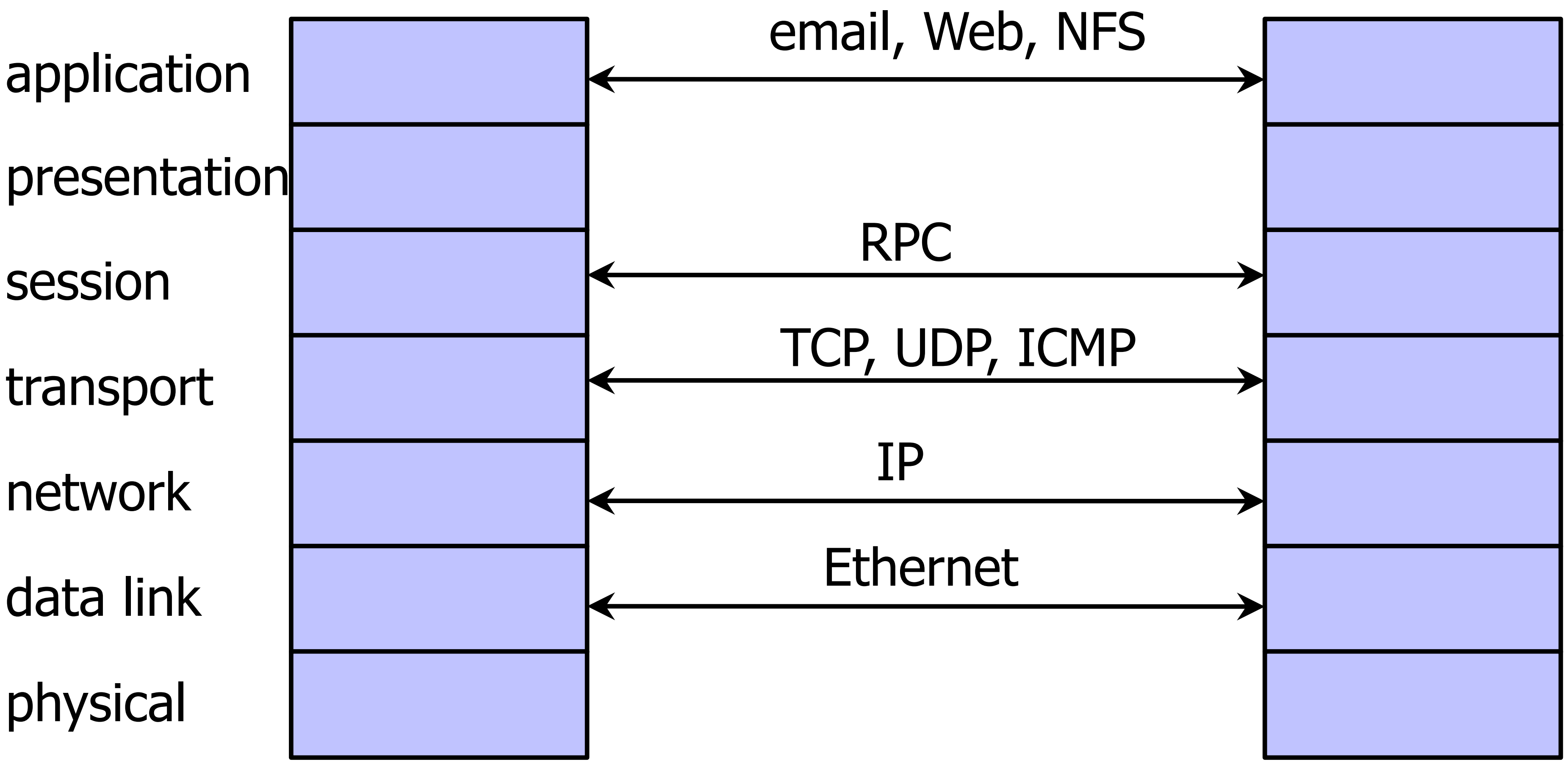
Impersonate servers

Server

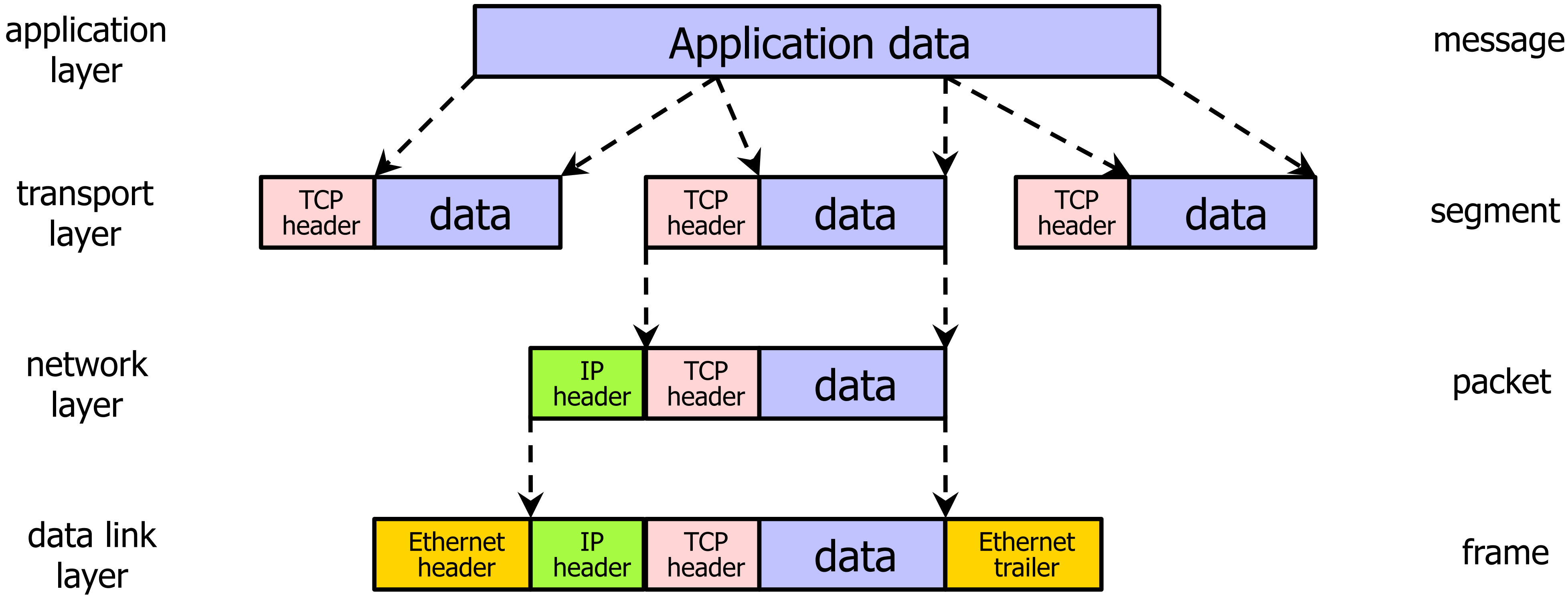
Spy on users

Identify anonymous users

OSI Protocol Stack

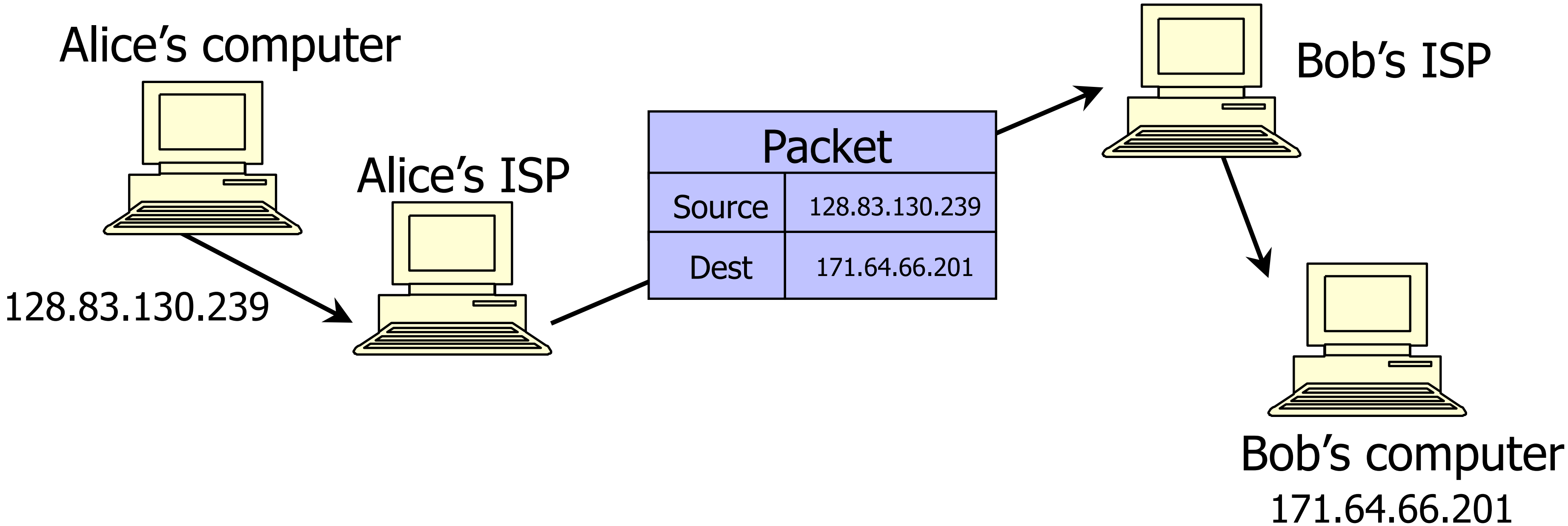


Data Formats



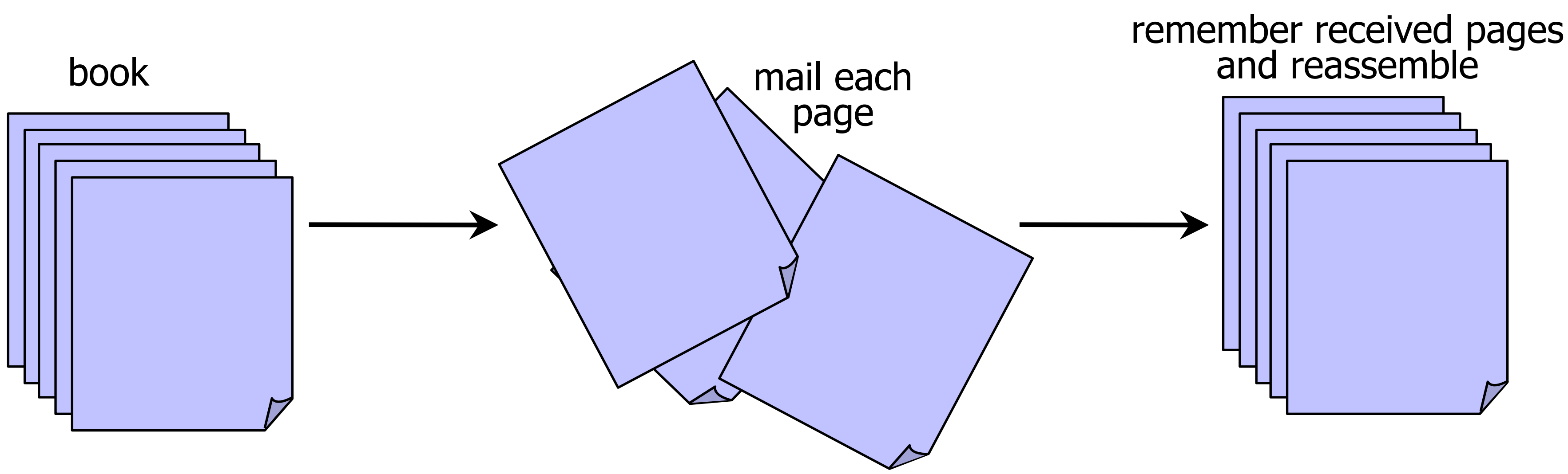
IP (Internet Protocol)

- Connectionless
  - Unreliable, "best-effort" protocol
- Uses numeric addresses for routing
  - Typically several hops in the route



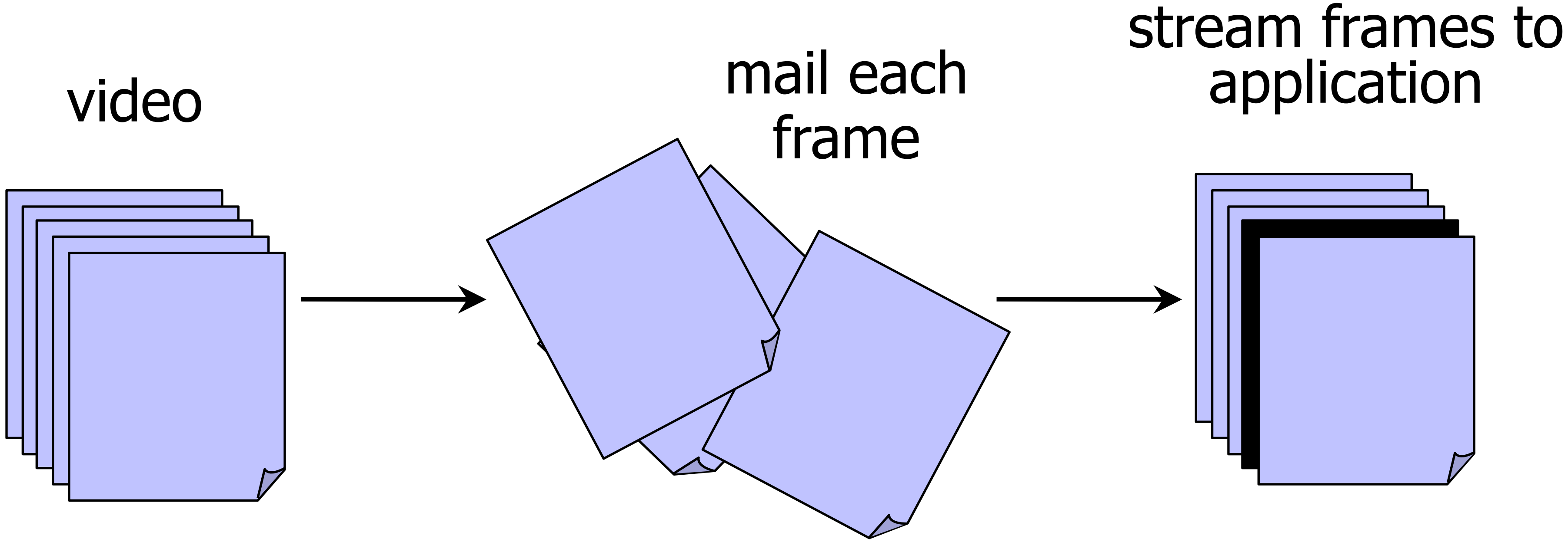
TCP (Transmission Control Protocol)

