Network Security

Soheil Abbasloo
Department of Computer Science
University of Toronto

Credit for some slides goes to **Dan Boneh** @ Stanford

Fall 2022

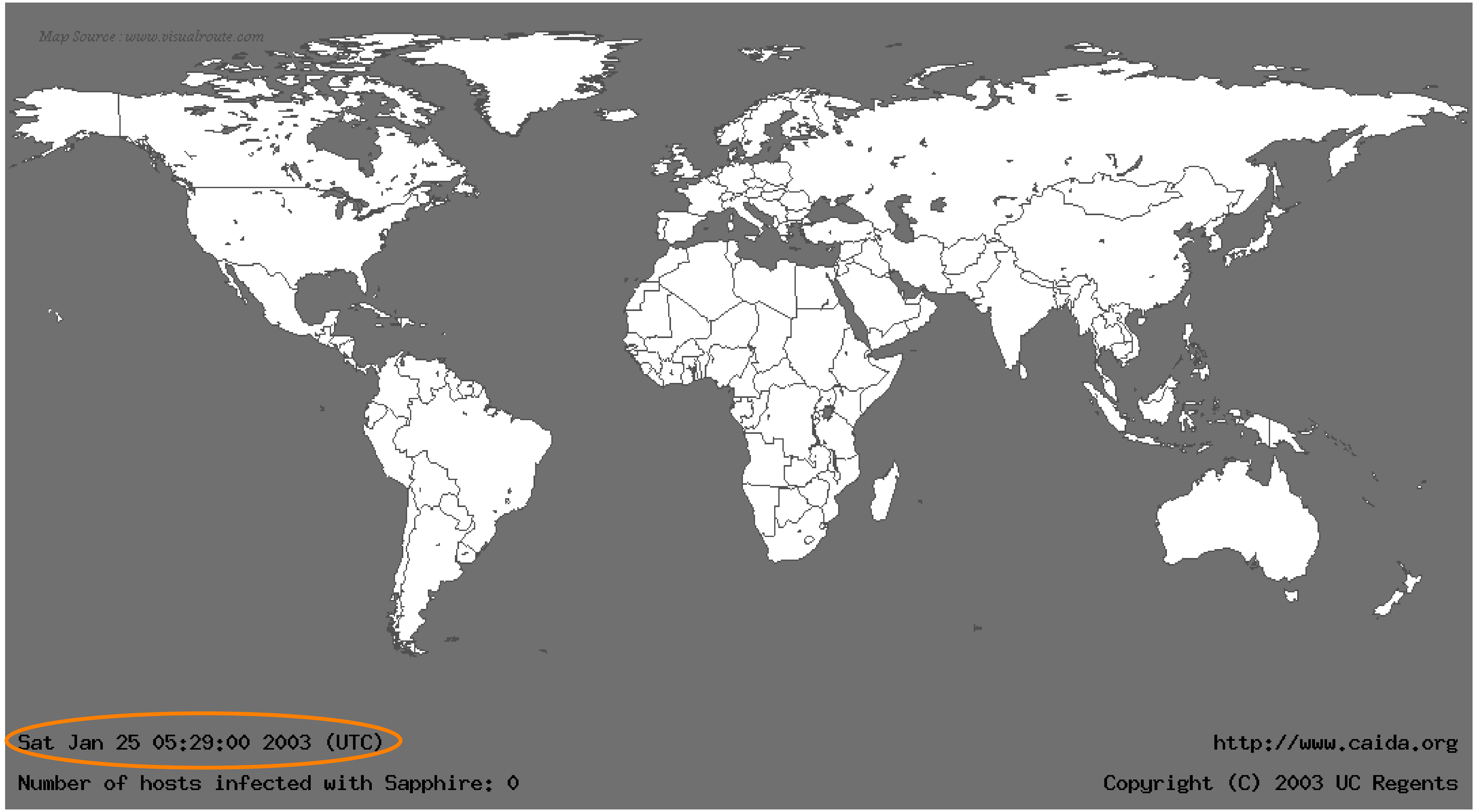
Outline

- Why Security?
- Internet Design vs. Security
- Attacks
  - DoS
  - Amplification
- Defenses

