TCP/IP ATTACKS
DENIAL OF SERVICE

VITALY SHMATIKOV

Warm Up: 802.11b

NAV (Network Allocation Vector)

- 15-bit field, max value: 32767
- Any node can reserve channel for NAV microseconds
- No one else should transmit during NAV period

  … but not followed by most 802.11b cards

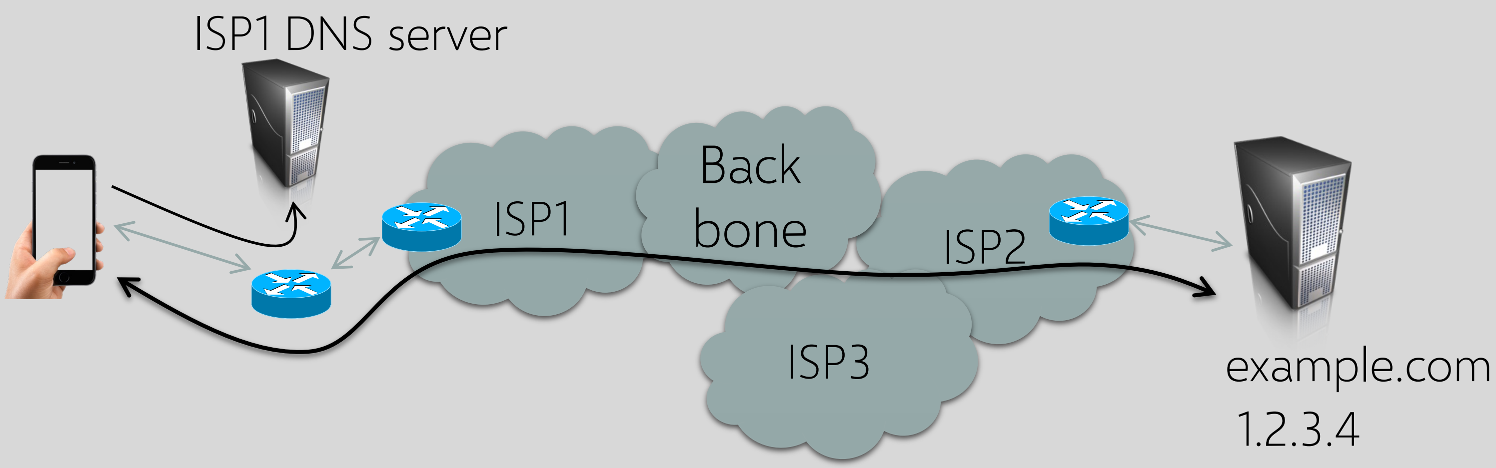
De-authentication

- Any node can send deauth packet to AP
- Deauth packet unauthenticated

  … attacker can repeatedly deauth anyone

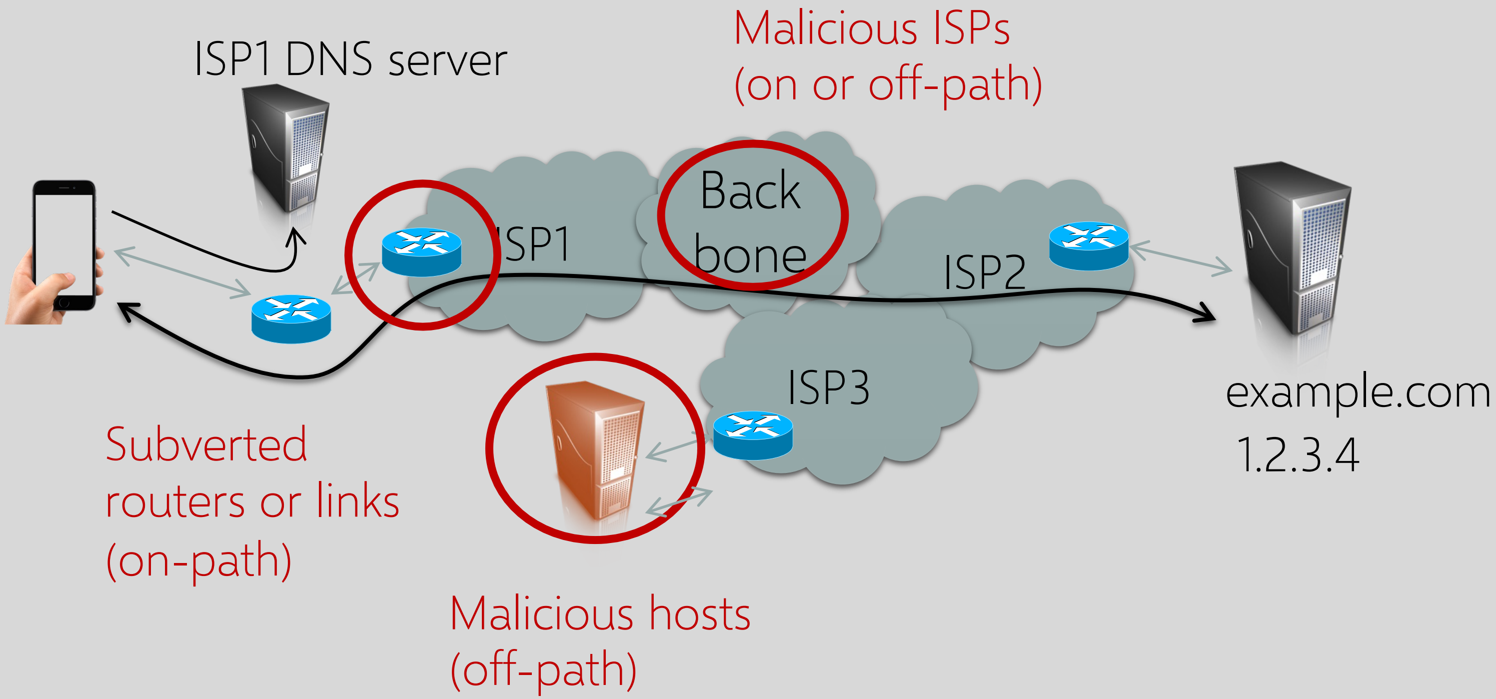
Pre HTTP/3

Steps to Send an HTTP Request

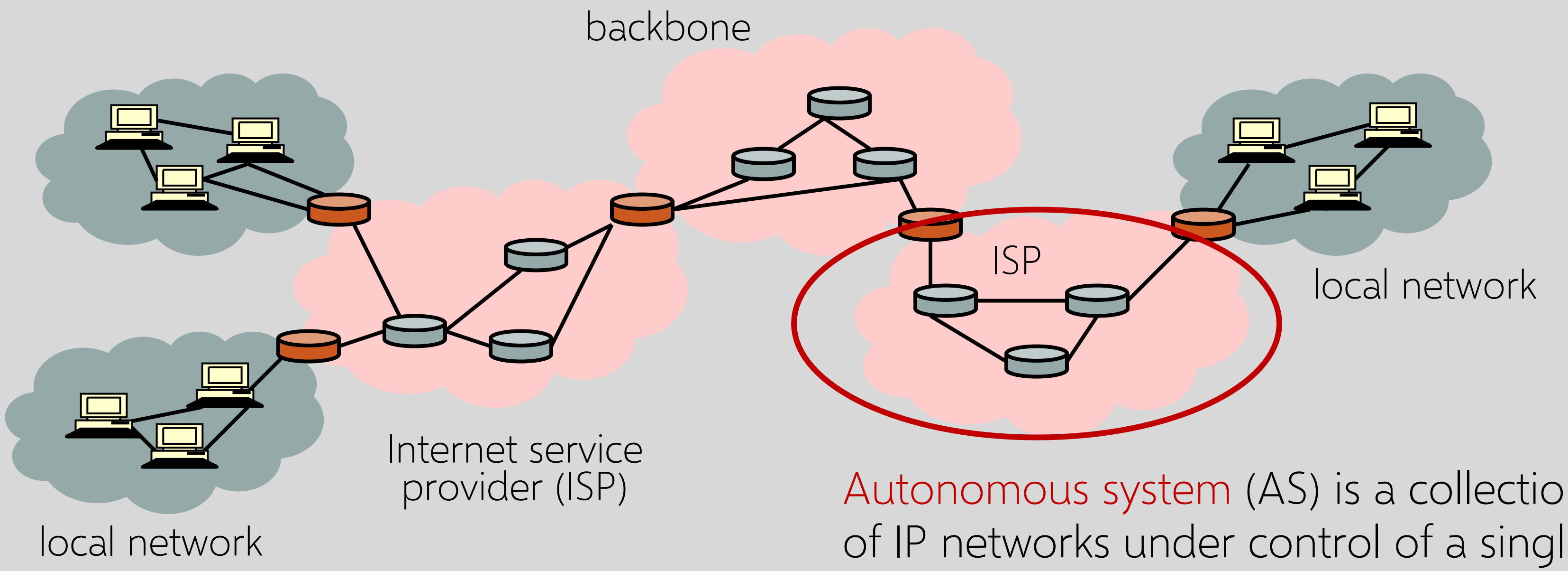


1. DNS lookup on example.com to get IP address (1.2.3.4)
2. TCP connection setup via 4-way handshake of IP packets to and from 1.2.3.4
3. Send HTTP request over TCP connection

Network Threat Models



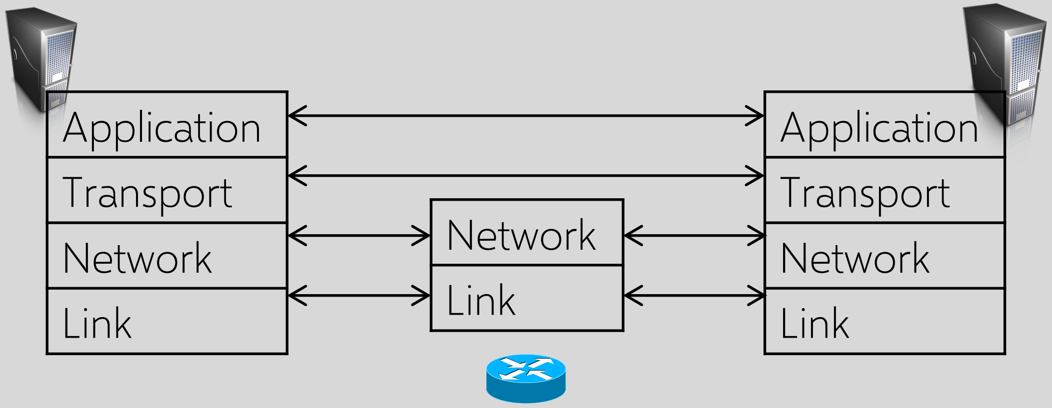
Internet Is a Network of Networks



Autonomous system (AS) is a collection of IP networks under control of a single administrator (e.g., ISP)

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

Internet Protocol Stack



HTTP, DNS, SMTP, SSH, etc.

TCP, UDP

IP, ICMP

802x (Ethernet, 802.11)

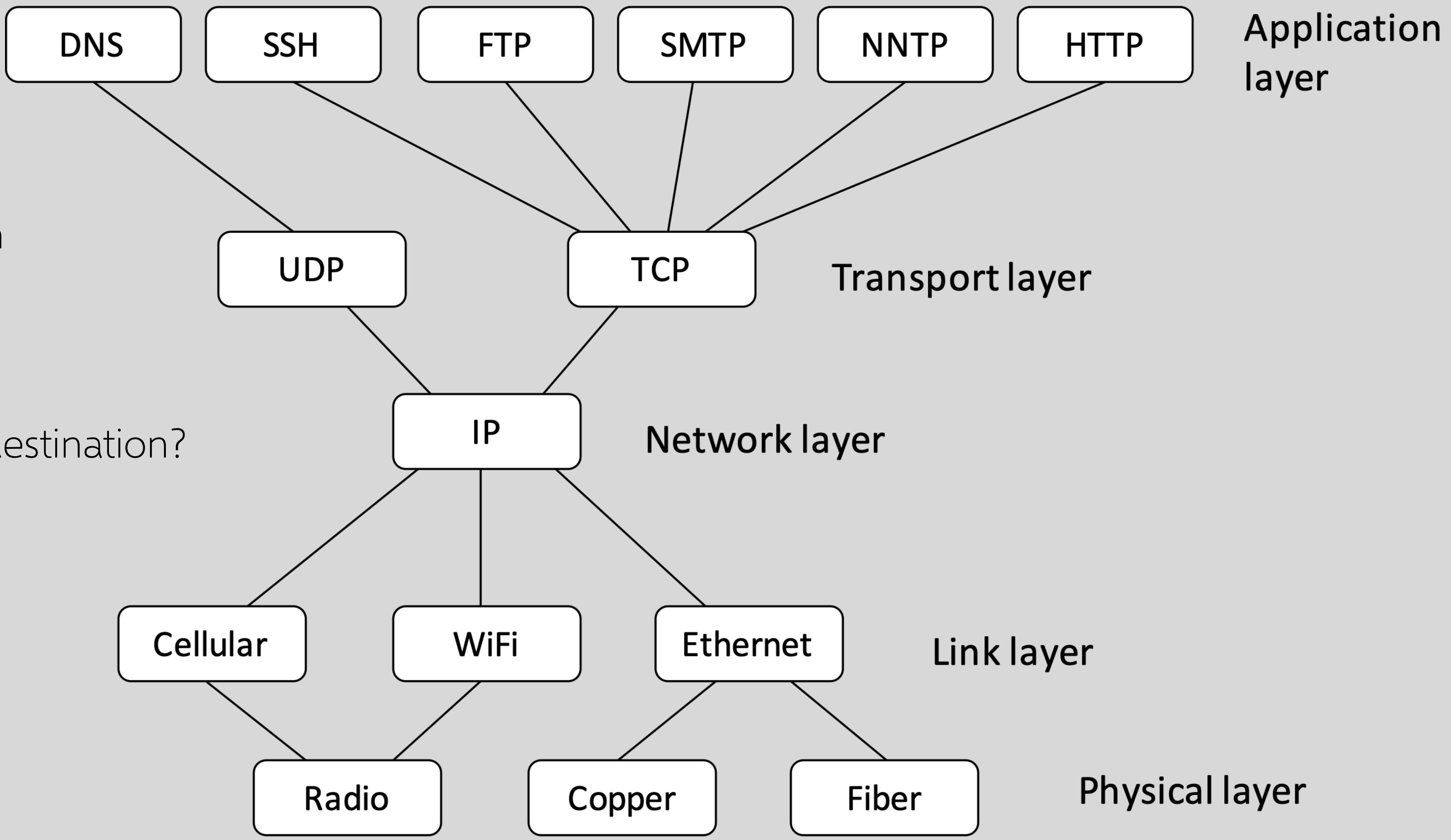
IP: "The Narrow Waist"

How does application structure data?