- Sender: break data into packets
  - Sequence number is attached to every packet
- Receiver: reassemble packets in correct order
  - Acknowledge receipt; lost packets are re-sent
- Connection state maintained on both sides



UDP (User Datagram Protocol)

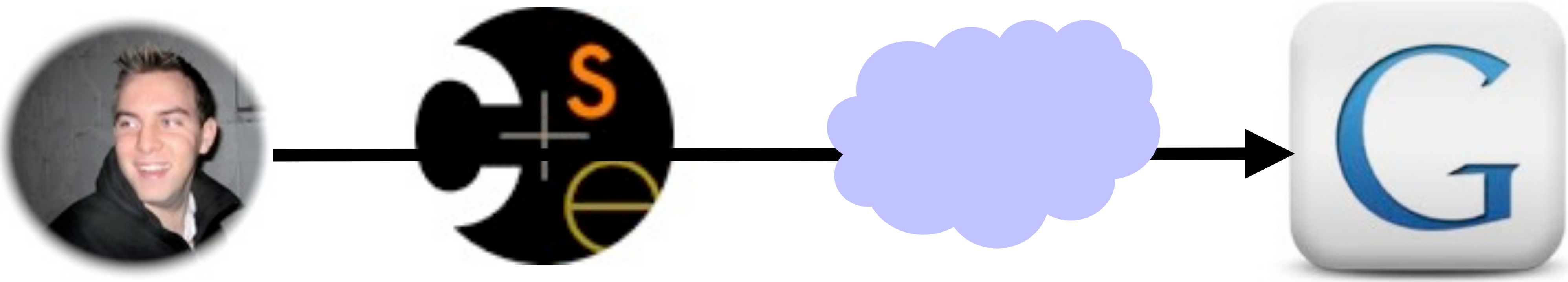
- Sender: break data into packets
  - Sequence number - maybe? If Application wants them
- Receiver: receive packets
  - No acknowledgement
  - Dropped packets are skipped - no retransmission



ICMP (Control Message Protocol)