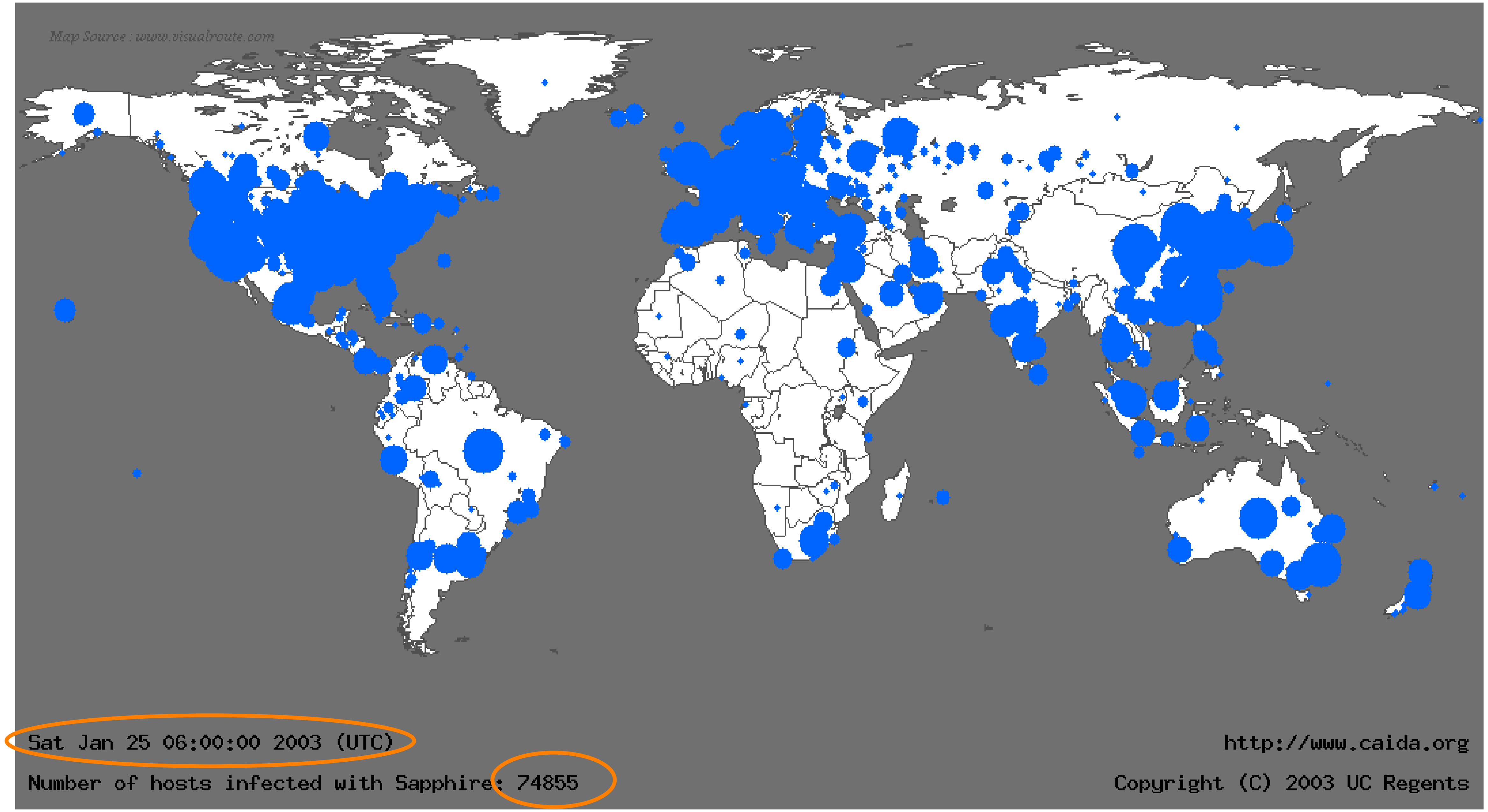
Connectivity: Good vs. Evil

- Network have improved significantly
  - In terms of bandwidth and latency
- **Good:** We can communicate
  - Exchange information
  - Transfer data
  - …
- **Evil:** It’s easier to do harm
  - Harmful code can propagate faster
  - Information collection, violating privacy
  - …

Life Just Before SQL Slammer



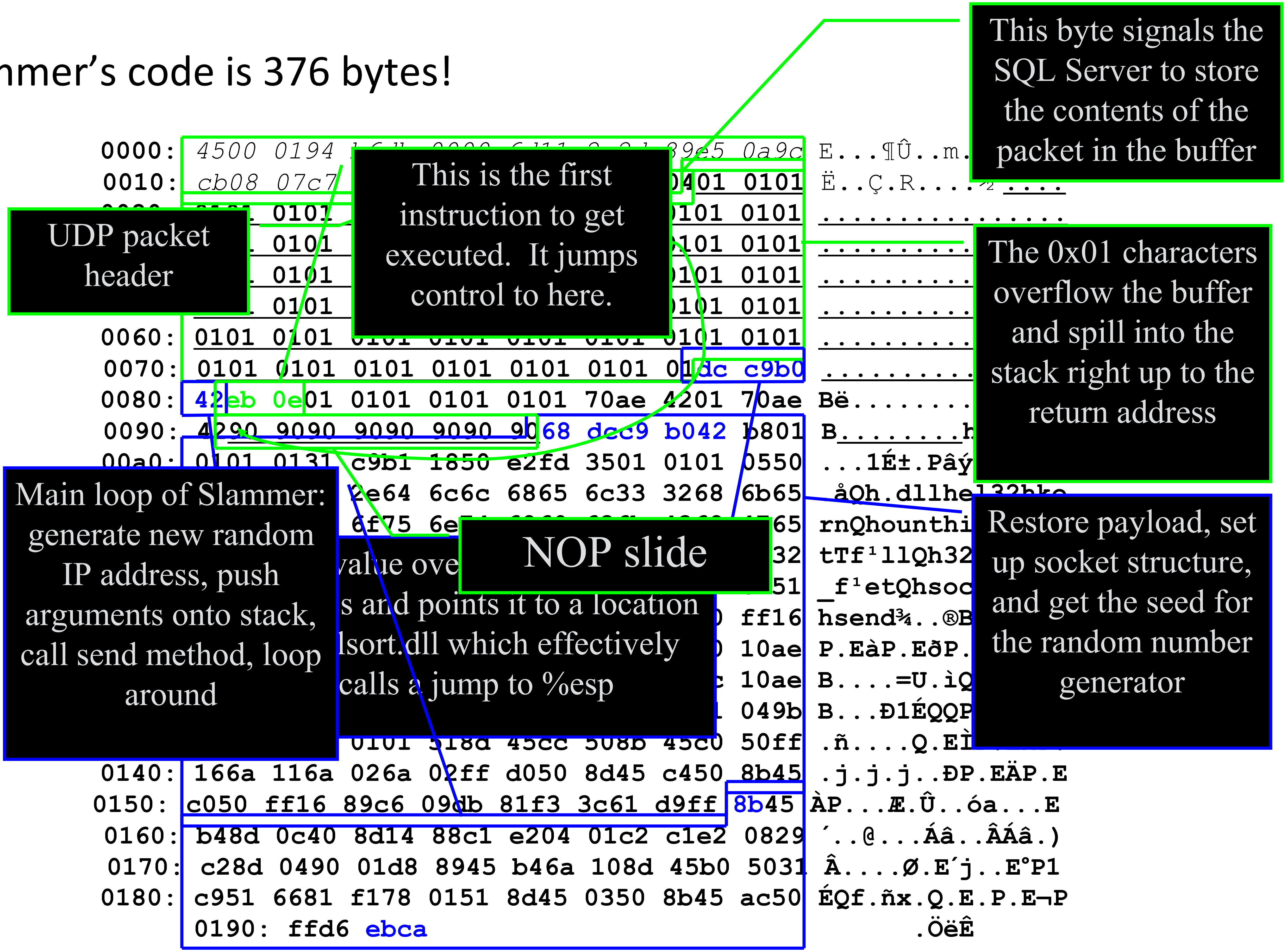
Life Just After SQL Slammer



SQL Slammer

- Affects Microsoft SQL 2000
- Exploits known buffer overflow vulnerability
  - Server Resolution service vulnerability reported June 2002
  - Patched released in July 2002 Bulletin MS02-39
- Vulnerable population (75,000+) infected in less than 10 minutes
  - At its peak, doubled every 8.5 seconds.
- Entire worm fits in a single packet! (376 bytes)