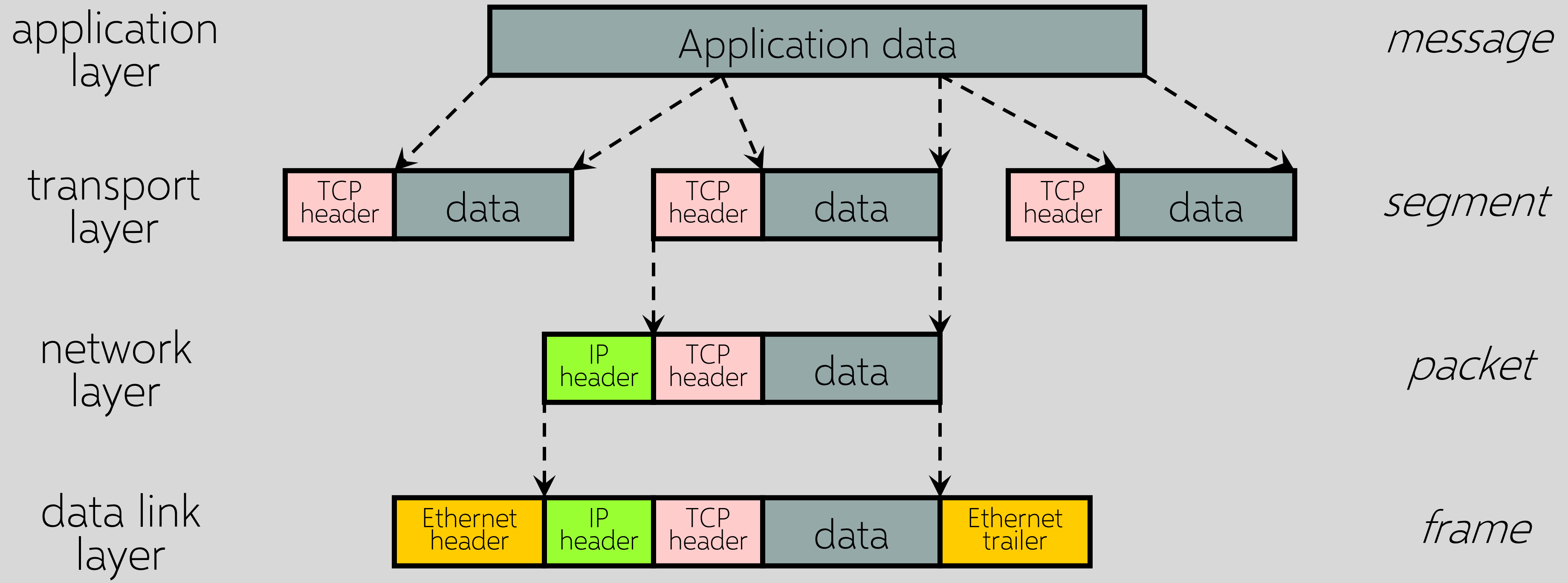
How to set up a communication stream or channel?

How does a packet reach final destination?

How does information get to next hop?



Data Formats



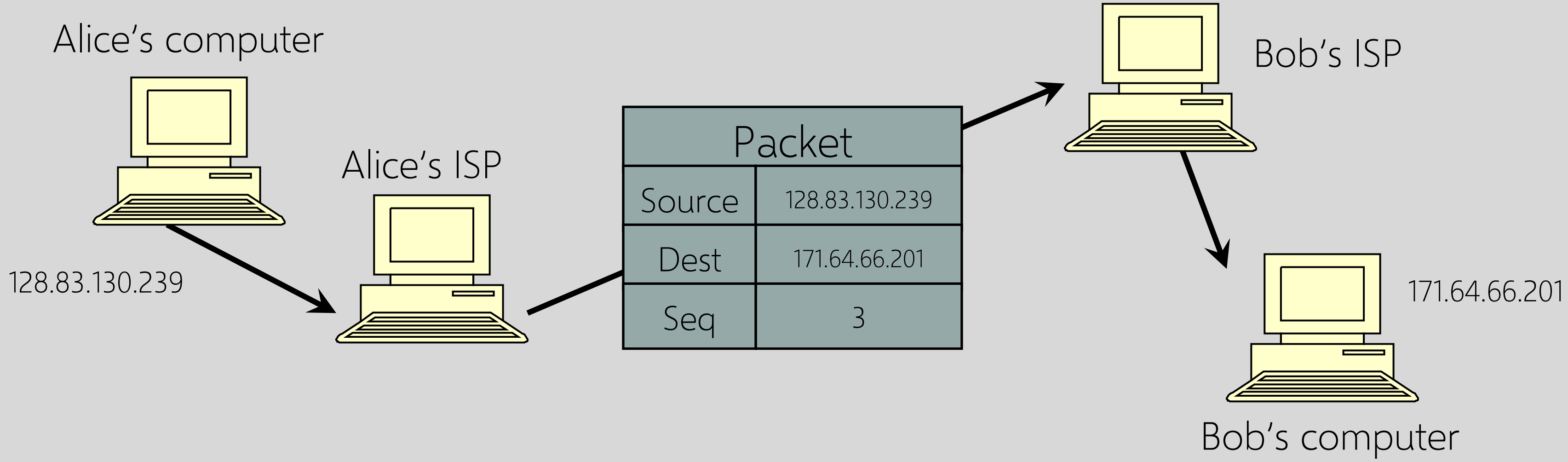
IP (Internet Protocol)

Connectionless

- Unreliable, “best-effort” protocol: no ordering, no retransmission, no error checking, no acknowledgement

Uses numeric addresses for routing

- Typically, several hops in the route



IP Is Not Enough for Packet Delivery

Given an IP packet, how does the router know where to send it next?

BGP

On a local network, what MAC address corresponds to a given IP address?

ARP

Global IPv4 Addresses

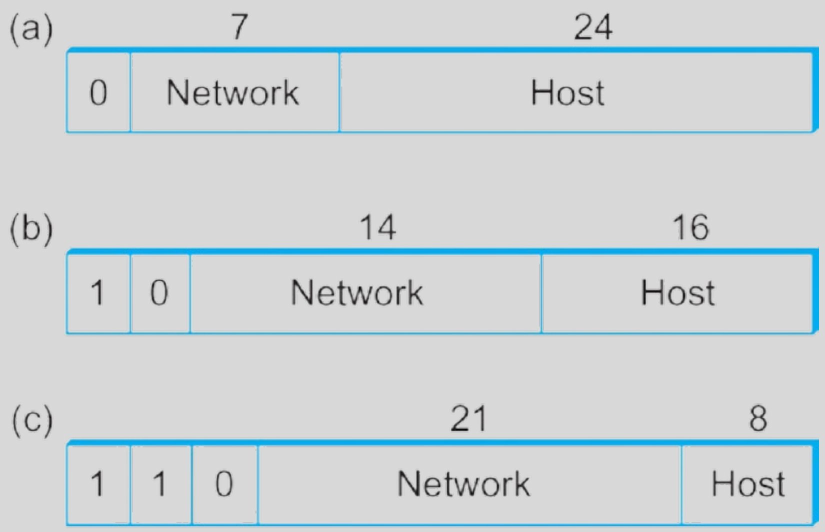
Globally unique

Hierarchical: network + host

Prefix of IP address identifies the network it belongs to

Dot notation

- 10.3.2.4
- 128.96.33.81
- 192.12.69.77



Class A network: owns all addresses with a given top byte
Class B network: ... top 2 bytes
Class C network: ... top 3 bytes

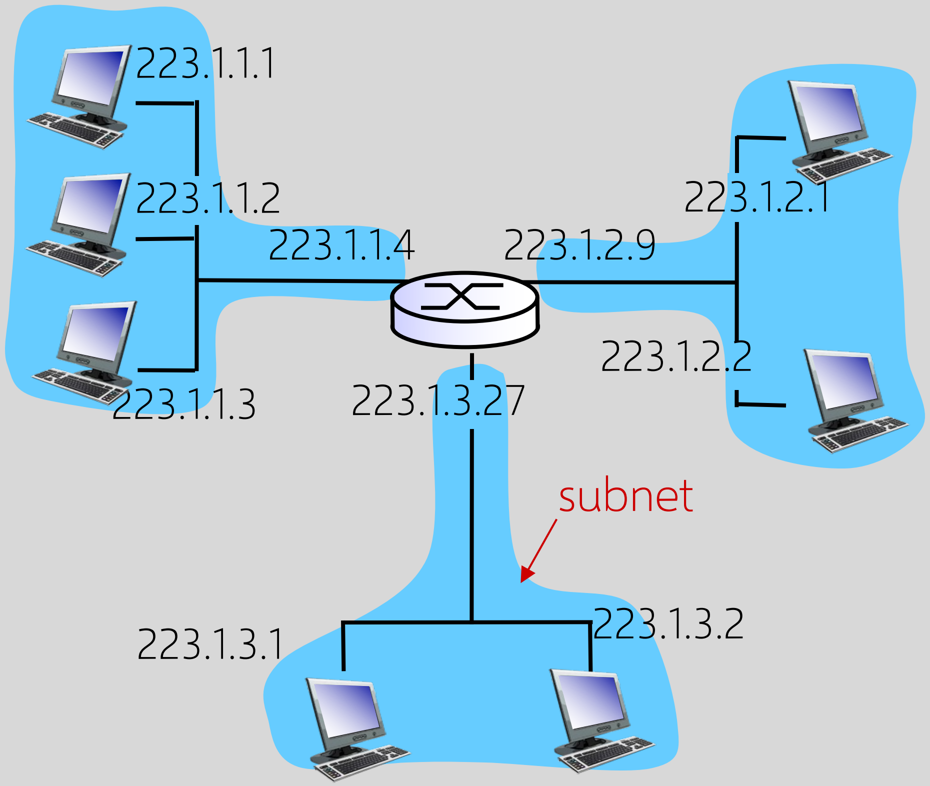
Subnet Definition

IP address

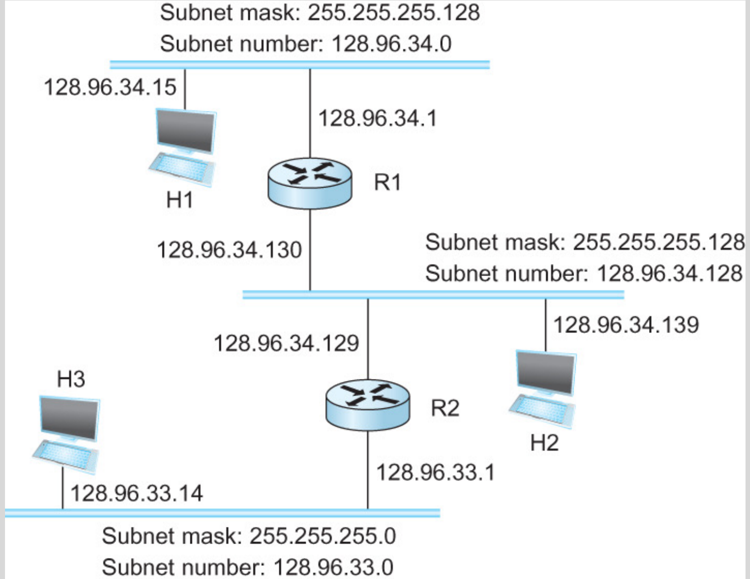
- Subnet part: high-order bits
- Host part: low-order bits

What's a subnet ?