- Provides feedback about network operation
  - "Out-of-band" messages carried in IP packets
  - Error reporting, congestion control, reachability, etc.
- Example messages:
  - Destination unreachable
  - Time exceeded
  - Parameter problem
  - Redirect to better gateway
  - Reachability test (echo / echo reply)
  - Message transit delay (timestamp request / reply)

(Some) Malicious Goals



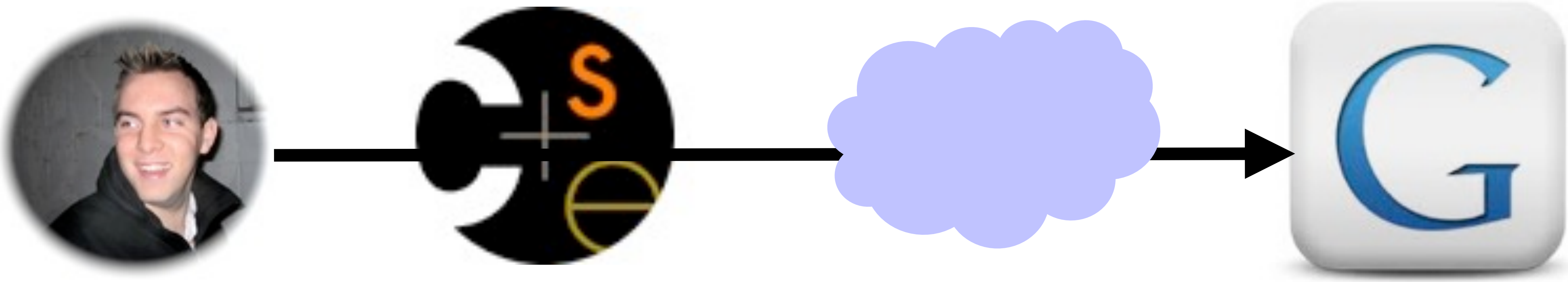
User

Network Admin

Intermediate ISPs

Server

(Some) Malicious Goals

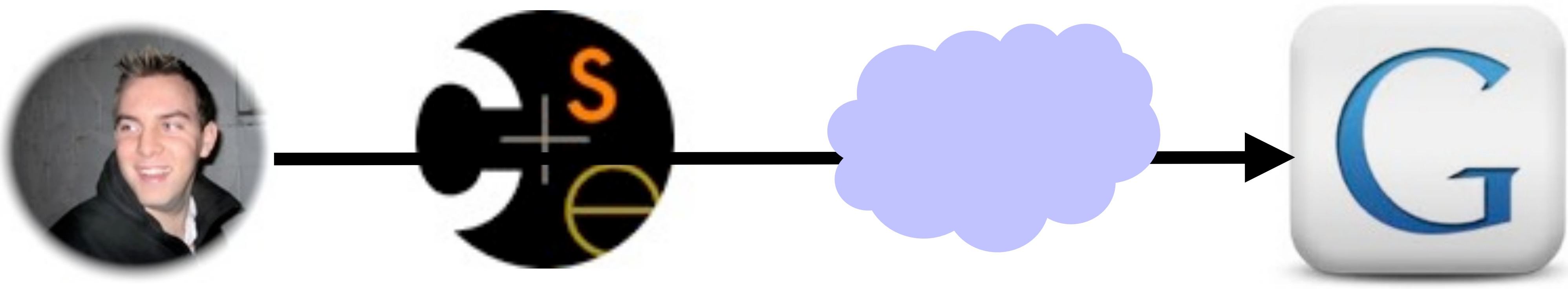


User | **Network Admin** | **Intermediate ISPs** | **Server**

Launch undetectable attacks

Probe for vulnerabilities

(Some) Malicious Goals



User | **Network Admin** | **Intermediate ISPs** | **Server**

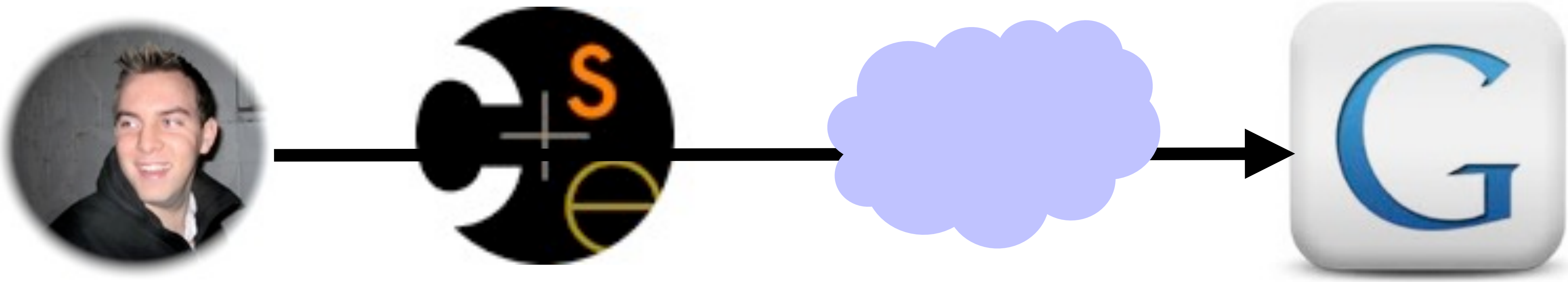
Launch undetectable attacks

Probe for vulnerabilities

Spy on/tamper with traffic

Impersonate servers

(Some) Malicious Goals



User	**Network Admin**	**Intermediate ISPs**	**Server**

Launch undetectable attacks

Probe for vulnerabilities

Spy on/tamper with traffic

Impersonate servers

Identify anonymous users

Detecting attacks

User

Launch undetectable attacks

Detecting attacks

User

Launch undetectable attacks

- **Problem:** IP packets contain source IP address

Detecting attacks

User

Launch undetectable attacks

- **Problem:** IP packets contain source IP address

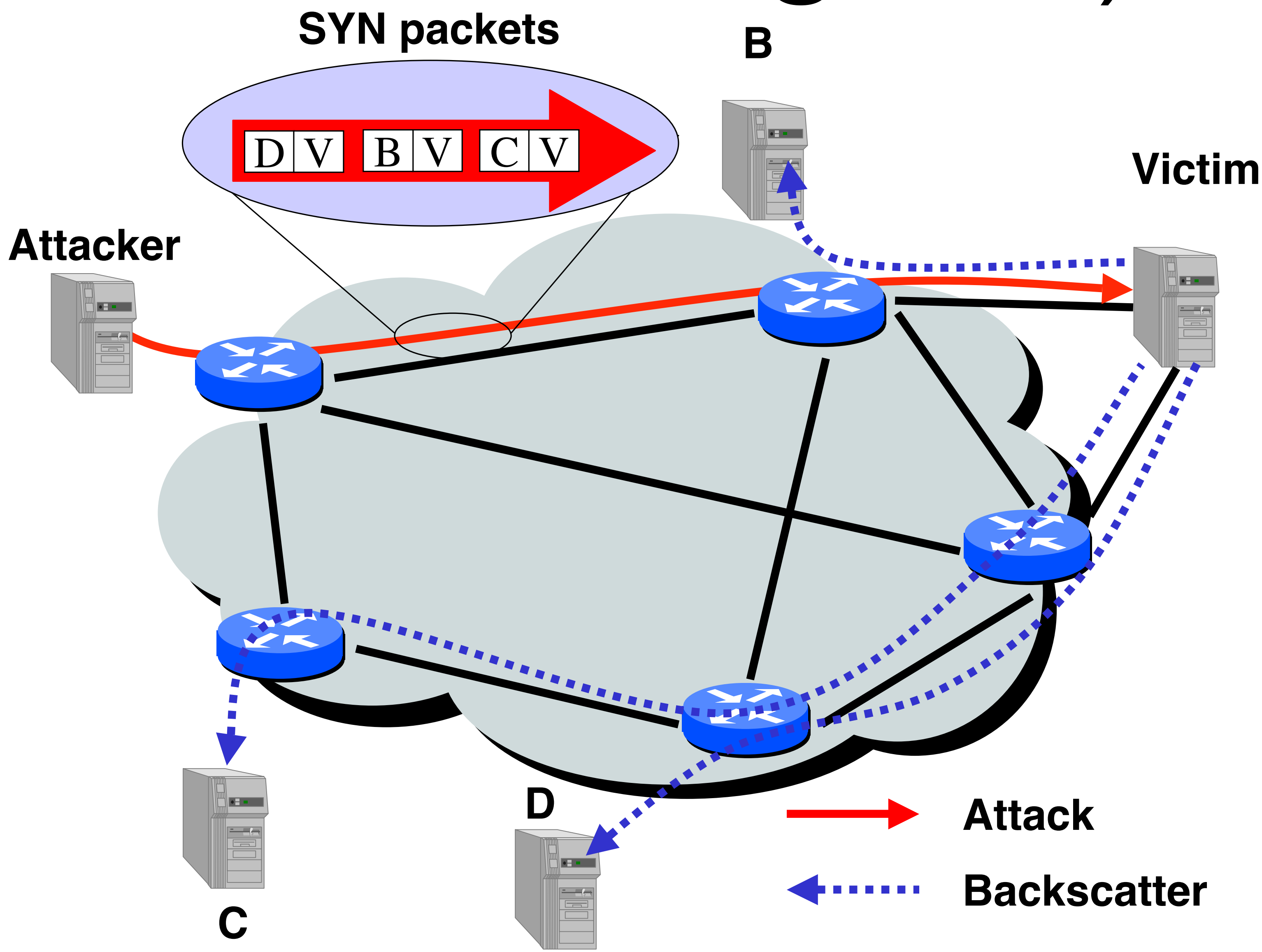
Detecting attacks

User

Launch undetectable attacks

- **Problem:** IP packets contain source IP address
- **Solution:** Spoof IP address

Inferring DDOS (Moore, Voelker, Savage '01)



Finding vulnerabilities

User

Probe for vulnerabilities

Finding vulnerabilities

User

Probe for vulnerabilities

- **Many, many tools**

Finding vulnerabilities

User

Probe for vulnerabilities

- **Many, many tools**
- One example: **Nmap**
  - Many services have known TCP/UDP ports
  - These give away what services you're running

Nmap example (me)

dhalperi@dhm cse484 % **nmap dsp.cs.washington.edu**

Starting Nmap 5.51 (http://nmap.org) at 2011-12-05 14:05 PST
Nmap scan report for dsp.cs.washington.edu (128.208.4.246)
Host is up (0.0062s latency).
Not shown: 996 closed ports
PORT STATE SERVICE
22/tcp open ssh
139/tcp open netbios-ssn
443/tcp open https
445/tcp open microsoft-ds

Nmap done: 1 IP address (1 host up) scanned in 1.36 seconds

Nmap example (aqua)

dhalperi@dhm cse484 % nmap aqua.cs.washington.edu

Starting Nmap 5.51 (http://nmap.org) at 2011-12-05 14:06 PST
Nmap scan report for aqua.cs.washington.edu (128.208.4.187)
Host is up (0.0022s latency).
Not shown: 990 filtered ports
PORT STATE SERVICE
80/tcp open http
135/tcp open msrpc
139/tcp open netbios-ssn
445/tcp open microsoft-ds
1025/tcp open NFS-or-IIS
1026/tcp open LSA-or-nterm
1027/tcp open IIS
1028/tcp open unknown
1048/tcp open neod2
3389/tcp open ms-term-serv