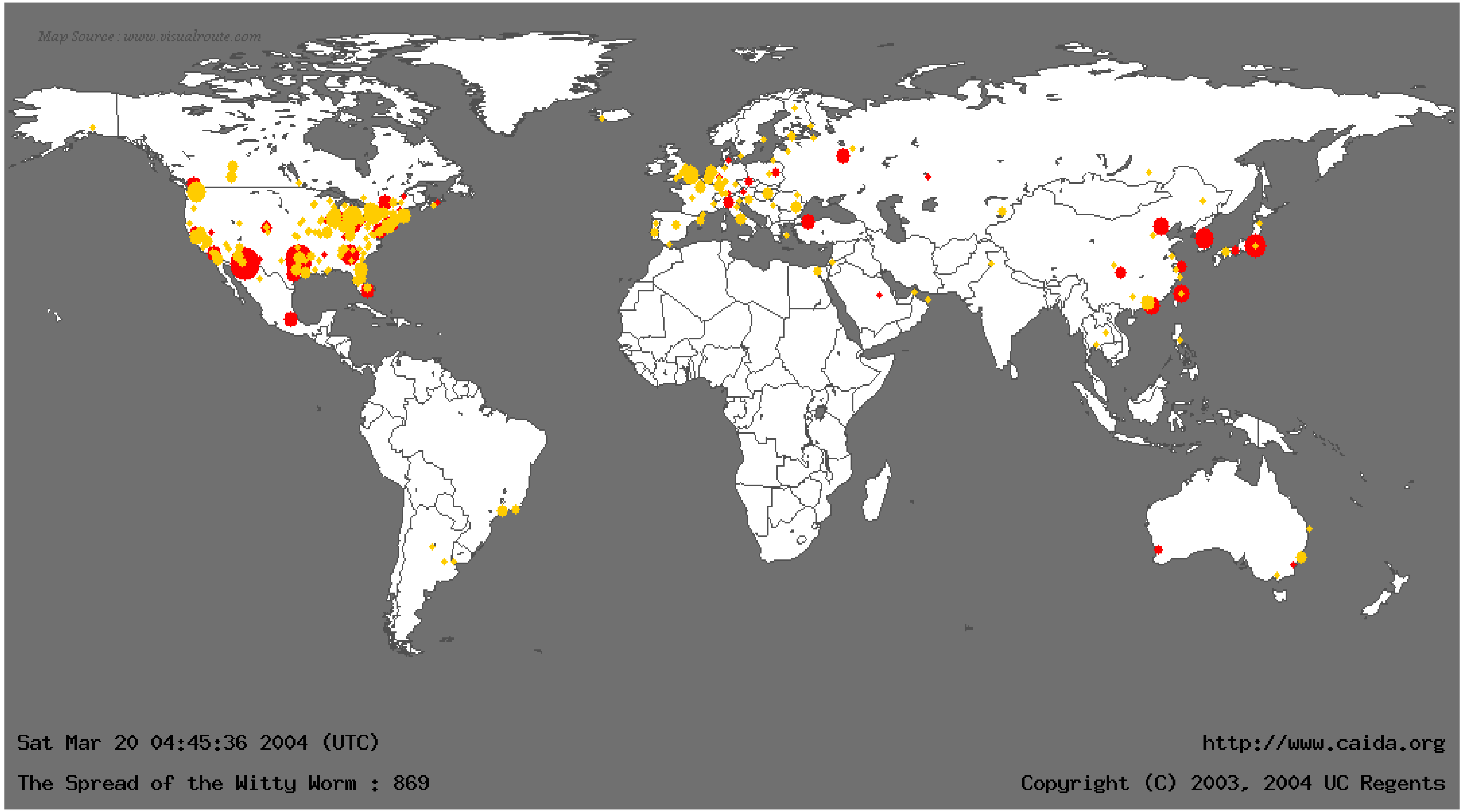
Slammer’s code is 376 bytes!



Why Security?

- First victim at 12:45 am
- By 1:15 am, transcontinental links starting to fail
- 300,000 access points downed in Portugal
- All cell and Internet in Korea failed (27 million people)
- Five root name servers were knocked offline
- 911 didn’t respond (Seattle)
- Flights canceled!

Witty Worm


Map Source : www.visualroute.com
Sat Mar 20 04:45:36 2004 (UTC)
The Spread of the Witty Worm : 869
http://www.caida.org
Copyright (C) 2003, 2004 UC Regents

Witty Worm

- Attacks firewalls and security products (of ISS {Internet Security Systems} company)
- First to use vulnerabilities in security software
- ISS announced a vulnerability
  - buffer overflow problem
  - Attack in just one day!
- Attack started from a small number of compromised machines
- In 30 minutes, 12,000 infected machines
  - 90 Gb/s of UDP traffic

Top 10 products by total number of “distinct” vulnerabilities in 2022

	Product Name	Vendor Name	Product Type	Number of Vulnerabilities
1	Debian Linux	Debian	OS	7331
2	Android	Google	OS	4707
3	Fedora	Fedoraproject	OS	3988
4	Ubuntu Linux	Canonical	OS	3680
5	Mac Os X	Apple	OS	3100
6	Linux Kernel	Linux	OS	3000
7	Windows 10	Microsoft	OS	2990
8	Iphone Os	Apple	OS	2820
9	Windows Server 2016	Microsoft	OS	2764
10	Chrome	Google	Application	2554
26	Mysql	Oracle	Application	1182
27	Internet Explorer	Microsoft	Application	1168
28	Safari	Apple	Application	1164
29	Thunderbird	Mozilla	Application	1038

https://www.cvedetails.com/top-50-products.php?year=2022

Internet Design vs. Security

Basic Security Properties

- Availability
  - Ability to use desired information/resource
- Protection
  - protect users from interactions they don't want
- Authenticity
  - Identification & assurance of origin of info
- Confidentiality
  - Concealment of information or resources
- Data Integrity
  - Trustworthiness of data/resources; preventing improper/unauthorized changes