- Device interfaces with the same subnet part of IP address
- Can physically reach each other without an intervening router



Subnetting Example



SubnetNumber	SubnetMask	NextHop
128.96.34.0	255.255.255.128	Interface 0
128.96.34.128	255.255.255.128	Interface 1
128.96.33.0	255.255.255.0	R2

Forwarding Algorithm

D = destination IP address

for each entry <SubnetNum, SubnetMask, NextHop>

    D1 = SubnetMask & D

    if D1 = SubnetNum

        if NextHop is an interface

            deliver datagram directly to destination

        else

            deliver datagram to NextHop (a router)

CIDR: Classless Inter-Domain Level Routing

CIDR balances two objectives: (1) minimize the number of routes that a router needs to know, and (2) hand out addresses efficiently

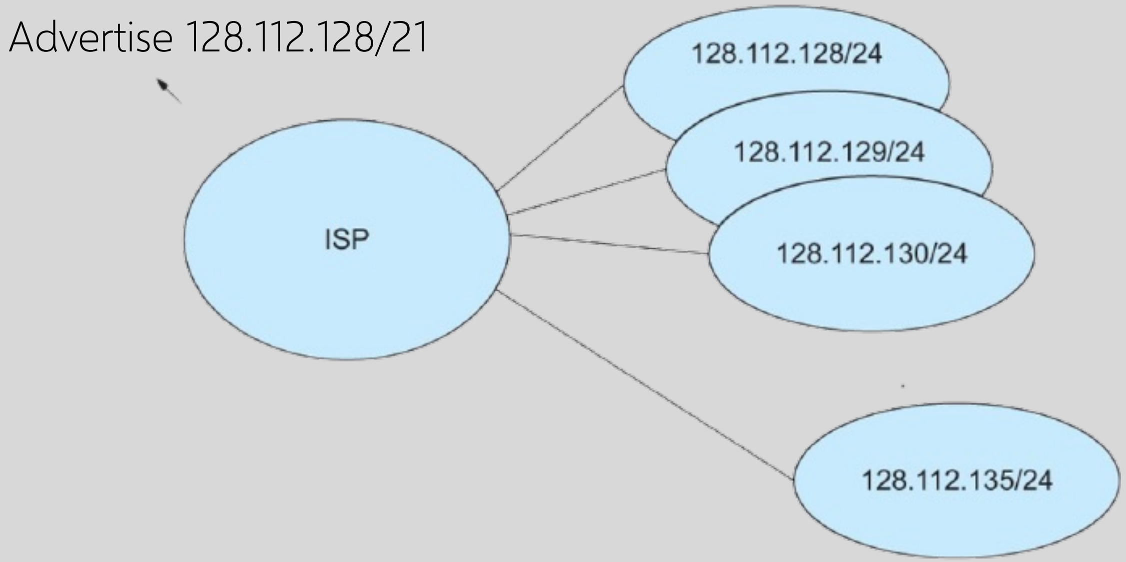
CIDR uses aggregate routes

- A single entry in the forwarding table tells the router how to reach a lot of different networks
- No rigid boundaries between address classes
- Variable number of bits per aggregated ranges of addresses

Classless Addressing

Network number may be of any length

Represent network number with a single <length, value> pair



Classless Address Block Management

AS with 16 class-C networks: instead of handing out 16 class-C addresses at random, hand out a block of class-C addresses that share a common prefix

- E.g., class-C network numbers from 192.4.16 through 192.4.31, so the top 20 bits of all addresses in this range are the same (11000000 00000100 0001)
- This implicitly creates a 20-bit network number

Prefix convention: /X after prefix, where X is prefix length in bits

- 20-bit prefix for 192.4.16 through 192.4.31: 192.4.16/20
- A single class-C network number, 24 bits long: 192.4.16/24

IP Forwarding w/ Longest Match

Router tables may have prefixes that overlap

- Some addresses may match more than one prefix
- Both 171.69 (a 16-bit prefix) and 171.69.10 (a 24-bit prefix) in the forwarding table of a single router
- Packet destined to 171.69.10.5 matches both prefixes

Matching is based on the principle of "longest match"

- 171.69.10 in the above case

A packet destined to 171.69.20.5 would match to 171.69 and not 171.69.10

Longest Prefix Matching

When looking for forwarding table entry for given destination address, use longest (ie, most specific) address prefix that matches destination address

Destination Address Range	Link interface
11001000 00010111 00010*** *********	0
11001000 00010111 00011000 *********	1
11001000 00010111 00011*** *********	2
otherwise	3

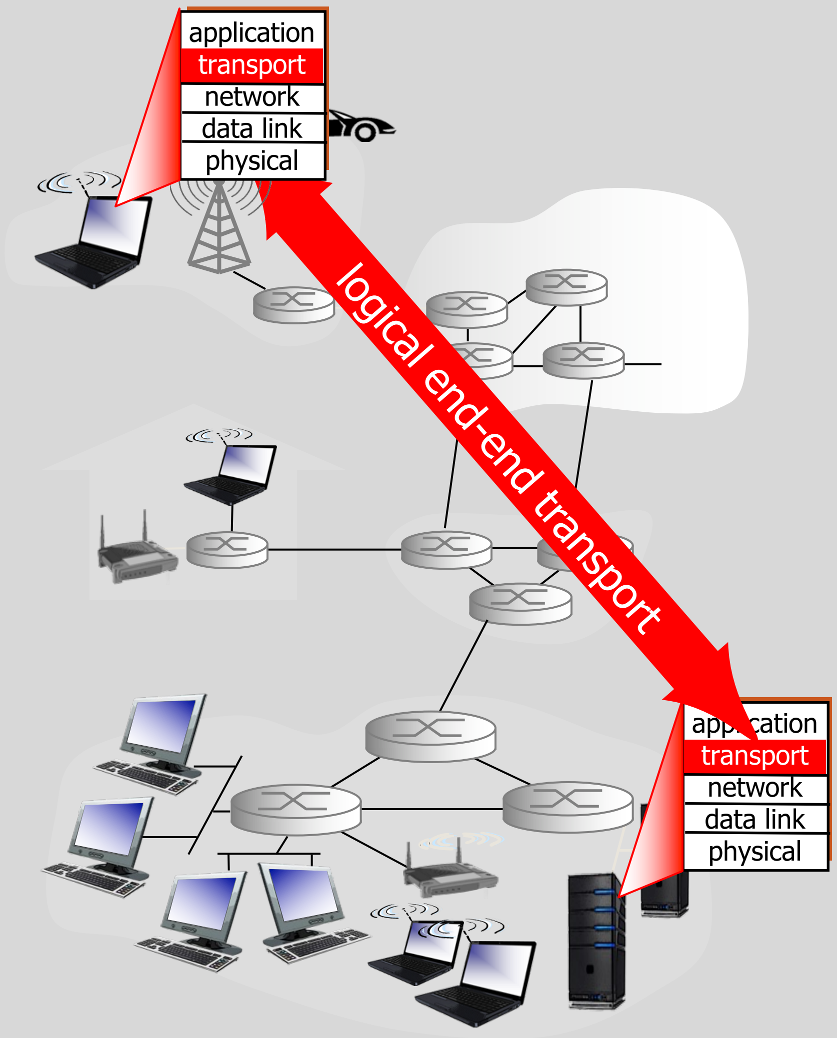
DA: 11001000 00010111 00010110 10100001 which interface?

DA: 11001000 00010111 00011000 10101010 which interface?

Transport Protocols

Goal: provide logical end-to-end communication channel between app processes running on different hosts

- Examples: TCP and UDP



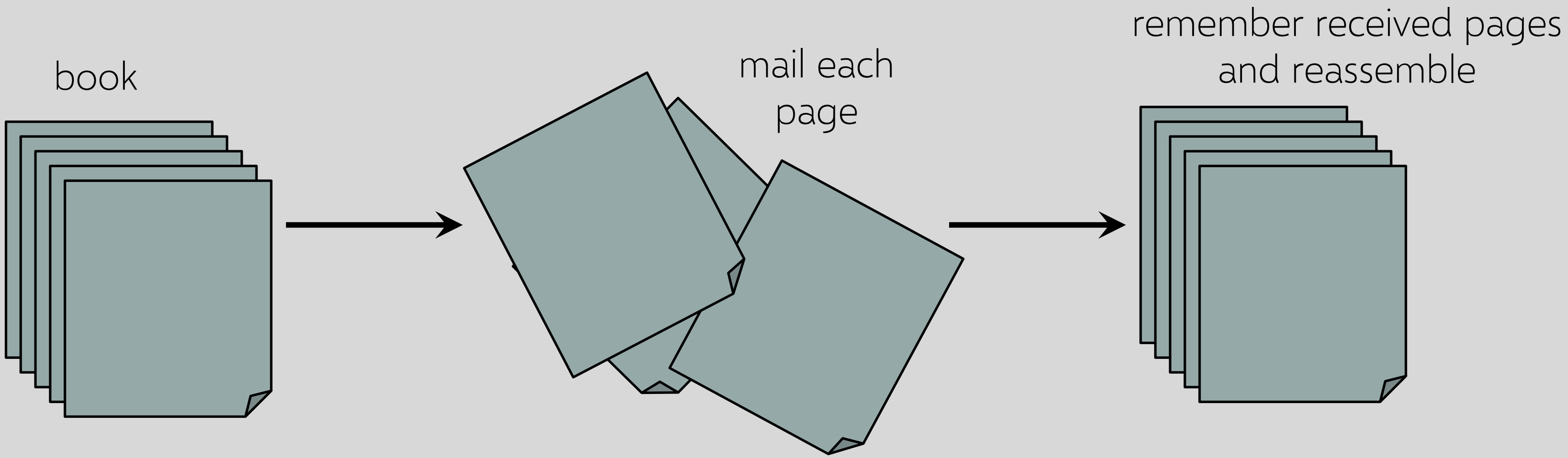
TCP (Transmission Control Protocol)

Sender: break data into packets, attach **sequence number** to every packet

Receiver: reassemble packets in correct order

- Acknowledge receipt; lost packets are re-sent

Connection state maintained on both sides



User Datagram Protocol (UDP)

UDP is a connectionless protocol

- Simply send datagram to application process at the specified port of the IP address
- Source port number provides return address
- Applications: media streaming, broadcast

No acknowledgement, no flow control, no message continuation

ICMP (Control Message Protocol)

Provides feedback about network operation

- “Out-of-band” messages carried in IP packets

Error reporting, congestion control, reachability…

- Destination unreachable
- Time exceeded
- Parameter problem
- Redirect to better gateway
- Reachability test (echo / echo reply)
- Message transit delay (timestamp request / reply)

The Internet Was Designed in the 1970s…

All transmissions are in the clear

- Eavesdropping

Nothing is authenticated or integrity-protected

- Spoofing, packet injection, etc.

Anyone can freely talk to anyone

- Denial of service

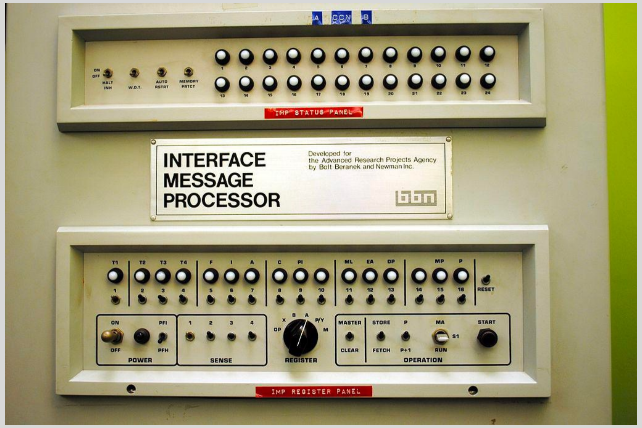
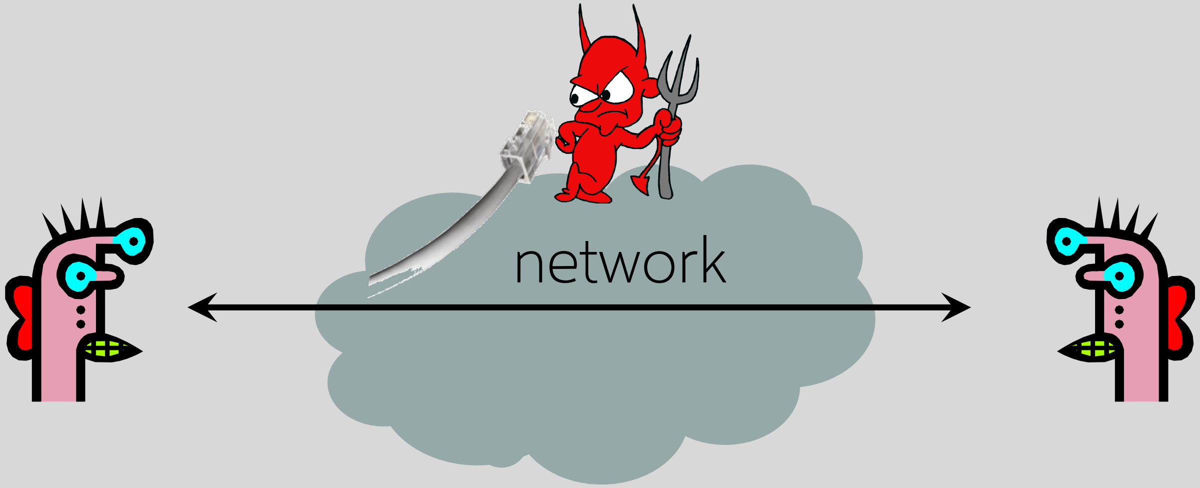