Nmap done: 1 IP address (1 host up) scanned in 5.29 seconds

telnet example

Fingerprinting users

Server

Identify anonymous users

Fingerprinting users

Server

Identify anonymous users

- **Browser**

Fingerprinting users

Server

Identify anonymous users

- **Browser**
- **Clocks**

Fingerprinting users

Server

Identify anonymous users

- **Browser**
- **Clocks**
- More

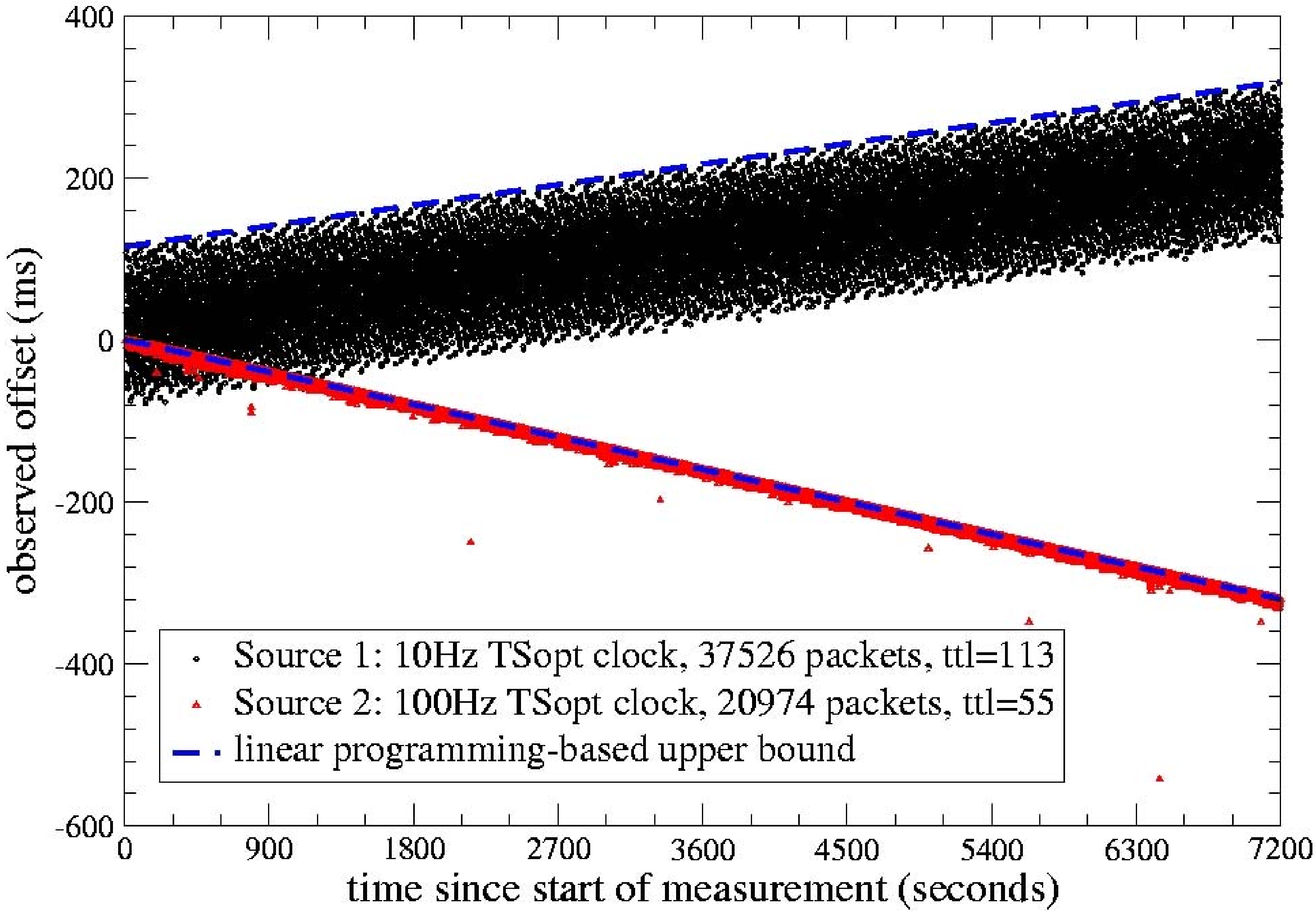
Browser example

http://panopticlick.eff.org/