Basic Security Properties

- Availability
  - Ability to use desired information/resource
- Protection
  - protect users from interactions they don’t want
- Authenticity
  - Identification & assurance of origin of info
- Confidentiality
  - Concealment of information or resources
- Data Integrity
  - Trustworthiness of data/resources; preventing improper/unauthorized changes

Internet Design

- Destination routing
- Packet based (statistical multiplexing)
- Global addressing (IP addresses)
- Simple to join (as infrastructure)
- Power at end hosts (end-to-end argument)

Internet Design vs. Security

- Destination routing
  - Keeps forwarding tables small
  - Simple to maintain forwarding tables
  - How do we know where packets are coming from?
    - Probably simple fix to spoofing, why isn't it in place?
- Packet based (statistical multiplexing)
- Global addressing (IP addresses)
- Simple to join (as infrastructure)
- Power in end hosts (end-to-end argument)

Internet Design vs. Security

- Destination routing
- Packet Based (statistical multiplexing)
  - Simple + Efficient
  - Difficult resource bound per-communication
    - How to keep someone from hogging?
      (remember, we can’t rely on source addresses)
- Global Addressing (IP addresses)
- Simple to join (as infrastructure)
- Power in End Hosts (end-to-end argument)

Internet Design vs. Security

- Destination routing
- Packet based (statistical multiplexing)
- Global Addressing (IP addresses)
  - Very democratic
  - Even people who don’t necessarily want to be talked to
    - “every psychopath is your next door neighbor” – Dan Geer
- Simple to join (as infrastructure)
- Power in end hosts (end-to-end argument)

Internet Design vs. Security

- Destination routing
- Packet based (statistical multiplexing)
- Global addressing (IP addresses)
- Simple to join (as infrastructure)
  - Very democratic
  - Misbehaving routers can do very bad things
    - No model of trust between routers
- Power in End Hosts (end-to-end argument)

Internet Design vs. Security

- Destination routing
- Packet based (statistical multiplexing)
- Global addressing (IP addresses)
- Simple to join (as infrastructure)
- Power in end-hosts (end-to-end argument)
  - Decouple hosts and infrastructure = innovation at the edge!
  - Giving power to least trusted actors!
    - How to guarantee good behavior?

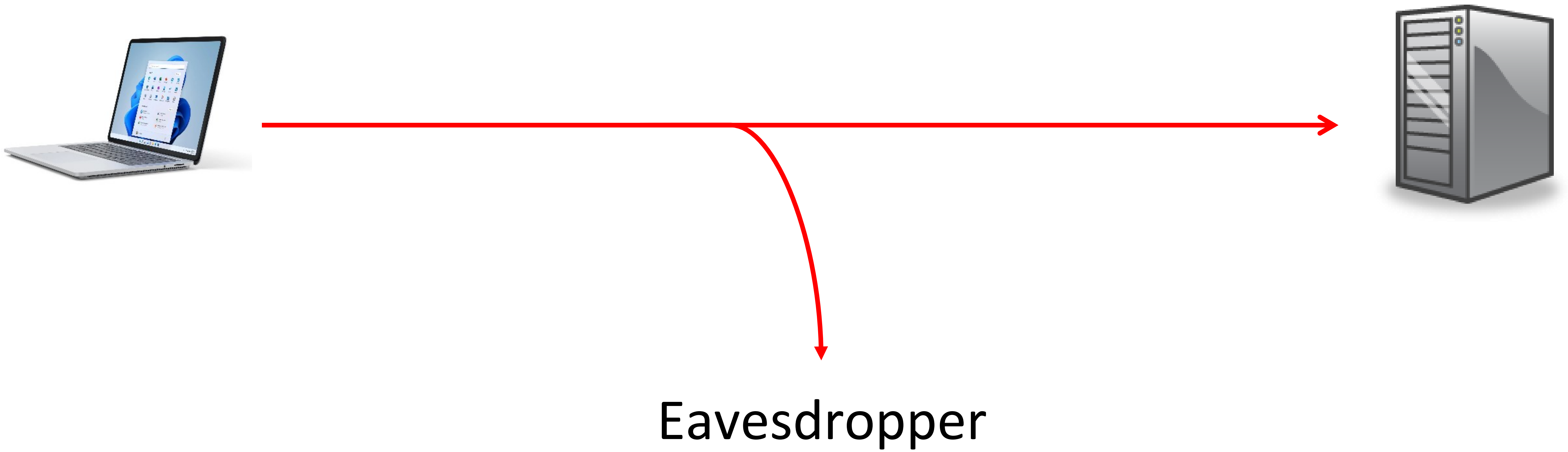
Attacks

Don’t Try These at Home!