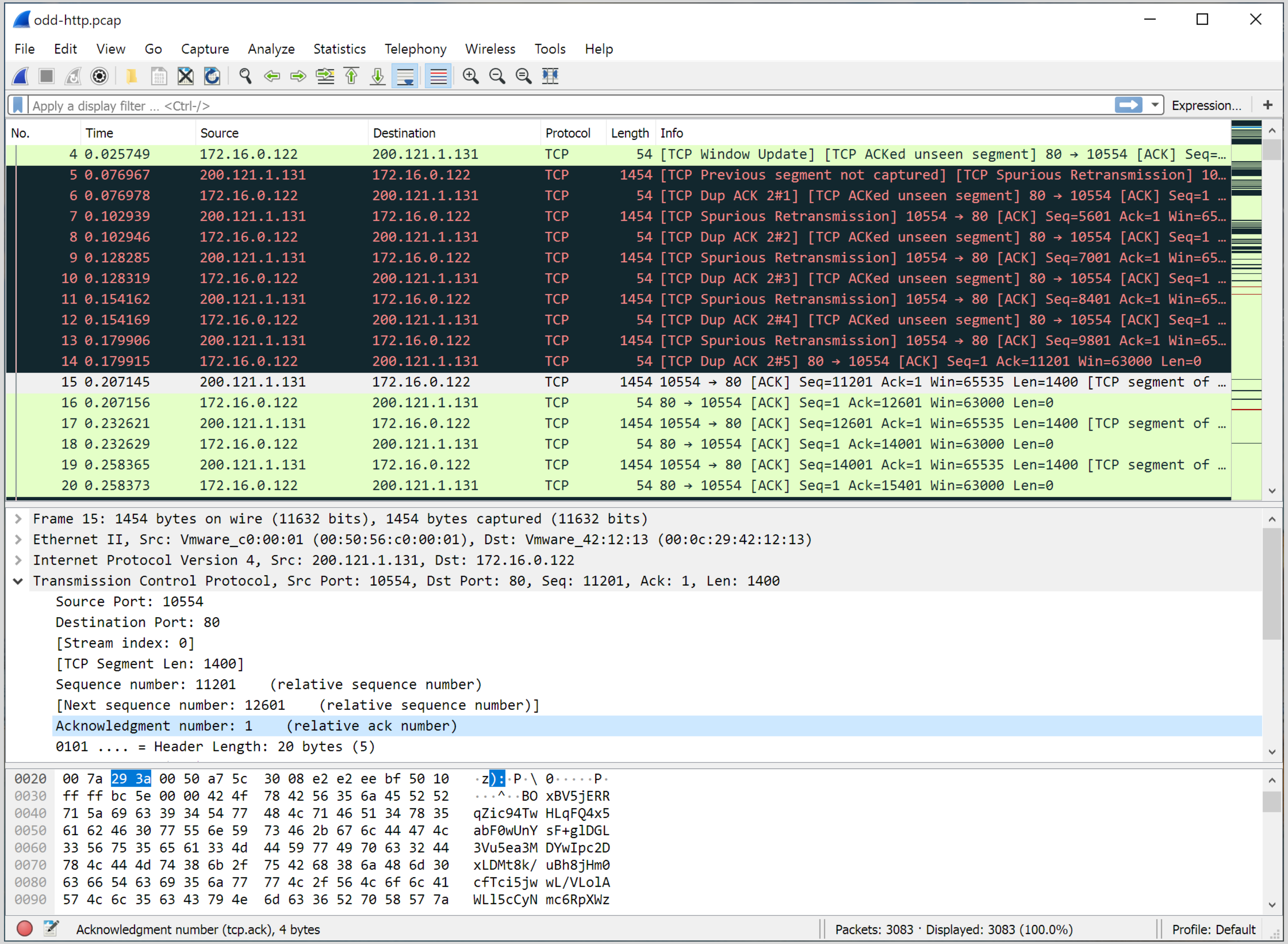
Image: Forbes

Packet Sniffing

Network interface card (NIC) in “promiscuous mode” reads all passing data

Many applications send data unencrypted





Wireshark: free, open-source protocol analyzer

Solution: use up-to-date TCP/IP implementation, ingress filtering

TCP/IP Implementation Bugs

"Ping of Death"

- Old versions of Windows would crash if ICMP packet has payload longer than 64K (illegal per protocol RFC)

"Teardrop" and "Bonk"

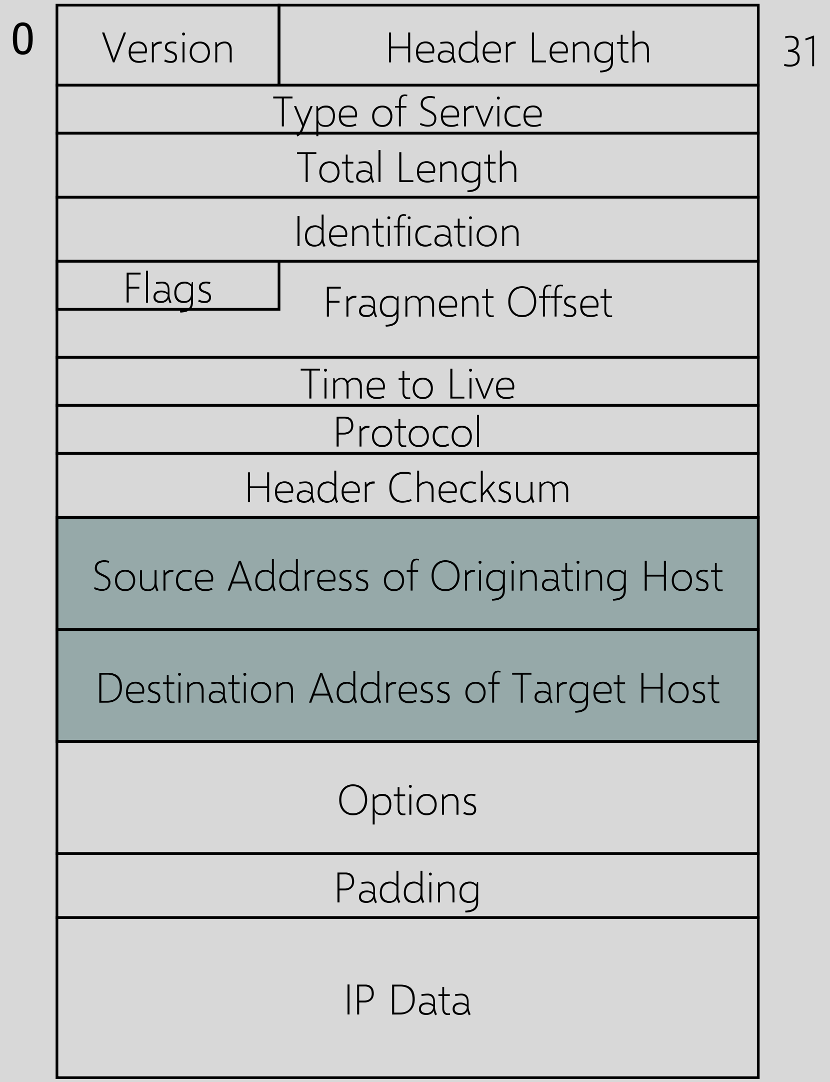
- Bad implementations of TCP/IP crash if Offset fields in TCP fragments are set to large or overlapping values

"Land"

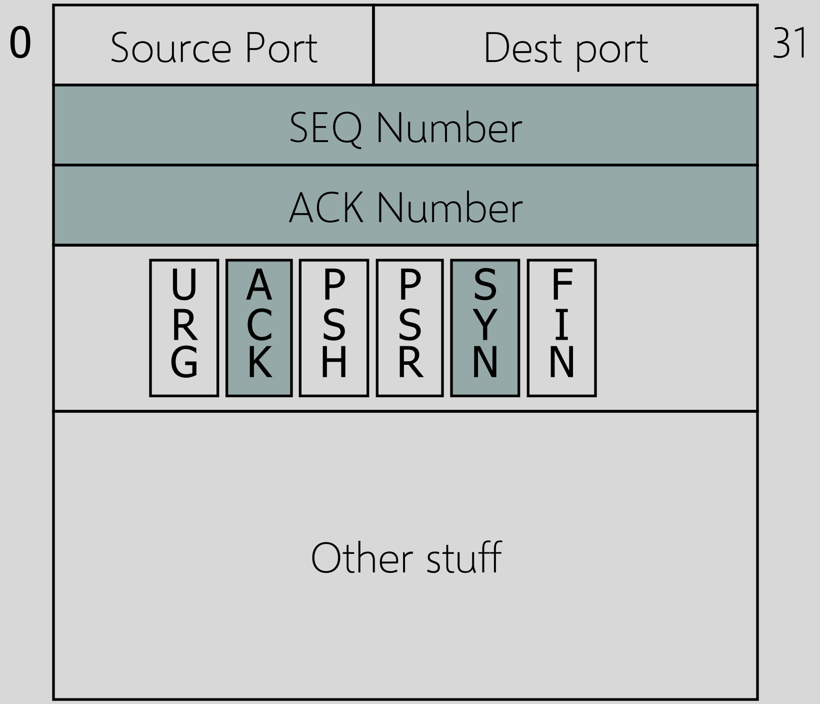
- IP packet with source address, port equal to destination address, port and SYN flag set triggers loopback in the Windows XP SP2 implementation of TCP/IP stack, locks up CPU

Common cause: implementors assume that input will follow specification … but attackers can and will send malformed inputs