Clocks

Kohno, et al. 2004

Clocks

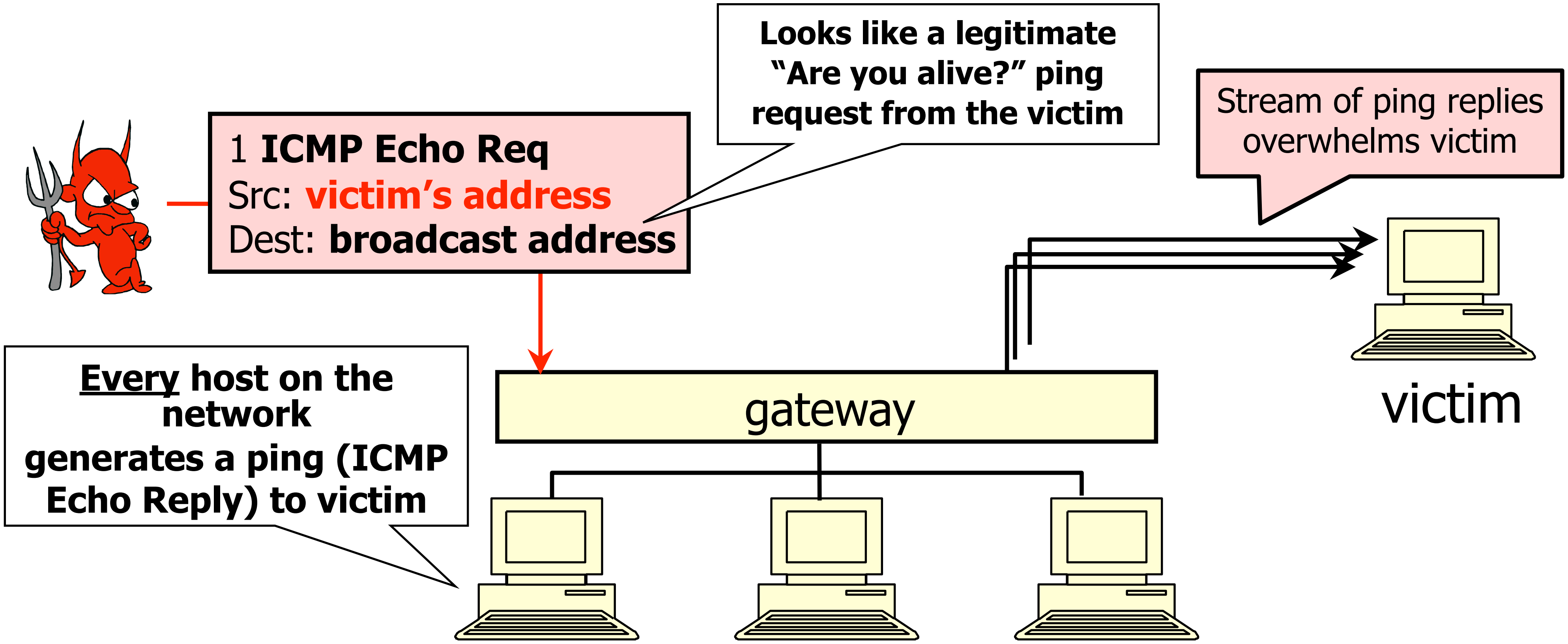


Kohno, et al. 2004

Security Issues in TCP/UDP

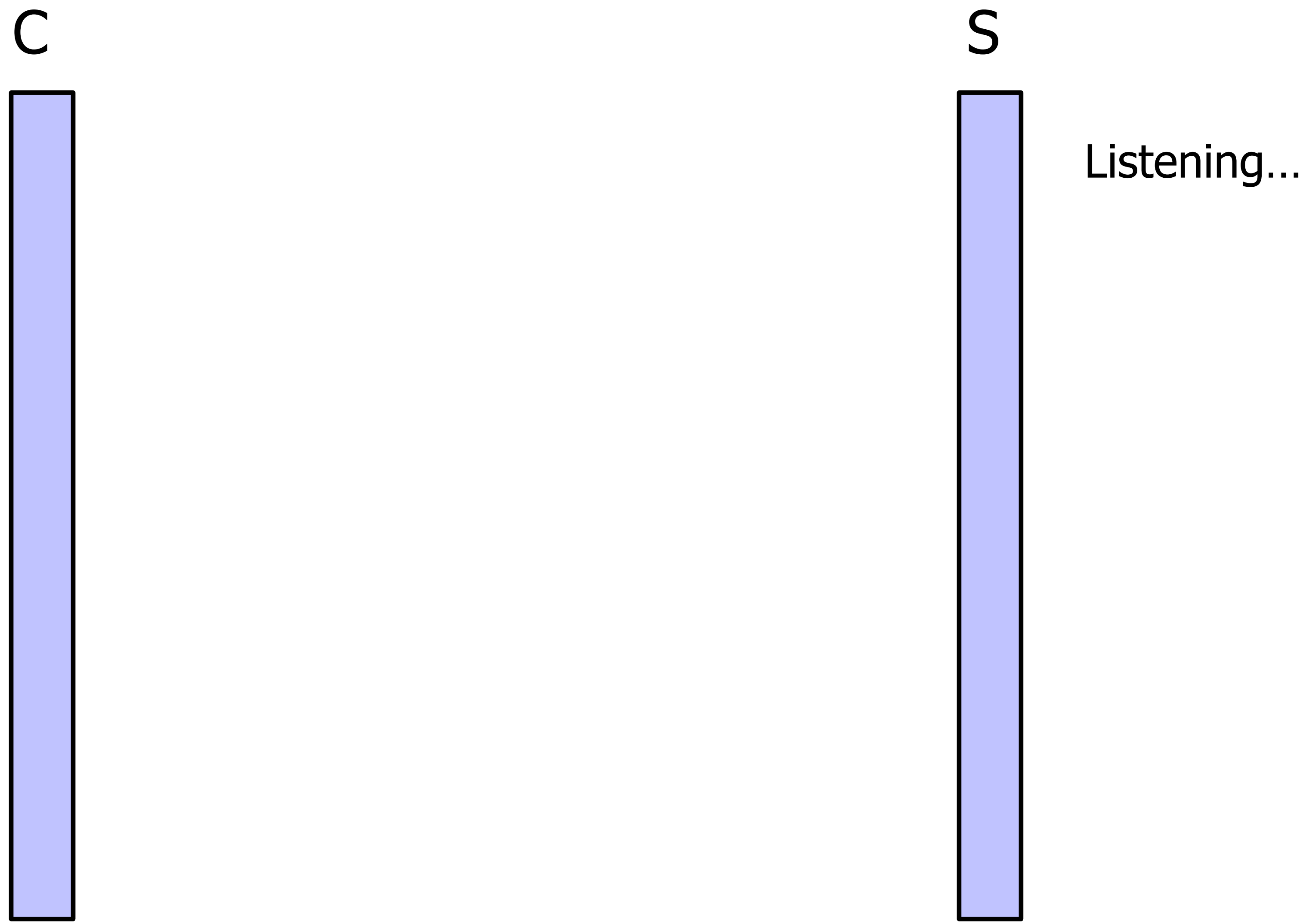
- Network packets pass through/by untrusted hosts
  - Eavesdropping (packet sniffing)
  - Modifications
- IP addresses are public
  - Smurf attacks
  - Anonymity?
- TCP connection requires state
  - SYN flooding
- TCP state is easy to guess
  - TCP spoofing and connection hijacking