Attack on Confidentiality

Eavesdropping - Message Interception

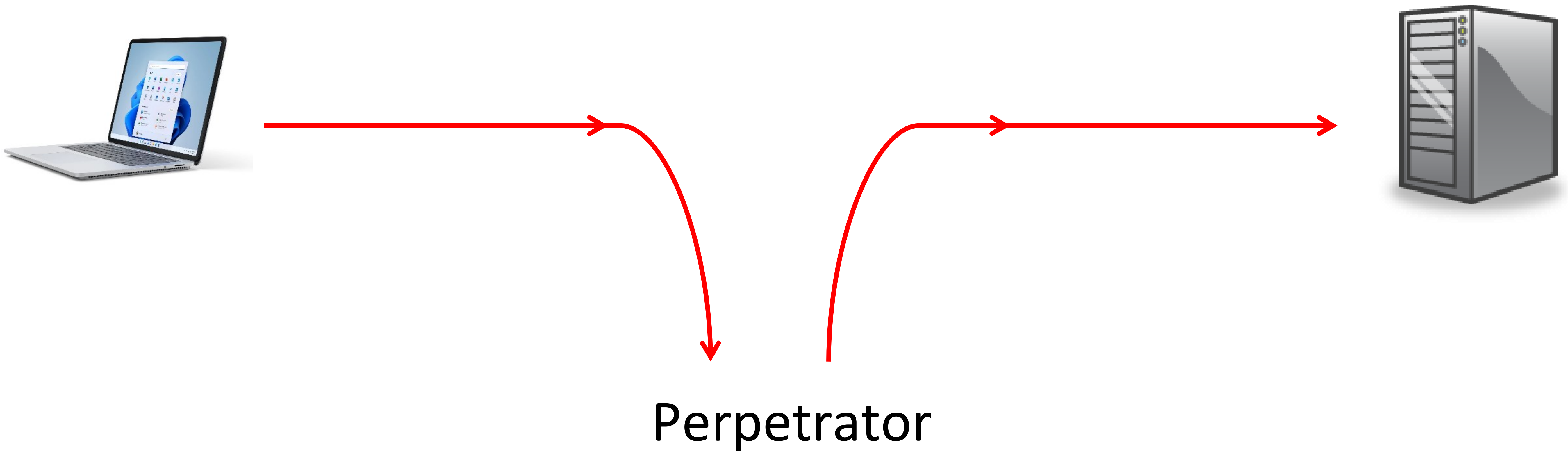
- Unauthorized access to information
- Packet sniffers and wiretappers (e.g. tcpdump)
- Illicit copying of files and programs



Attack on Integrity

Tampering

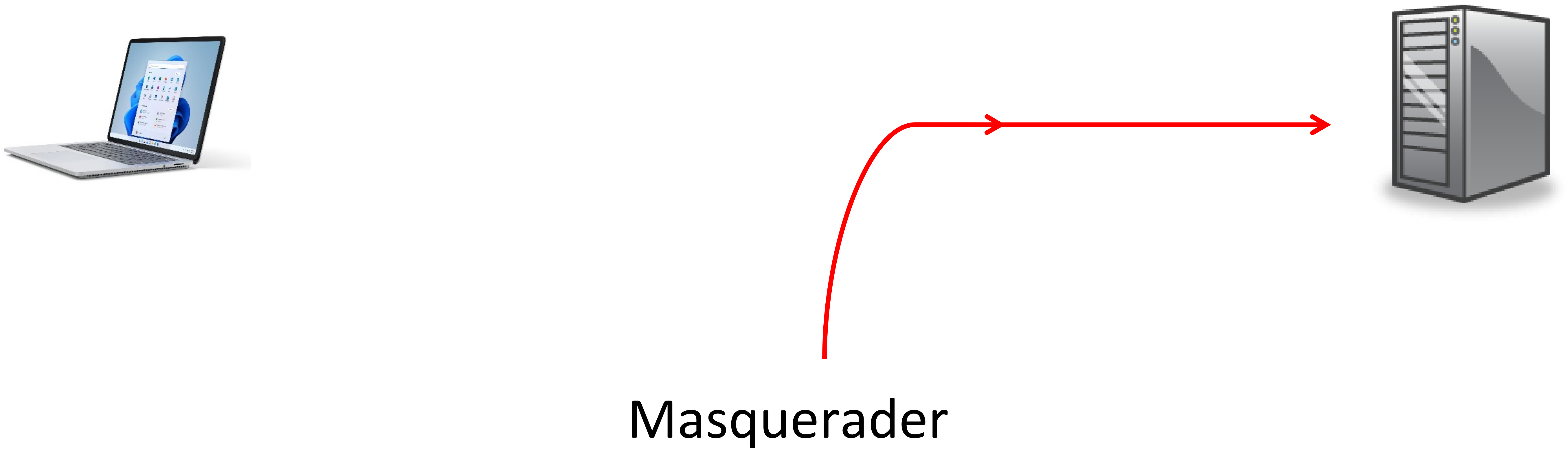
- Stop the flow of the message
- Delay and optionally modify the message
- Release the message again



Attack on Authenticity

Fabrication

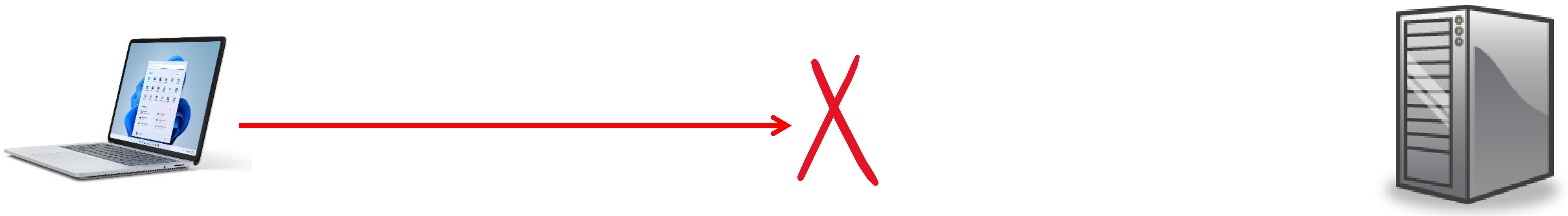
- Unauthorized assumption of other’s identity
- Generate and distribute objects under identity



Attack on Availability

Denial of Service

- Destroy hardware (cutting fiber) or software
- Modify software in a subtle way
- Corrupt packets in transit