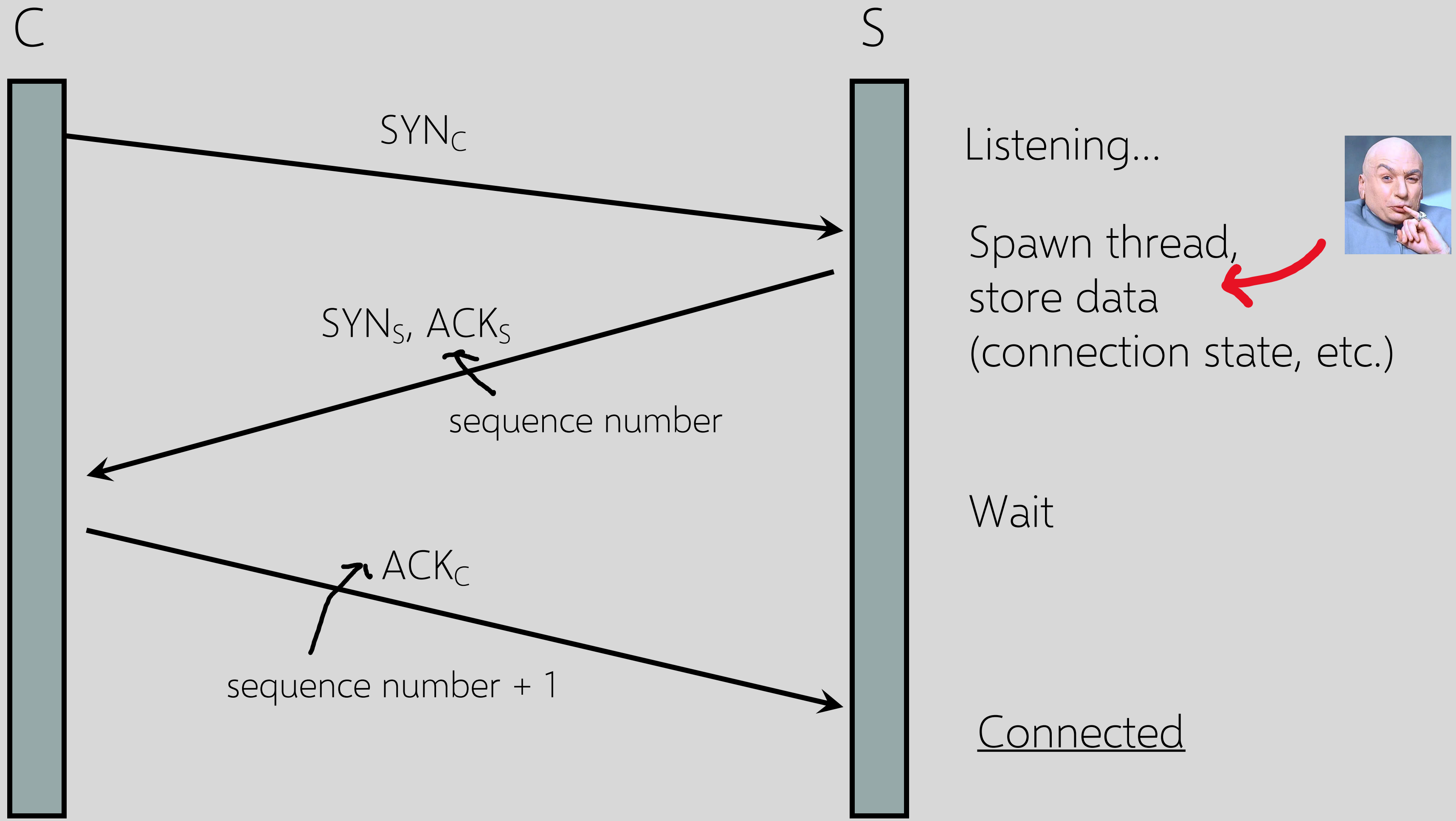
IP header



TCP header

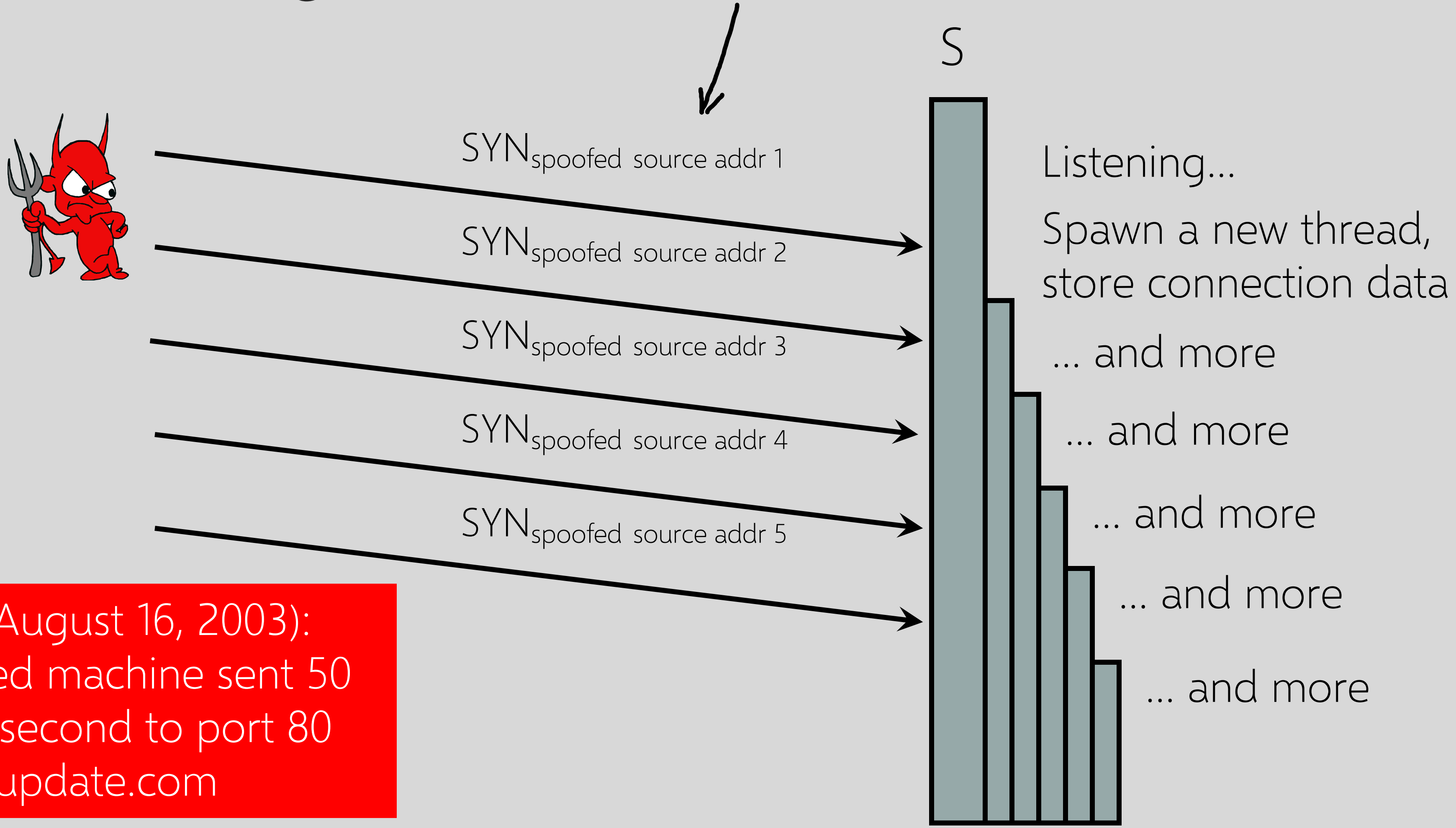


TCP Handshake



SYN Flooding Attack

IP does not verify or authenticate source address of packets



MS Blaster (August 16, 2003): every infected machine sent 50 packets per second to port 80 on windowsupdate.com

Low-Rate SYN Floods

OS	Backlog queue size
Linux 1.2.x	10
FreeBSD 2.1.5	128
WinNT 4.0	6

Backlog timeout: 3 minutes

Attacker need only send 128 SYN packets every 3 minutes

Phrack 48, no 13, 1996

SYN Flooding Explained

Attacker sends many connection requests with spoofed source addresses

Victim allocates resources for each request

- New thread, 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 pattern: it costs nothing to TCP initiator to send a connection request, but TCP responder must spawn a thread for each request - asymmetry!

Preventing SYN Floods

DoS is caused by asymmetric state allocation

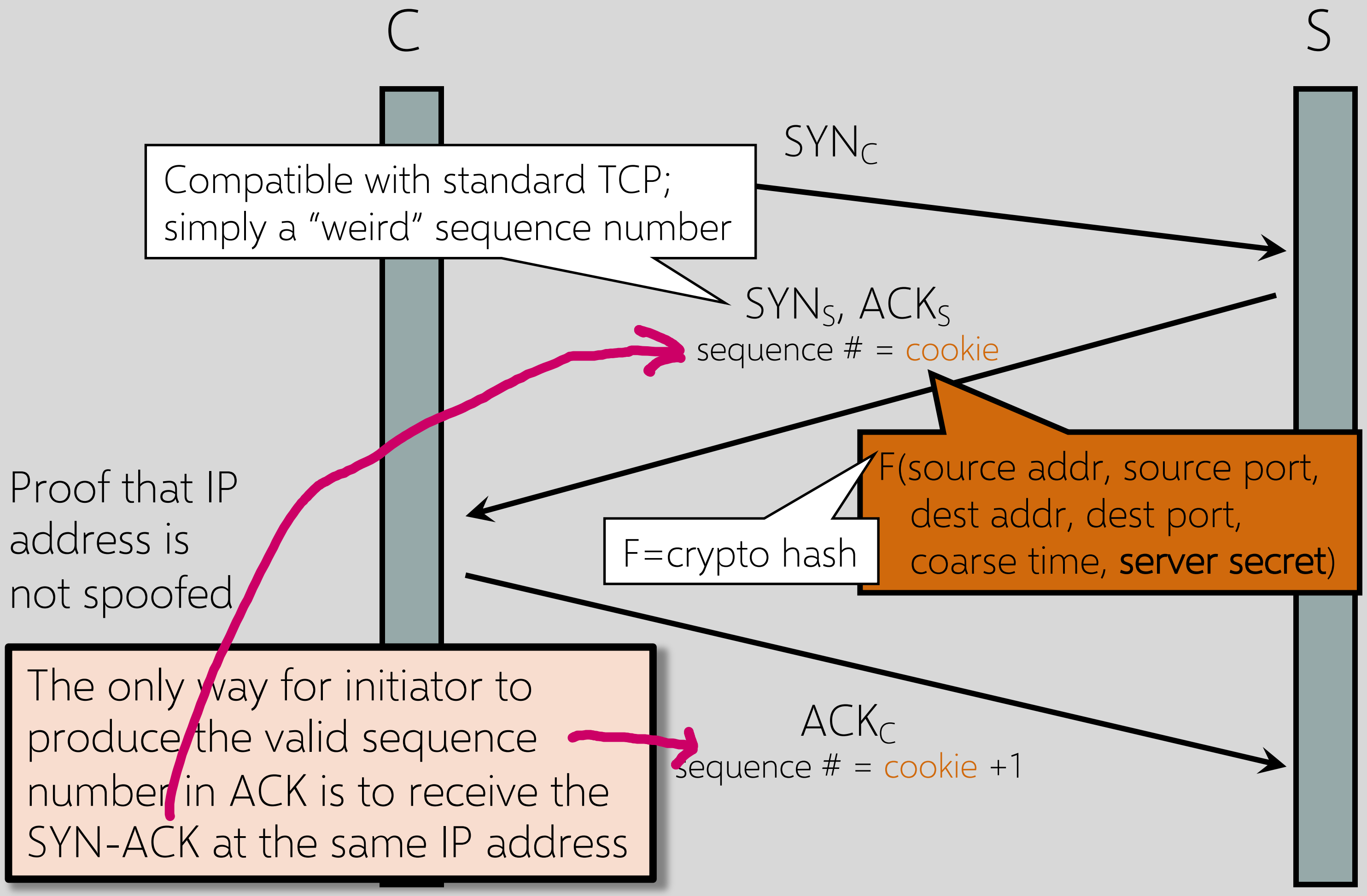
- If responder opens new state for each connection attempt, attacker can initiate thousands of connections from bogus or spoofed IP addresses

Cookies ensure that the responder is stateless until initiator produced at least two messages

- Responder's state (IP addresses and ports of the connection) is stored in a cookie and sent to initiator
- After initiator responds, cookie is regenerated and compared with the cookie returned by the initiator

http://cr.yp.to/syncookies.html

SYN Cookies



Listening…

Does not store state

Cookie must be unforgeable and tamper-proof

Recompute cookie, compare with with the received cookie, establish connection only if they match

Anti-Spoofing Cookies: Basic Pattern

Client sends request (message #1) to server

Typical protocol:

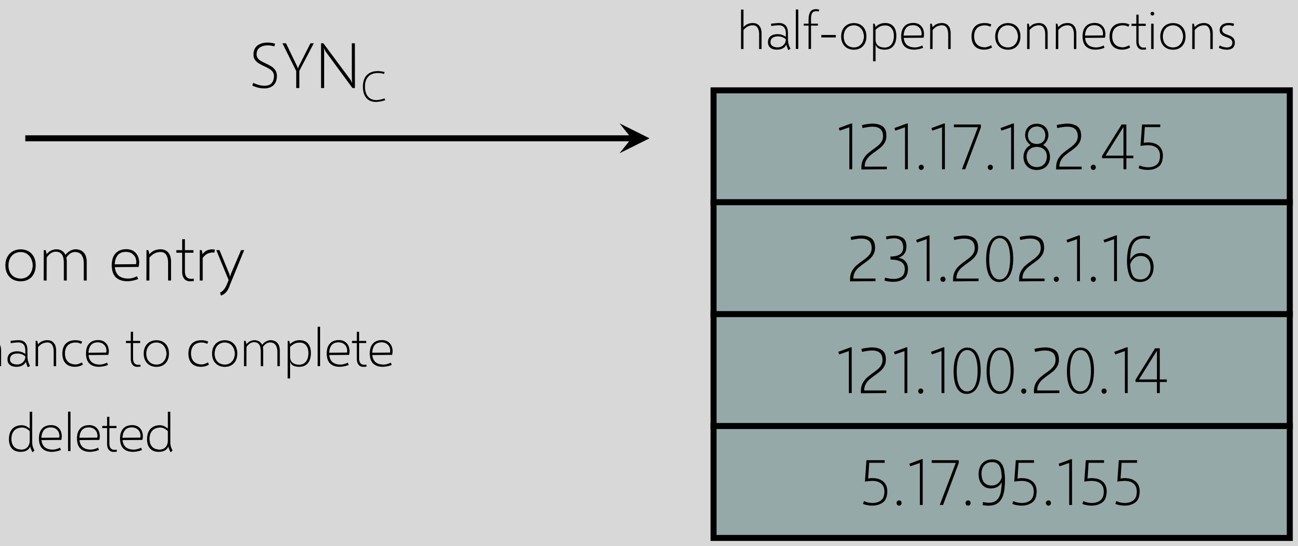
- Server sets up connection, responds with message #2
- Client may complete session or not - potential DoS!

Cookie version:

- Server responds with hashed connection data instead of message #2
- Client confirms by returning hashed data
- Need an extra step to send postponed message #2, <u>except</u> in TCP (can piggyback on SYN-ACK in TCP)

If source IP address is spoofed, attacker can't confirm

Another Defense: Random Deletion



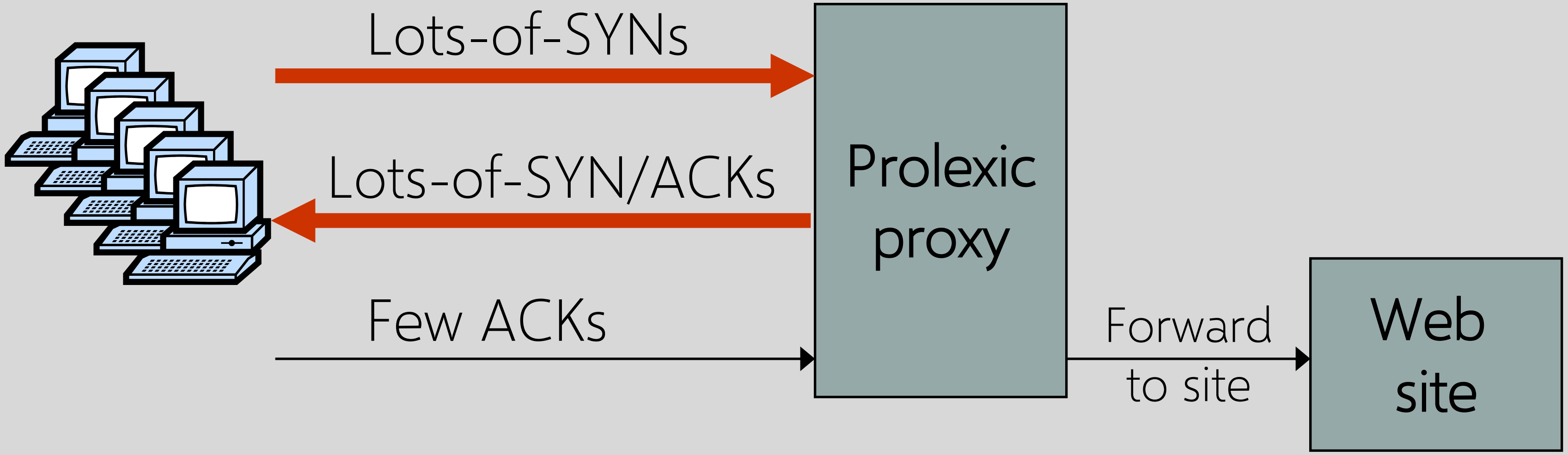
If SYN queue is full, delete random entry

- Legitimate connections have a chance to complete
- Fake addresses will be eventually deleted

Easy to implement

Prolexic, Google Project Shield, etc.