Smurf Attack

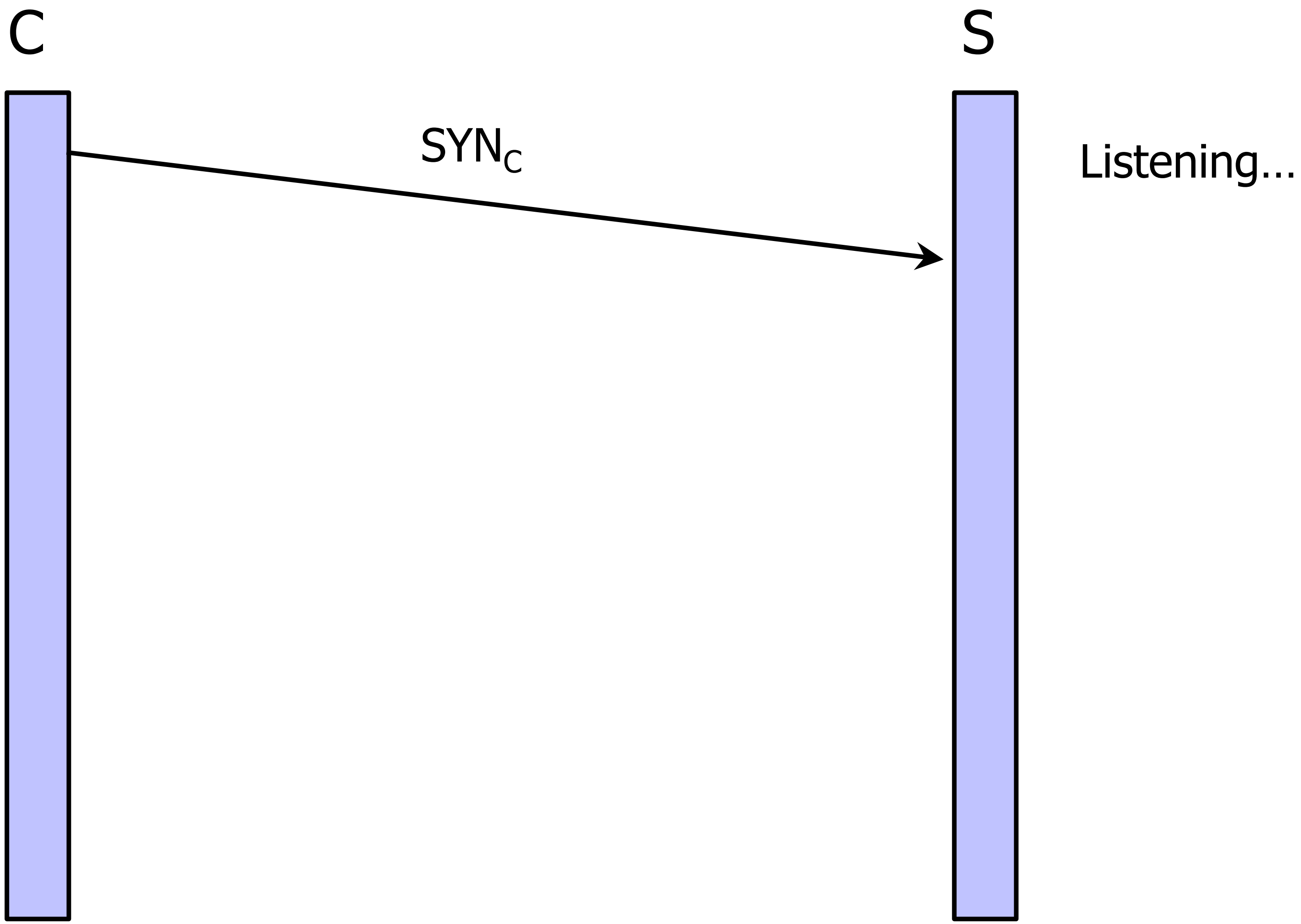


Solution: reject external packets to broadcast addresses

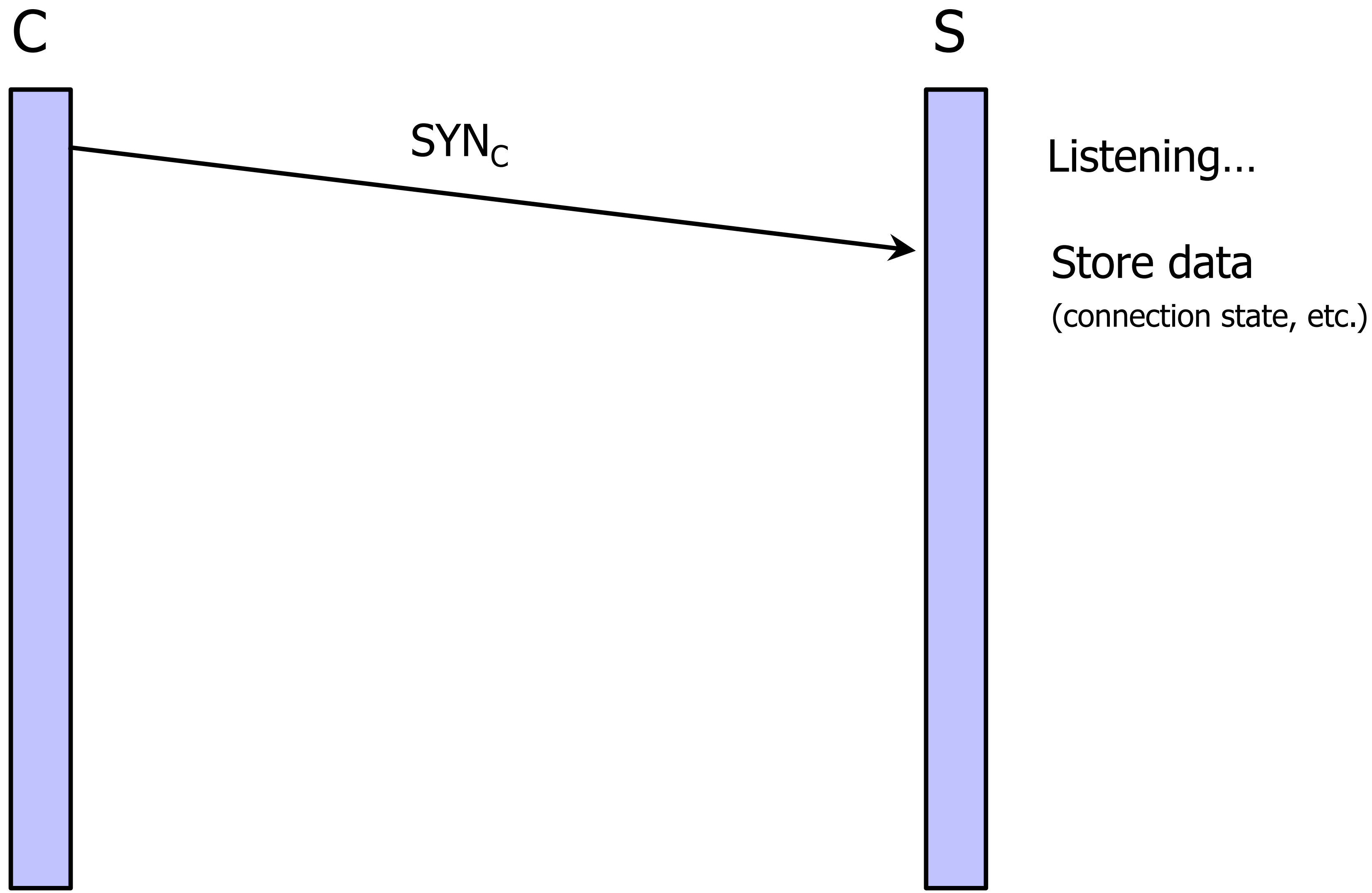
TCP Handshake



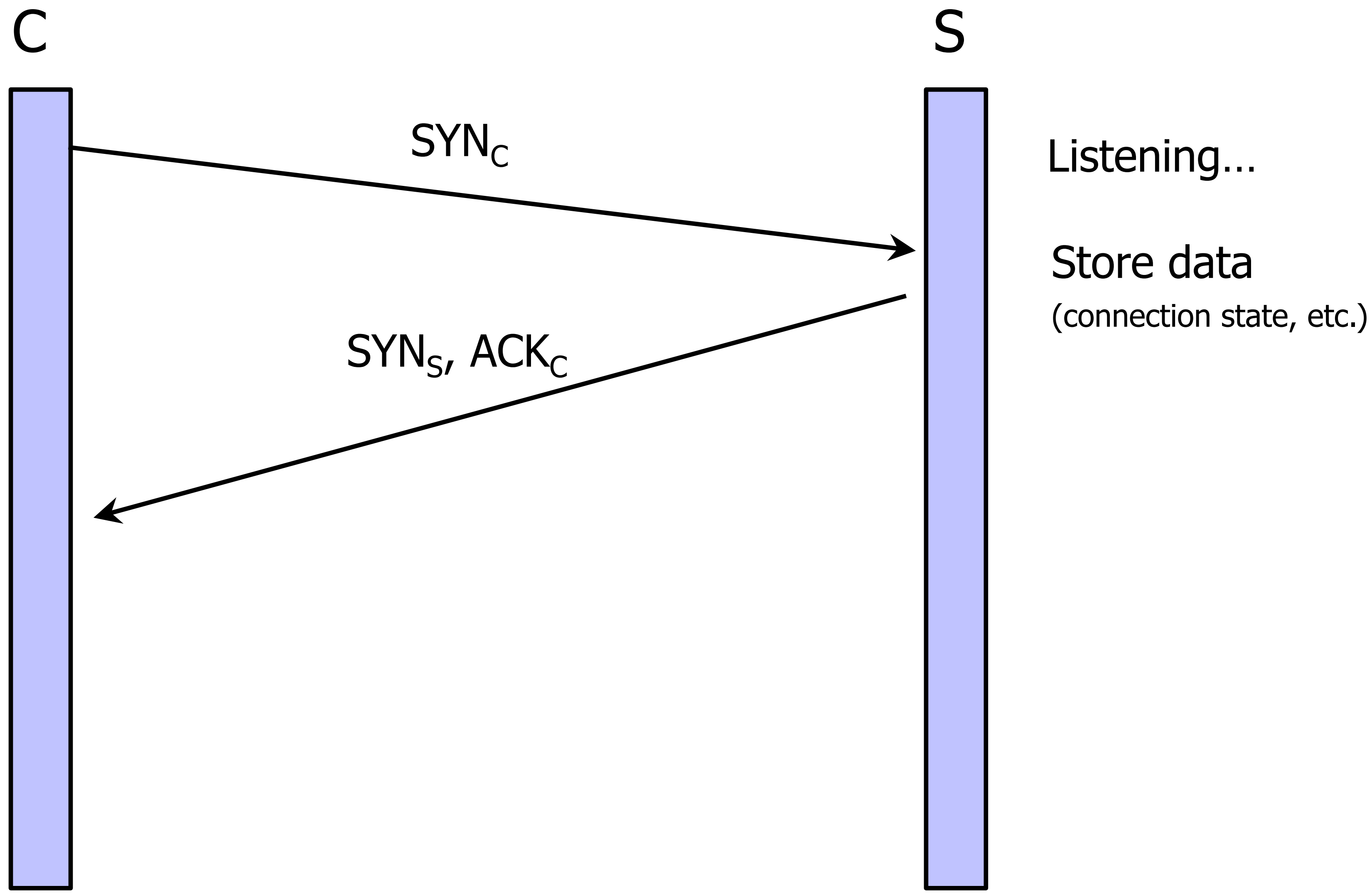
TCP Handshake



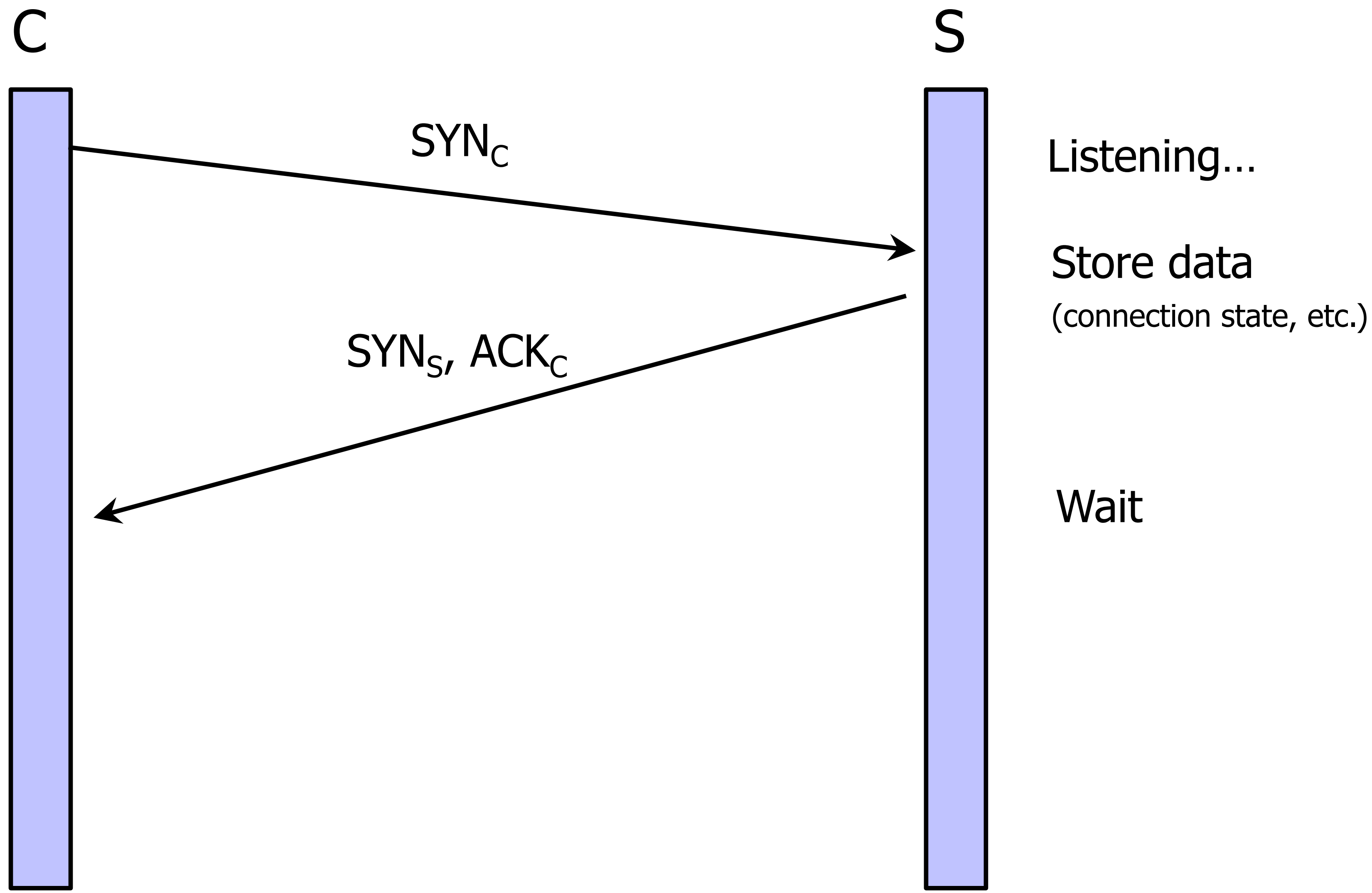
TCP Handshake



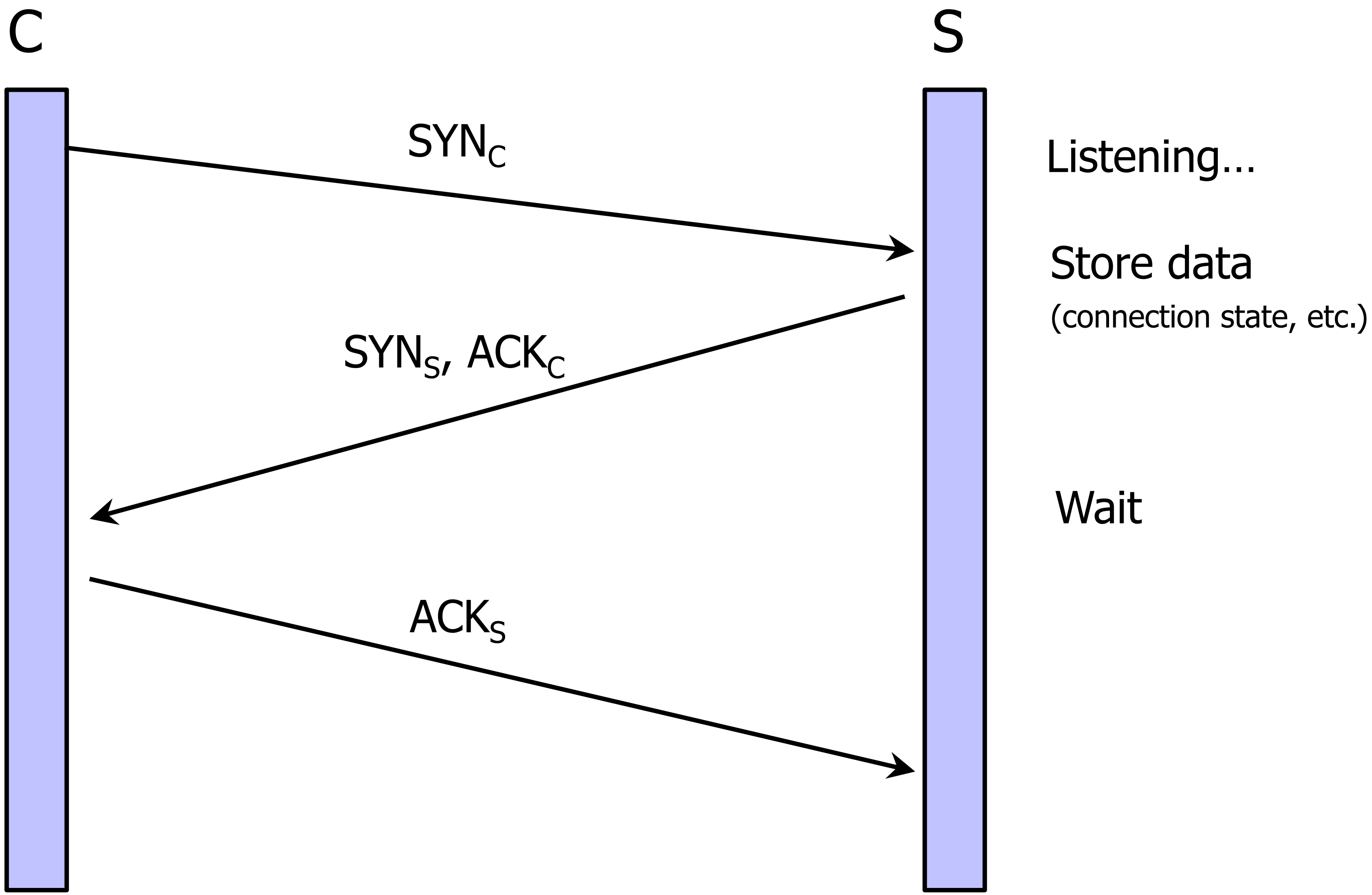