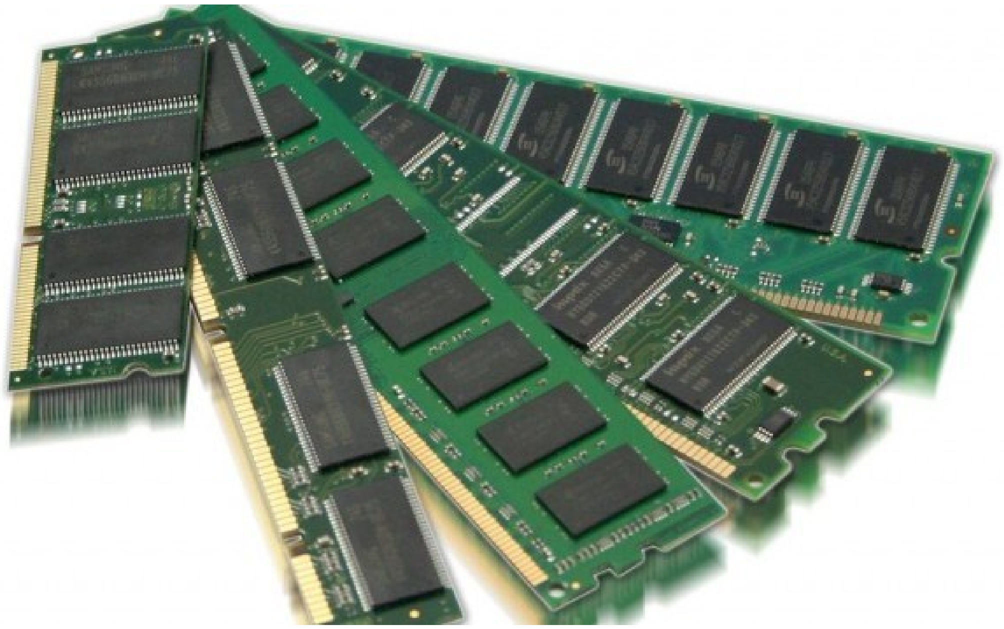


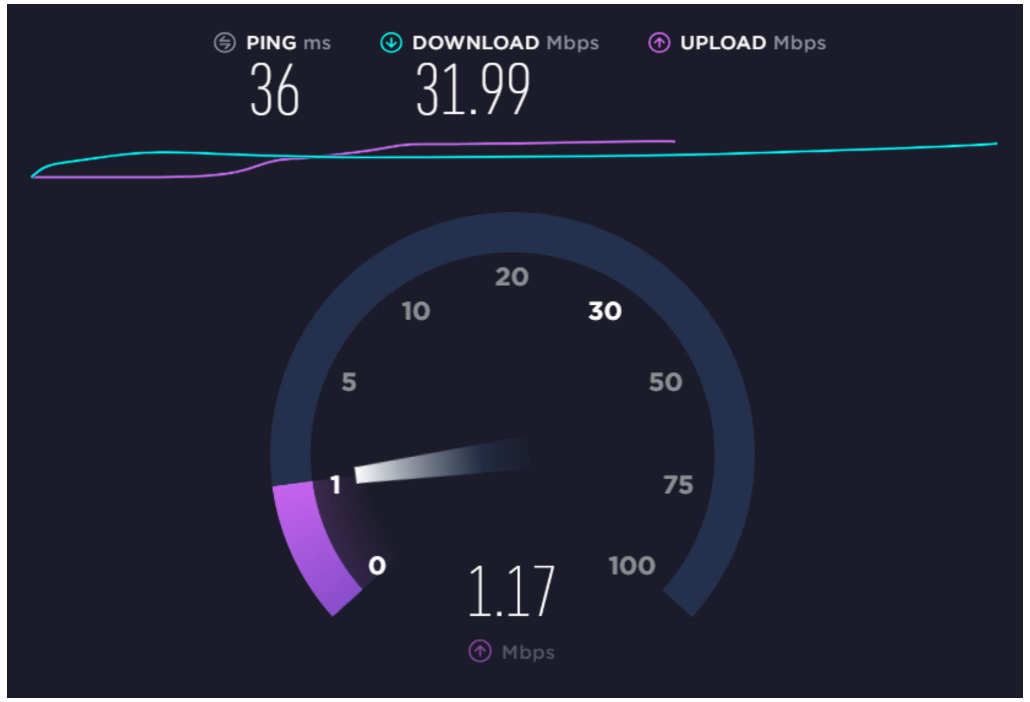
- Blatant denial of service (DoS):
  - Crashing the server
  - Overwhelm the server (use up its resource)

Denial of Service (DoS)