Idea: only forward established TCP connections to site



Prolexic purchased by Akamai in 2014

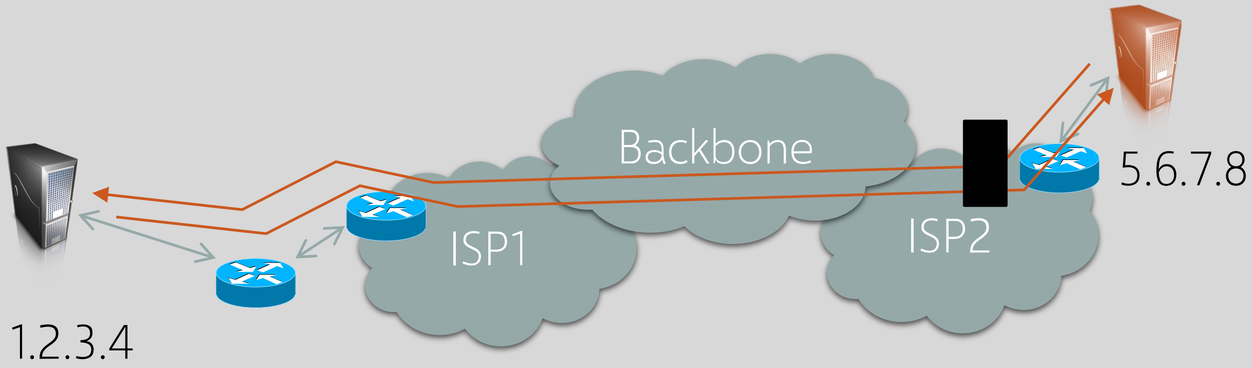
Many companies: Cloudflare, Imperva, Arbor Networks, …

Other Junk-Packet Attacks

Proxy must keep floods of these away from website

Attack Packet	Victim Response	Rate: attk/day [ATLAS 2013]
TCP SYN to open port	TCP SYN/ACK	773
TCP SYN to closed port	TCP RST	
TCP ACK or TCP DATA	TCP RST	
TCP RST	No response	
TCP NULL	TCP RST	
ICMP ECHO Request	ICMP ECHO Response	50
UDP to closed port	ICMP Port unreachable	387

Ingress Filtering

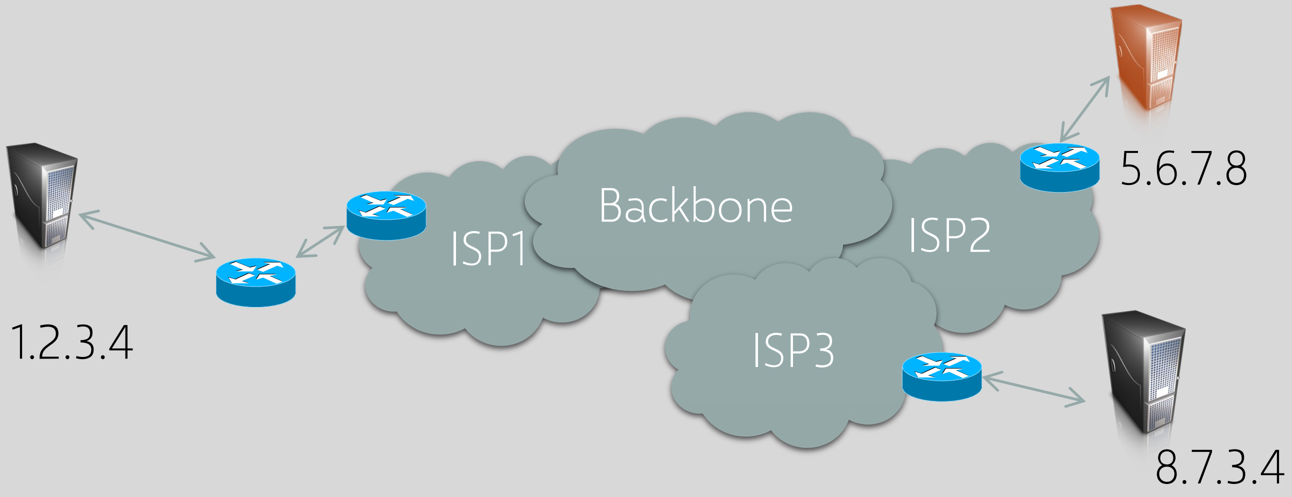


Attacker's goal: prevent legitimate users from accessing victim (1.2.3.4)

ICMP ping flood

- Attacker sends ICMP pings as fast as possible to victim
- When will this work as a DoS? Attacker resources > victim's
- How can this be prevented? Ingress filtering of attacker IP addresses near victim once attack identified

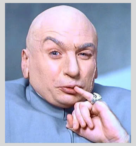
Predictable Sequence Numbers



BSD 4.4 used predictable initial sequence numbers (ISNs)

- At system initialization, set ISN to 1
- Increment ISN by 64,000 every half-second

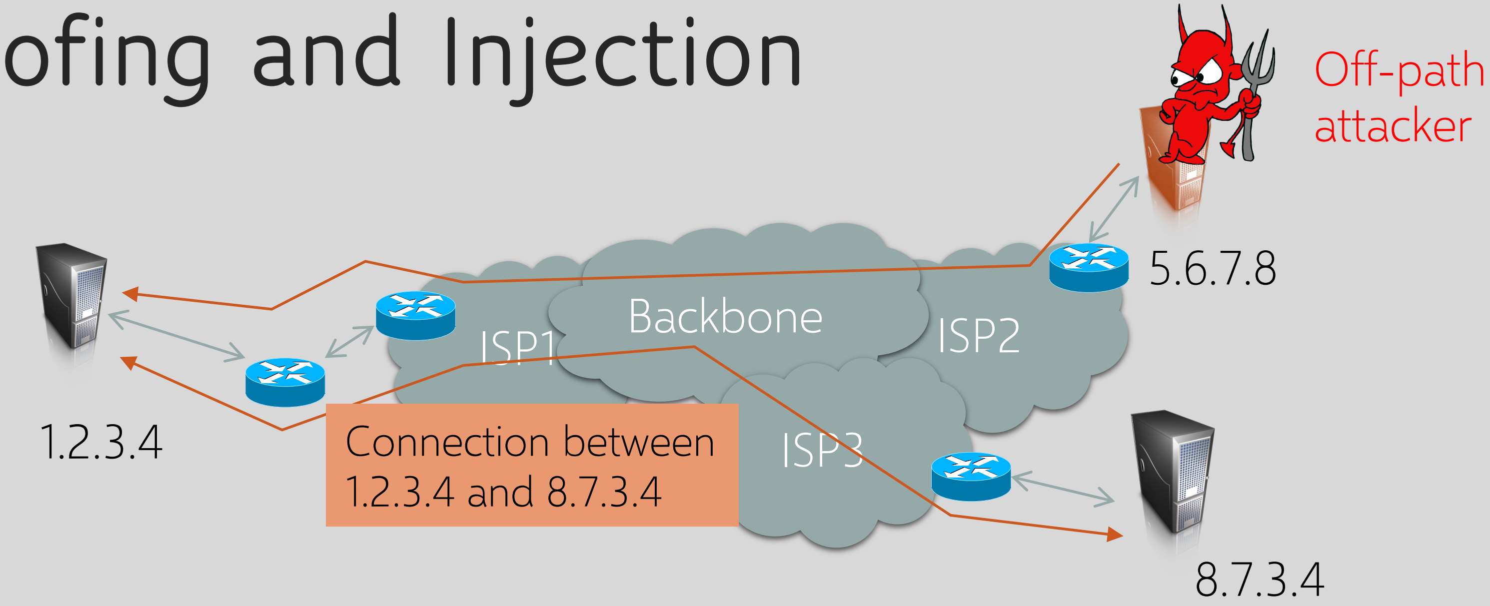
What can a clever attacker do?
(assume spoofing possible)



TCP Spoofing

- Each TCP connection has associated state
  - Sequence number, port number
- TCP state is easy to guess
  - Port numbers standard, seq numbers predictable
- Can inject packets into existing connections
  - If attacker knows the initial sequence number and amount of traffic, can guess likely current number
  - Guessing a 32-bit seq number is not practical, but most systems accept large windows of sequence numbers (to handle packet losses), so send a flood of packets with likely sequence numbers

TCP Spoofing and Injection



Forge a FIN packet from 8.7.3.4 to 1.2.3.4

src: 8.7.3.4
dst: 1.2.3.4

seq#(8.7.3.4)
FIN

DoS by connection reset

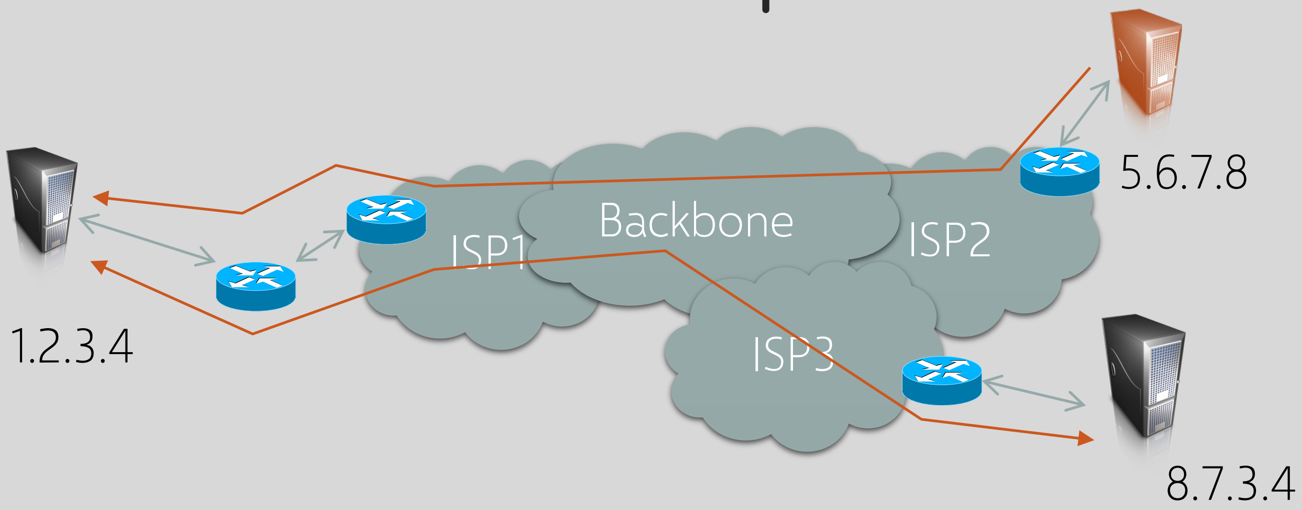
Forge some application-layer packet from 8.7.3.4 to 1.2.3.4

src: 8.7.3.4
dst: 1.2.3.4

seq#(8.7.3.4)
"rsh rm –rf /"

Attacker can't see server's responses, but can bypass IP address-based authentication (remote shell, SPF defense against spam)

How to Fix Predictable Seq Numbers



Fix idea 1:

- Random ISN at system startup
- Increment by 64,000 each half second

Better fix:

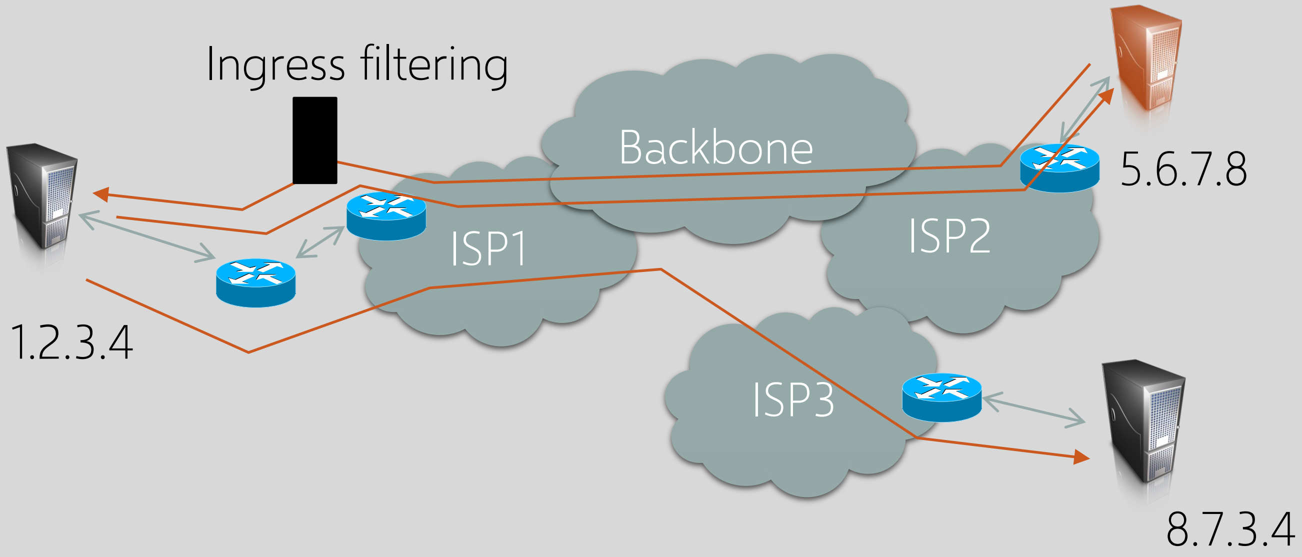
- Random ISN for every connection

SYN cookies ensure this

Remains an issue in some cases:

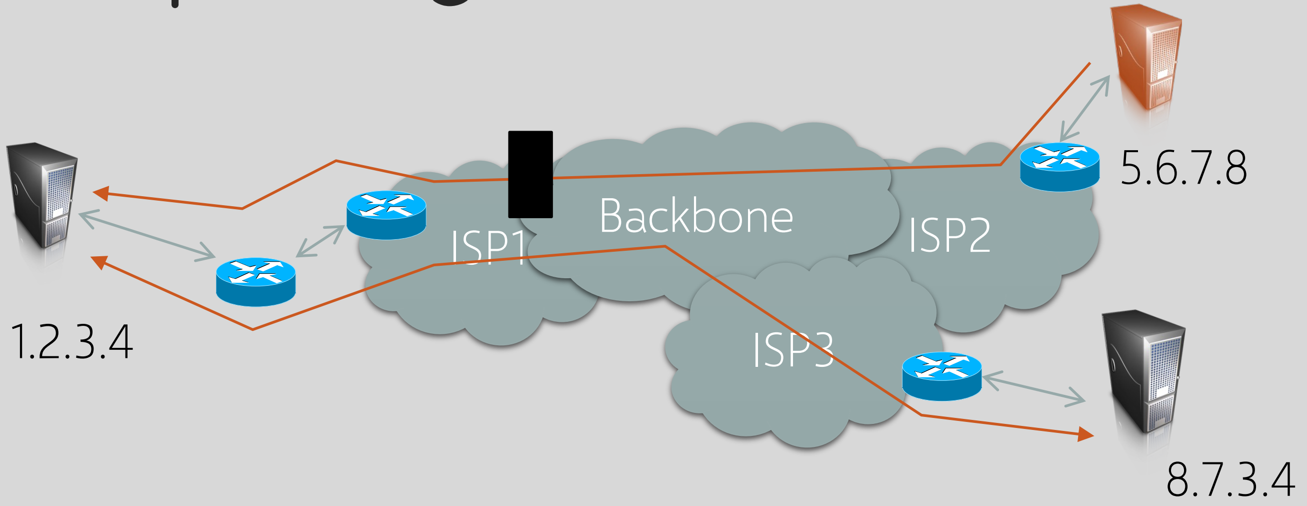
- Any FIN accepted with seq number in receiver window: 2^{17} attempts
- Side-channel attacks to infer seq number

Avoiding Ingress Filtering



1. Attacker can send packet with fake ("spoofed") source IP address. Packet will get routed correctly. Replies will not.
2. Distribute attack across many IP addresses 

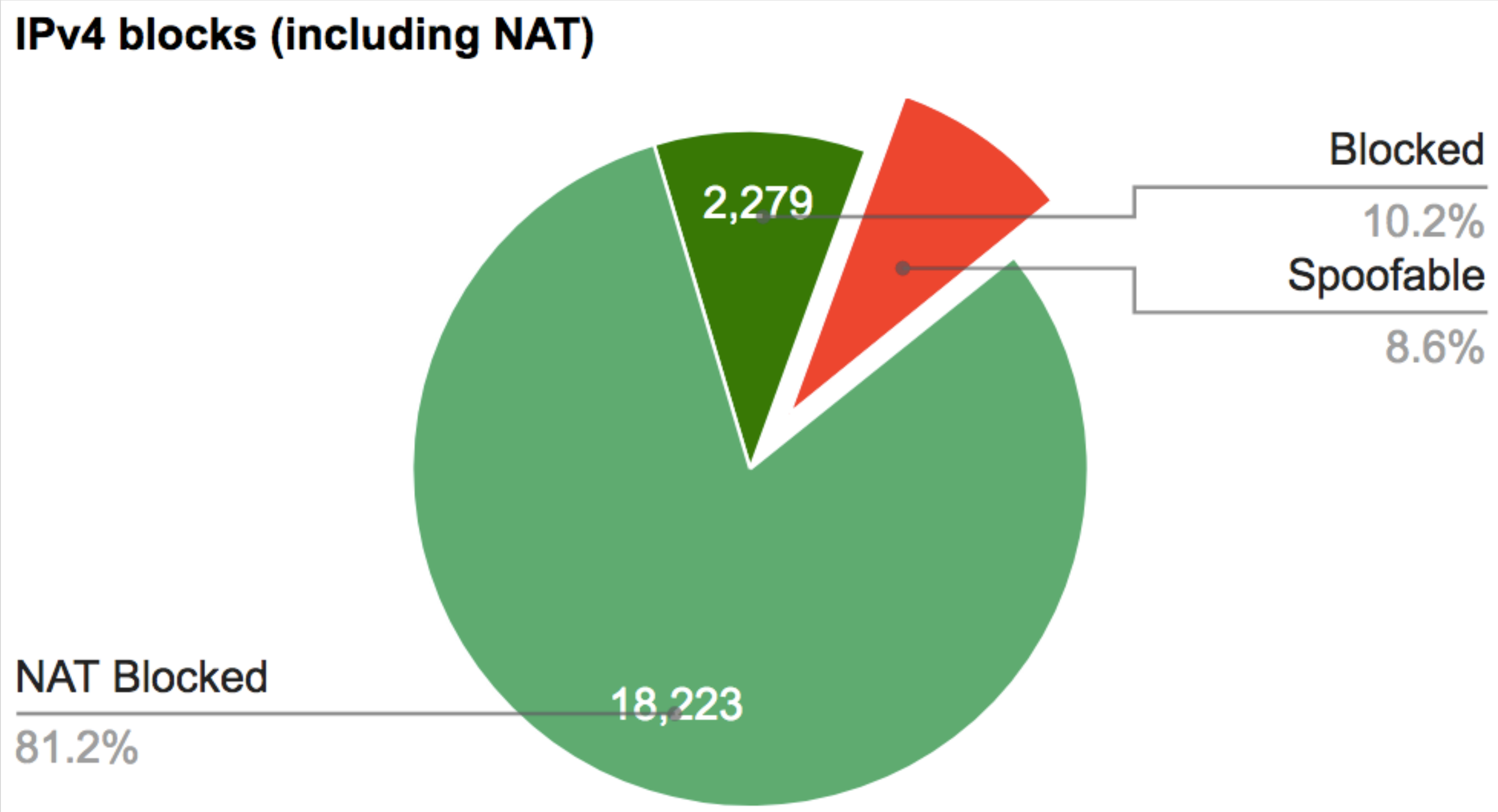
How to Fix Spoofing



IP traceback: techniques for inferring actual source of a (spoofed) packet

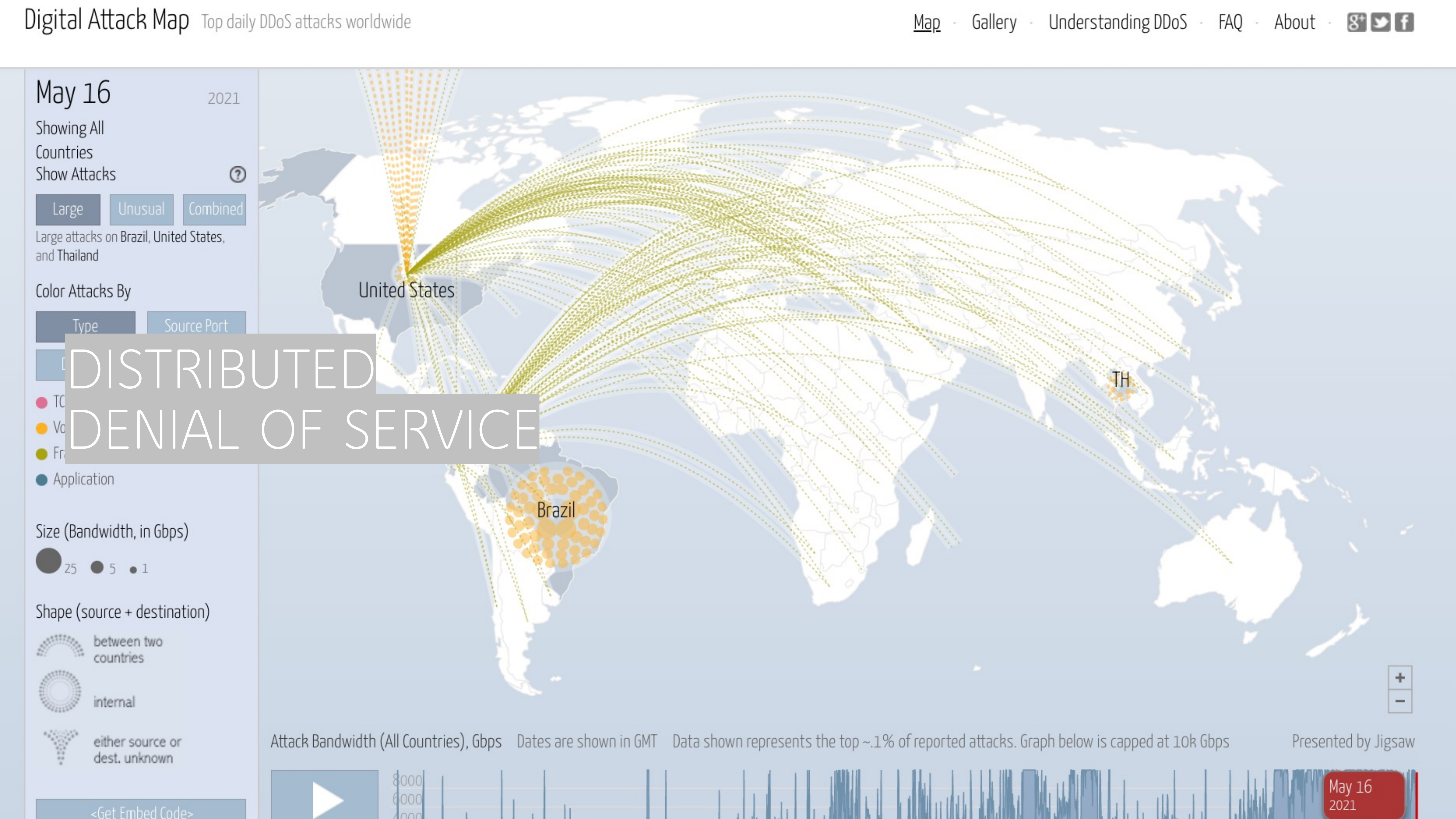
BCP 38 (RFC 2827): upstream ingress filtering to drop spoofed packets

- Ideally, all network traffic providers would perform ingress filtering… but they don't



https://spoofer.caida.org/summary.php

DISTRIBUTED DENIAL OF SERVICE



TCP Con Flood

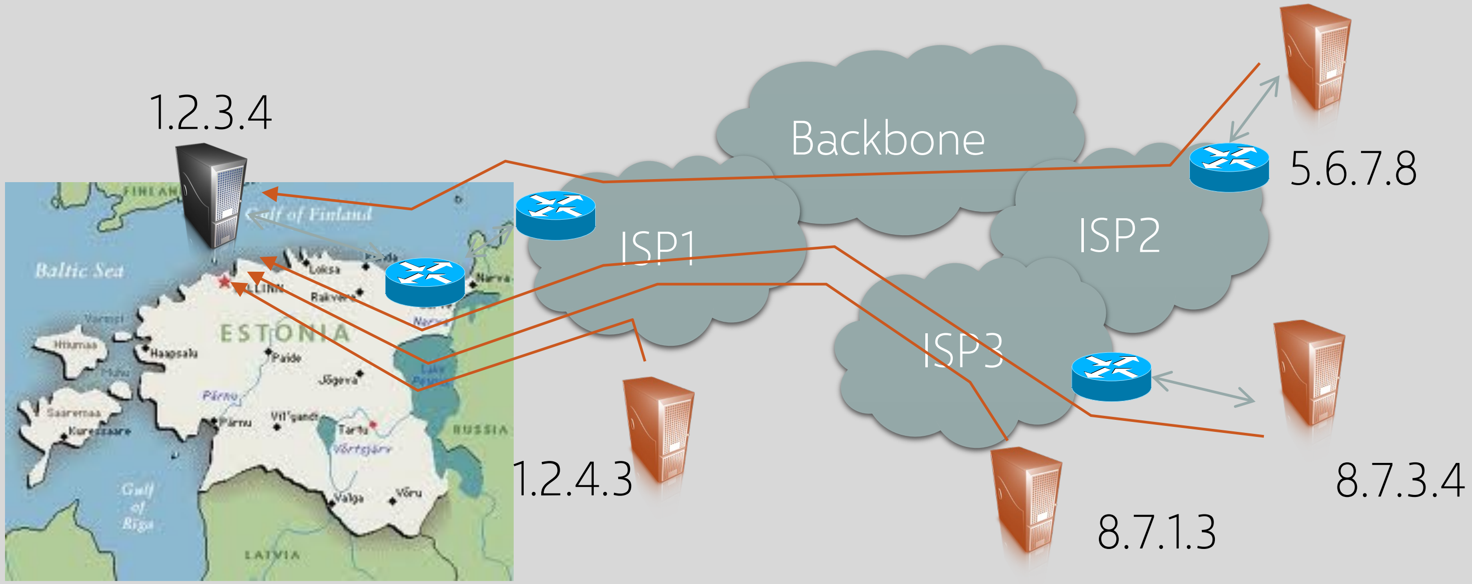
Command a bot/zombie army to:

- Complete TCP connection to web site
- Send short HTTP HEAD request
- Repeat

Will bypass SYN flood protection proxy but attacker cannot use spoofed source IPs

- Reveals location of bot zombies
- Proxy can now block or rate-limit bots

DDoS Attack on Estonia



April 27, 2007

Continued for weeks, with varying levels of intensity

Government, banking, news, university websites

Government shut down international Internet connections

Telegram blames China for 'powerful DDoS attack' during Hong Kong protests

Telegram CEO says 'IP addresses coming mostly from China' were to blame

By Jon Porter | @JonPorty | Jun 13, 2019, 4:21am EDT

Mirai Botnet

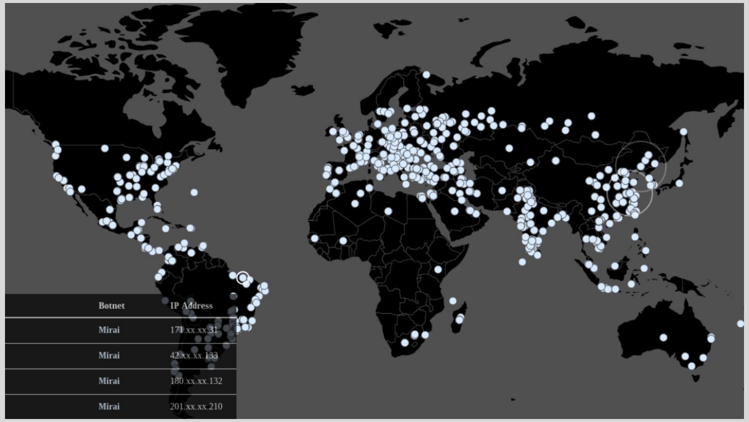


image: vice.com

Origin: game-booting code from Lizard Squad

- Used in attacks on Sony PlayStation and Xbox Live networks

Scans big blocks of Internet address space for open telnet ports, logs in using default passwords