TCP Handshake



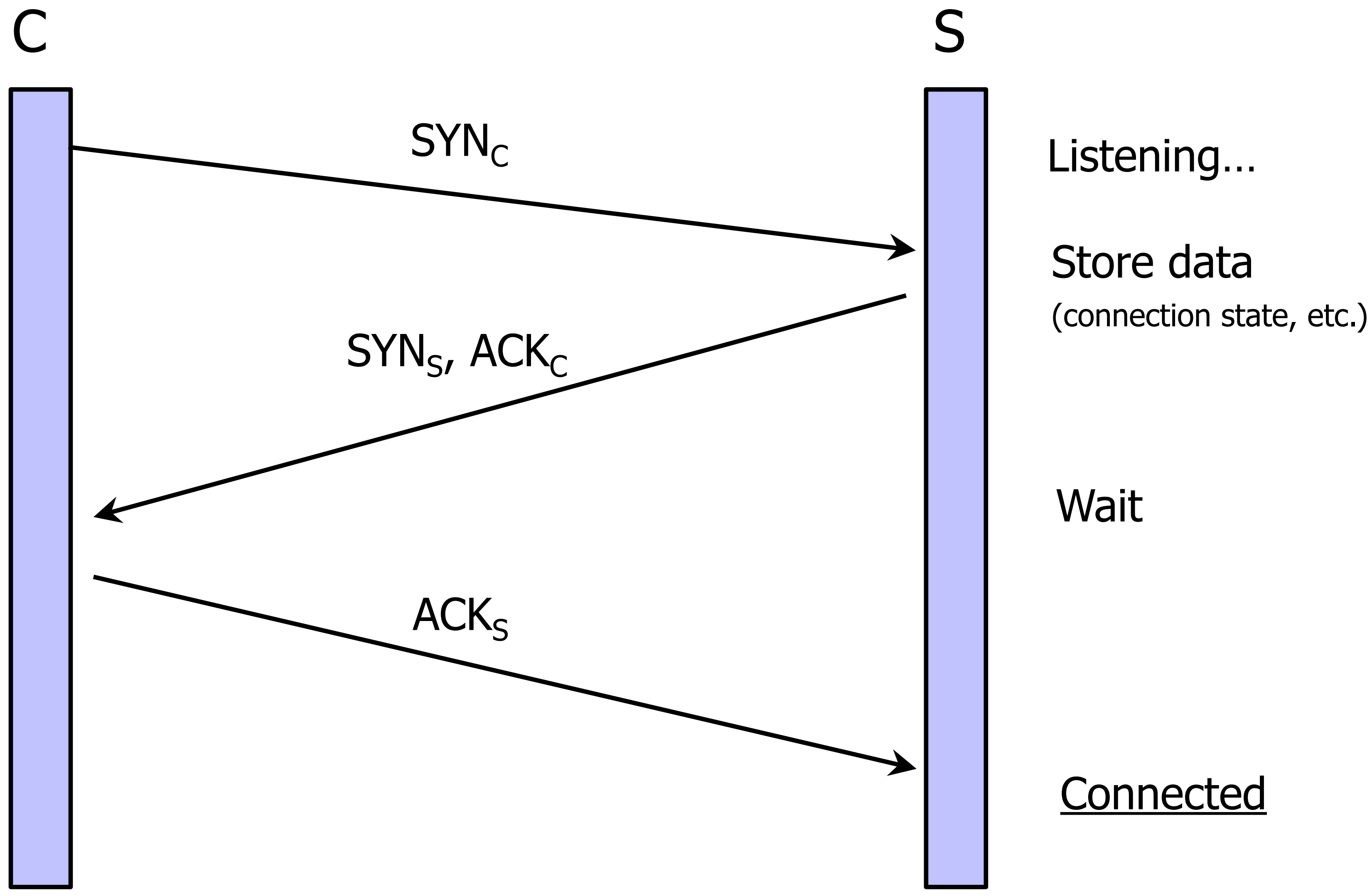
TCP Handshake



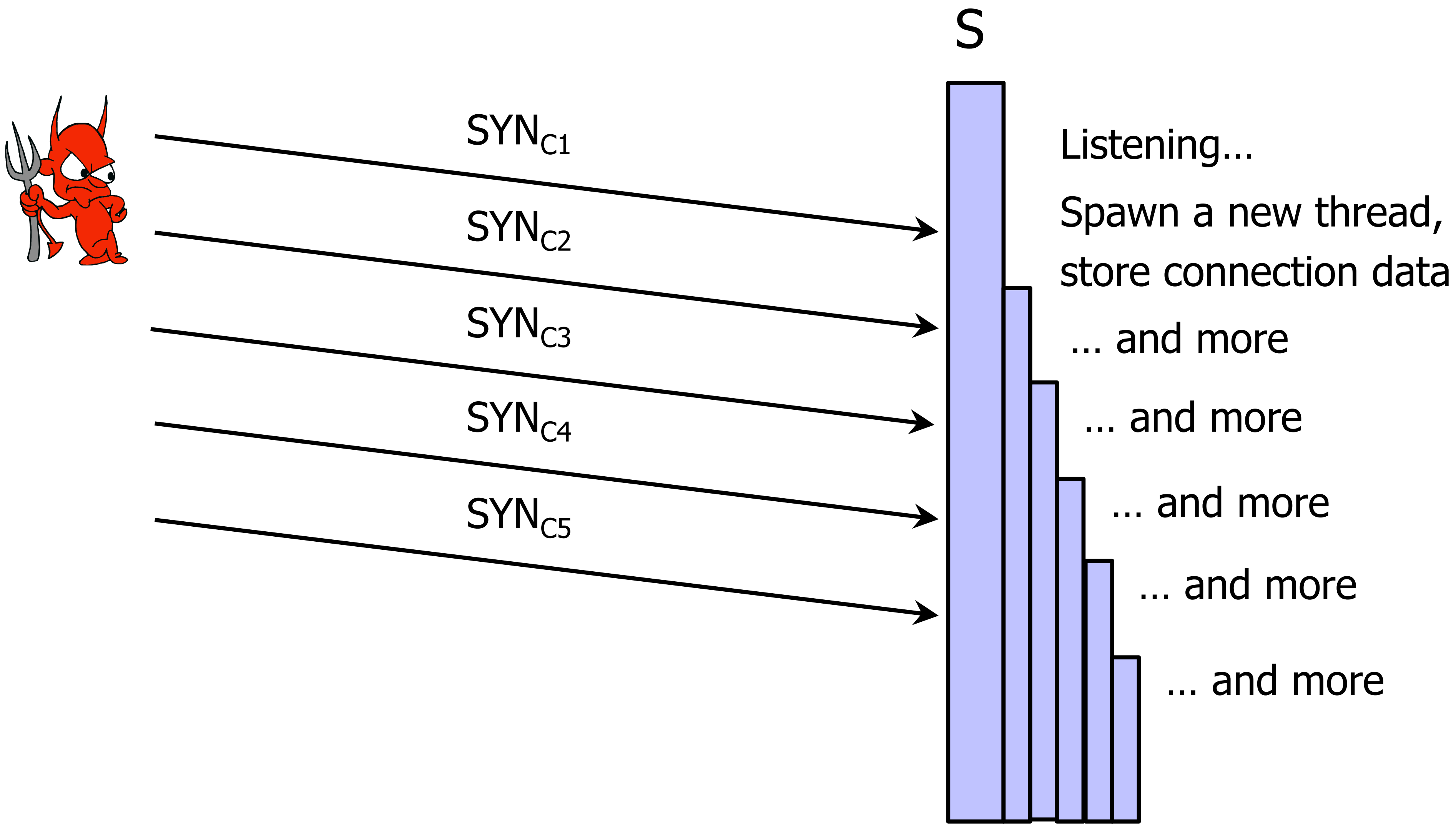
TCP Handshake



TCP Handshake



SYN Flooding Attack



SYN Flooding Explained

- Attacker sends many connection requests with spoofed source addresses
- Victim allocates resources for each request
  - Connection state maintained until timeout
  - Fixed bound on half-open connections
- Once resources exhausted, requests from legitimate clients are denied
- This is a classic denial of service (DoS) attack
  - Common pattern: it costs nothing to TCP initiator to send a connection request, but TCP responder must allocate state for each request (asymmetry!)