Via Resource Exhaustion

- CPU
- Bandwidth
- Memory
  - E.g., TCP connections require state





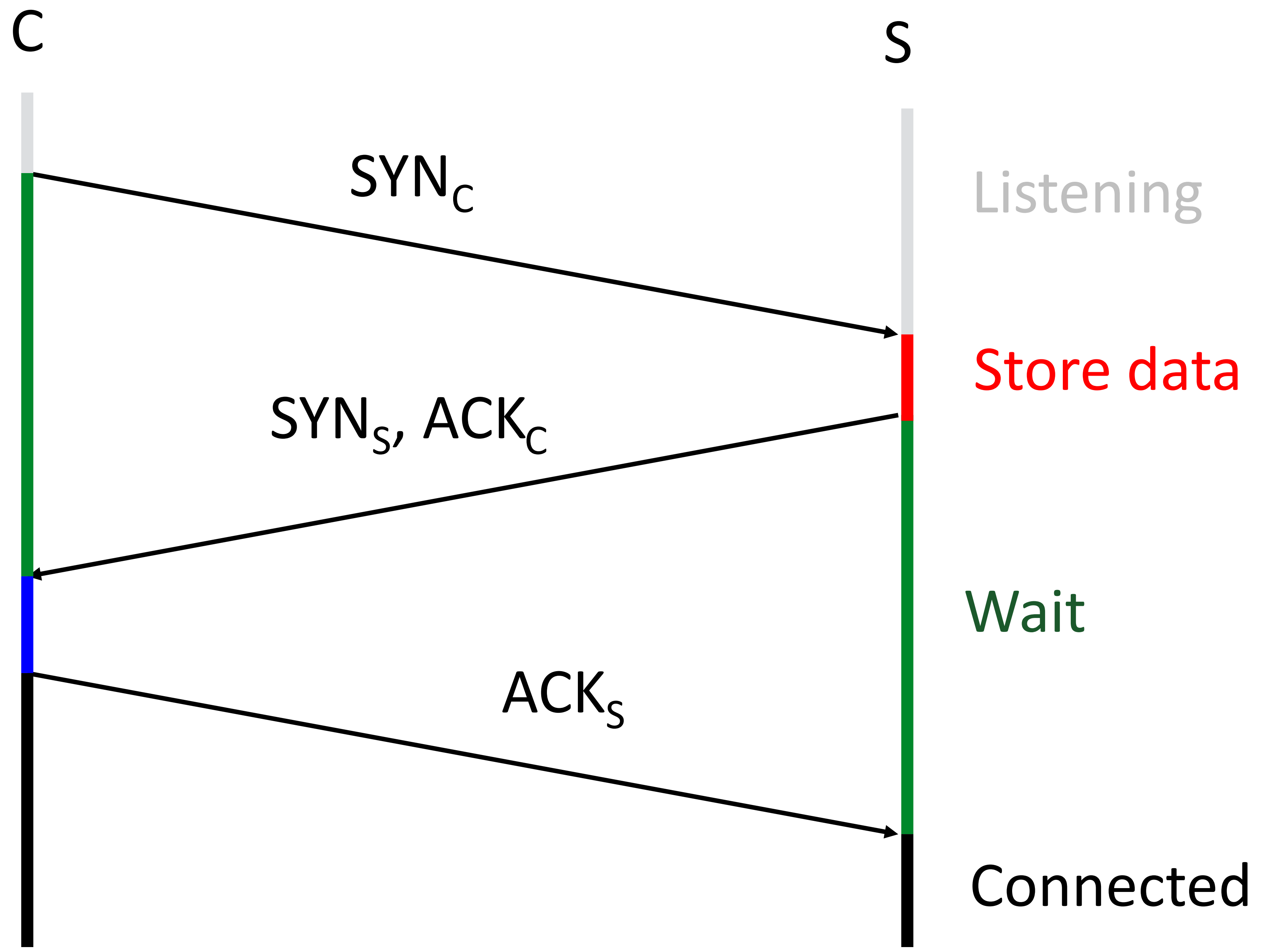
DoS Attacks

- **Goal:** take large site offline by overwhelming it with network traffic such that they can’t process real requests
- **How:** find mechanism where attacker doesn’t have to spend a lot of effort, but requests are difficult/expensive for victim to process

Possible at Every Layer!

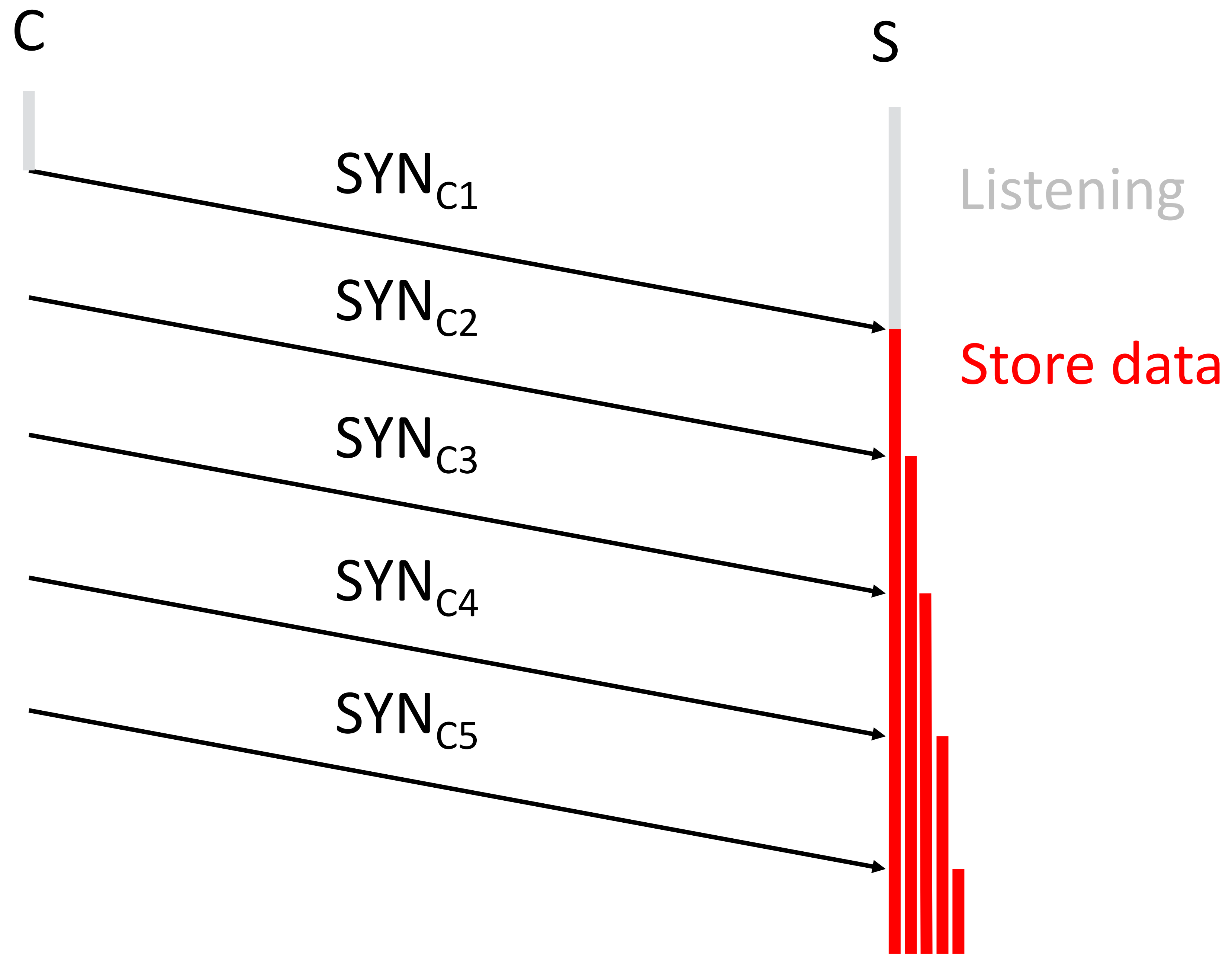
- **Link Layer:** send too much traffic for switches/routers to handle
- **TCP/UDP:** require servers to maintain large number of concurrent connections or state
- **Application Layer:** require servers to perform expensive queries or cryptographic operations

TCP Handshake



TCB (transmission control block) contains information about the connection state per connection

Example: SYN Flooding



- Single machine:
  - SYN packets with random source IP addr
  - Fills up backlog queue
  - No further connection possible!