- Assembled an army of 1 to 2.5 million IoT devices

In 2016, used to stage massive DDoS attacks on DYN's DNS servers

Knocked out access to 1200 websites, including Twitter, Netflix, Paypal, Shopify, GitHub…

Much more about DNS later

Mirai Exploits Default Passwords

Password	Device Type	Password	Device Type	Password	Device Type
123456	ACTi IP Camera	klv1234	HiSilicon IP Camera	1111	Xerox Printer
anko	ANKO Products DVR	jvbzd	HiSilicon IP Camera	Zte521	ZTE Router
pass	Axis IP Camera	admin	IPX-DDK Network Camera	1234	Unknown
888888	Dahua DVR	system	IQinVision Cameras	12345	Unknown
666666	Dahua DVR	meinsm	Mobotix Network Camera	admin1234	Unknown
vizxv	Dahua IP Camera	54321	Packet8 VOIP Phone	default	Unknown
7ujMko0vizxv	Dahua IP Camera	00000000	Panasonic Printer	fucker	Unknown
7ujMko0admin	Dahua IP Camera	realtek	RealTek Routers	guest	Unknown
666666	Dahua IP Camera	1111111	Samsung IP Camera	password	Unknown
dreambox	Dreambox TV Receiver	xmhdipc	Shenzhen Anran Camera	root	Unknown
juantech	Guangzhou Juan Optical	smcadmin	SMC Routers	service	Unknown
xc3511	H.264 Chinese DVR	ikwb	Toshiba Network Camera	support	Unknown
OxhlwSG8	HiSilicon IP Camera	ubnt	Ubiquiti AirOS Router	tech	Unknown
cat1029	HiSilicon IP Camera	supervisor	VideoIQ	user	Unknown
hi3518	HiSilicon IP Camera	<none>	Vivotek IP Camera	zlxx.	Unknown
klv123	HiSilicon IP Camera				

Amplification

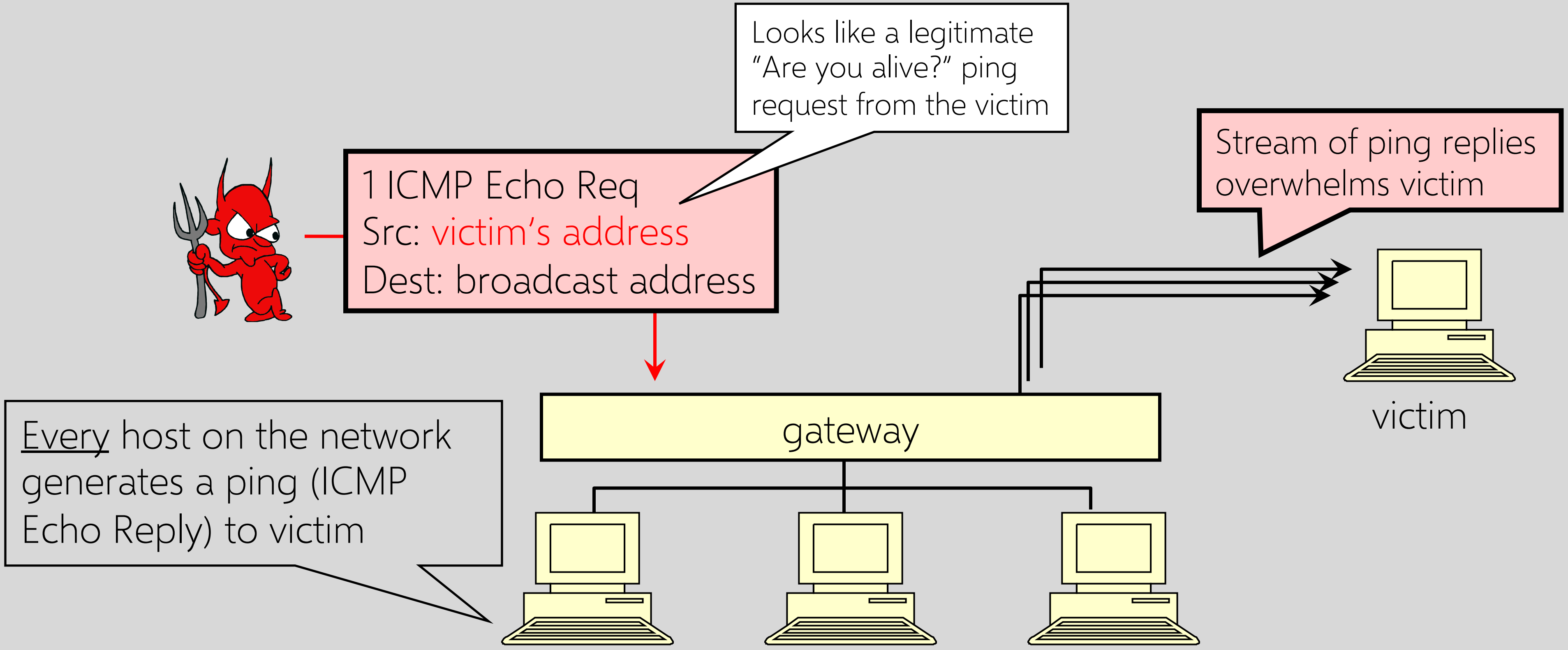
Key element of many powerful DoS attacks

Achieves attacker resources >>> victim's

- 1 request sent by attacker => N requests received by the victim
- 1 byte sent by attacker => N bytes received by the victim (N can be 200+ in some attacks)
- Force victim to expend computation, logical resources

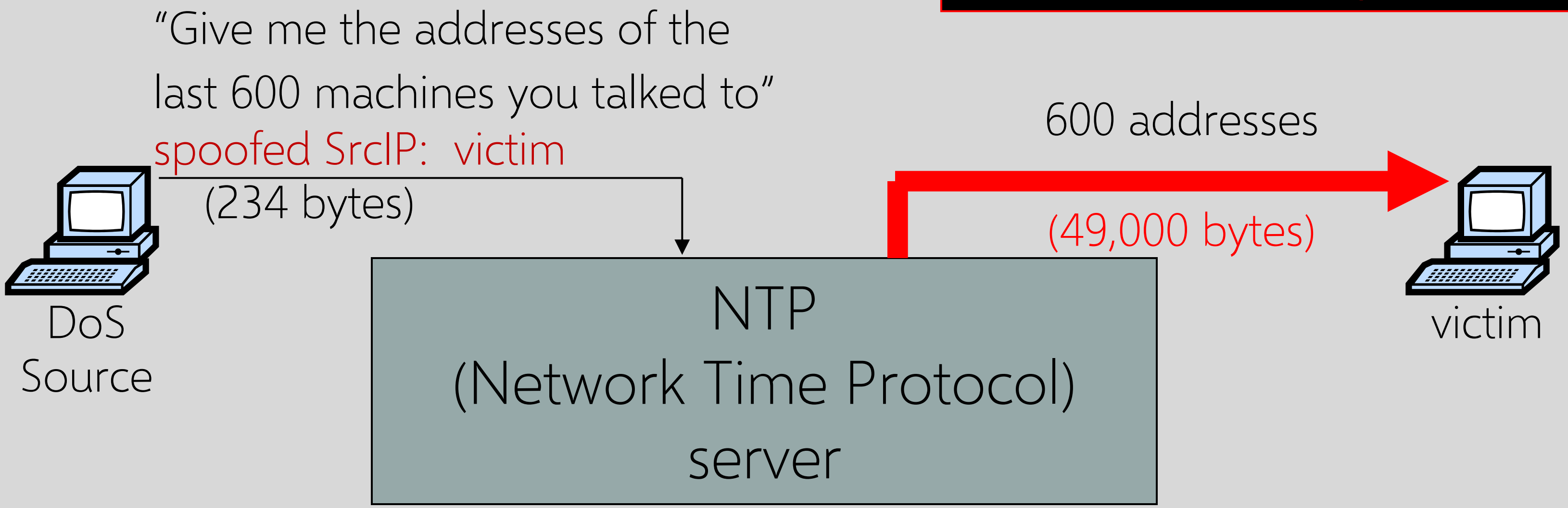
"Smurf" Reflector Attack

Solution: reject external packets to broadcast addresses



NTP Reflection + Amplification

x206 amplification



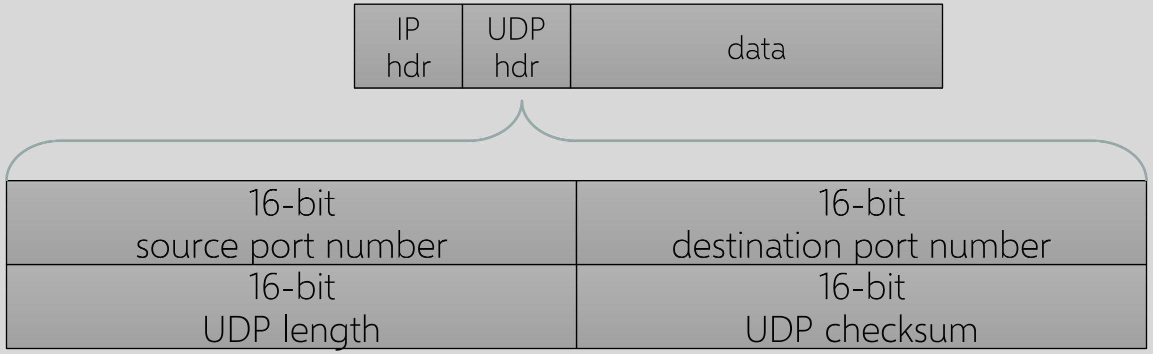
December 2013 – February 2014:
400 Gbps DDoS attacks involving 4,529 NTP servers

7 million unsecured NTP servers on the Internet (Arbor)

UDP in Reflection Attacks

DNS, memcached, … application-layer protocols running on UDP are often exploited in DoS attacks

Single packet to victim service yields response, so spoofing + reflection works

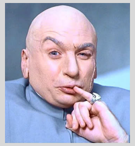


length = header len + data len

Memcached DDoS Attacks

Memcached is a popular in-memory data store

Supports UDP requests

Default configuration: accept UDP requests from anywhere 

Standard reflector attack

- Insert data into the Memcached server
- Send UDP requests with source IP addr of victim

DDoS Attack on GitHub (Feb 2018)

- 51,000x bandwidth amplification 
- 1.3 TB/s of traffic to GitHub from 1000+ ASes
- GitHub offline for 5 minutes
- Response ?
  - Use BGP announcement to route GitHub traffic through Akamai
  - Akamai gives more capacity + helps filter out bogus requests
  - Turn off UDP support in Memcached (now off by default)

https://github.blog/2018-03-01-ddos-incident-report/

Largest European DDoS Attack on Record

Craig Sparling
July 27, 2022

“

On Thursday, July 21, 2022, Akamai detected and mitigated the largest DDoS attack ever launched against a European customer on the Prolexic platform.

Victim: an Akamai customer in Eastern Europe

Source: a highly-sophisticated, global botnet of compromised devices

75 attacks over 30 days using UDP, UDP fragmentation, ICMP flood, RESET flood, SYN flood, TCP anomaly, TCP fragment, PSH ACK flood, FIN push flood, and PUSH flood. UDP most popular.

Peak rates: 854 Gbps and 660M packets/sec

Same customer attacked again on Sep 22: 705M packets/sec

https://www.akamai.com/blog/security/largest-european-ddos-attack-ever

Security News This Week: The Biggest DDoS Attack in History Hit Russian Tech Giant Yandex

Wired (September 11, 2021)

A massive botnet, dubbed Mēris, is believed responsible, flooding Yandex with millions of HTTP requests for webpages at the same time.

This DDoS technique is called HTTP pipelining, where a browser requests a connection to a server and, without waiting for a response, sends multiple more requests. Those requests reportedly originated from networking gear made by MikroTik. Attackers, according to Qrator Labs, exploited a 2018 bug unpatched in more than 56,000 MikroTik hosts involved in the DDoS attack.

The Mēris botnet delivered the largest attack against Yandex it has ever spotted (by traffic volume) – peaking at 21.8 million requests per second (RPS).

https://threatpost.com/yandex-meris-botnet/169368/

Record DDoS Attack with 25.3 Billion Requests Abused HTTP/2 Multiplexing

June 27, 2022

Target: a Chinese telecom company

Method: HTTP/2 multiplexing (multiple packets in one) from a botnet of 170,000 different routers, security cameras, compromised servers in over 180 countries

Peak rate: 3.9 million requests per second

https://thehackernews.com/2022/09/record-ddos-attack-with-253-billion.html

Other Countermeasures

Above Transport Layer	**Kerberos** • Provides authentication, protects against application-layer spoofing • Does not protect against connection hijacking
Above Network Layer	**SSL/TLS and SSH** • Protects against connection hijacking and injected data • Does not protect against DoS by spoofed packets
Above Network Layer	**IPsec** • Protects against hijacking, injection, DoS using connection resets, IP address spoofing