How to resolve that?!

Core Problem

- **Problem:** server commits resources (memory) before confirming identify of client (when client responds)
- **Bad Solution:**
  - Increase backlog queue size
  - Decrease timeout
- **Real Solution:** Avoid state until 3-way handshake completes!
- A useful strategy to remember!

Protection against SYN Attacks

Bernstein, Schenk’1996

- Don't create the TCB until the ACK comes back!
- Use SYN Cookies
  - Make a one-way hash of the incoming information!

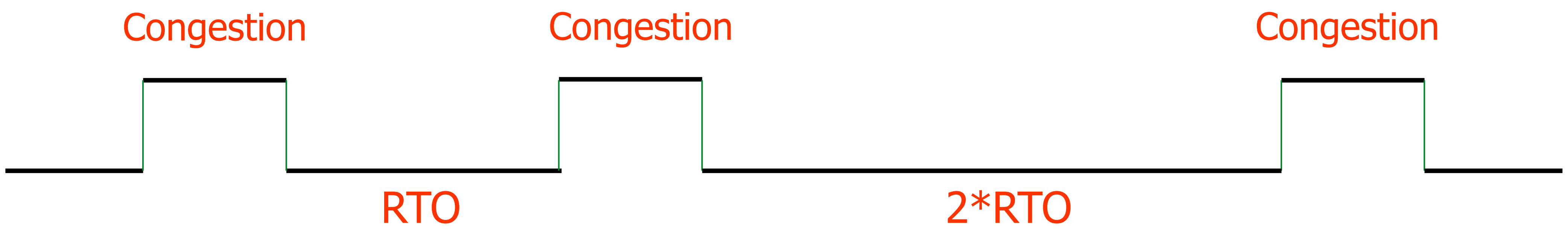
- Client sends SYN
- Server responds to Client with SYN-ACK cookie
    - sqn = f(src addr, src port, dest addr, dest port, key, rand(time))
    - Server does not save state
  - Honest client responds with ACK(sqn)
  - Server checks response
  - If matches SYN-ACK, establishes connection **and allocates space**

See if it is enabled on your system! sysctl net.ipv4.tcp_syncookies

DoS

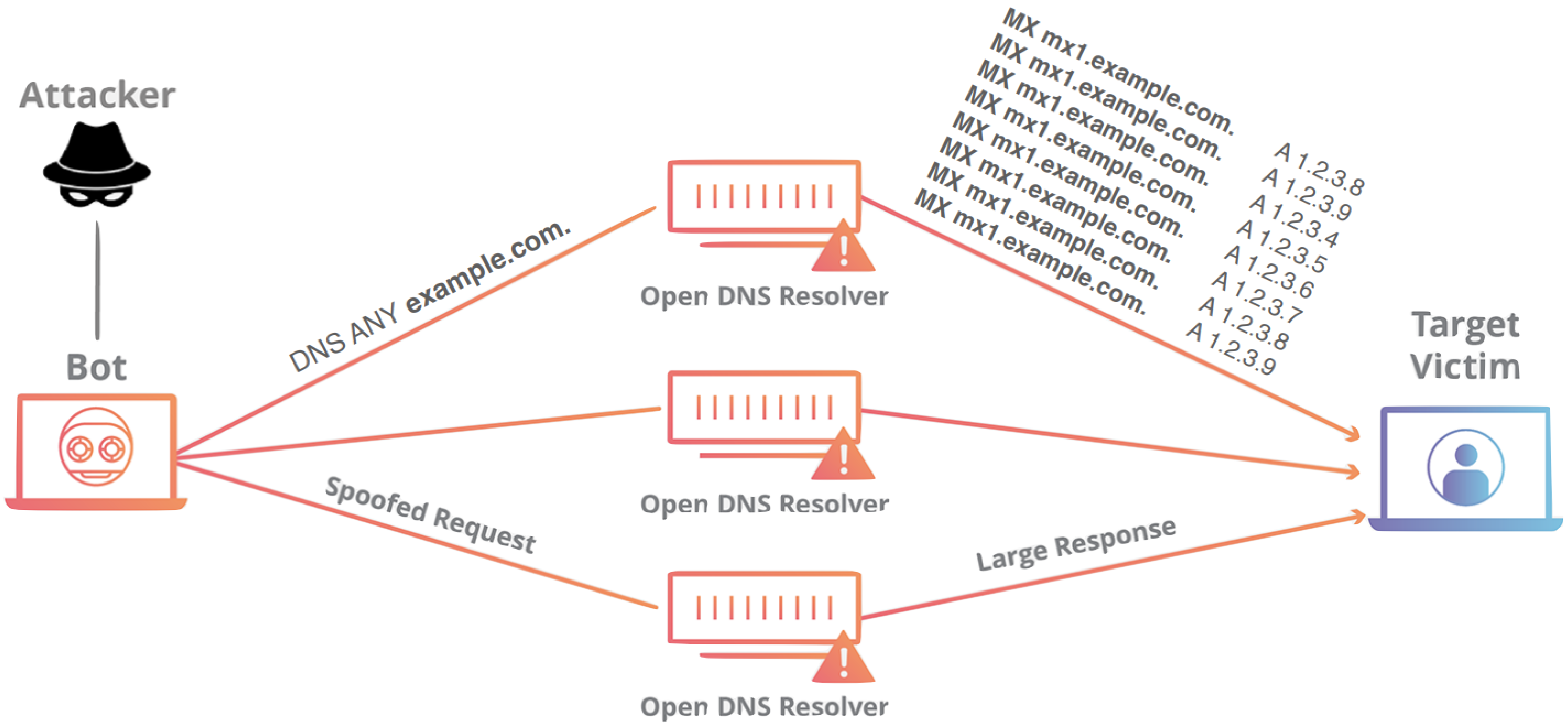
Another Example

- Congestion control DoS attack!



  - Generate TCP flow to force target to repeatedly enter retransmission timeout state
  - Difficult to detect because packet rate is low

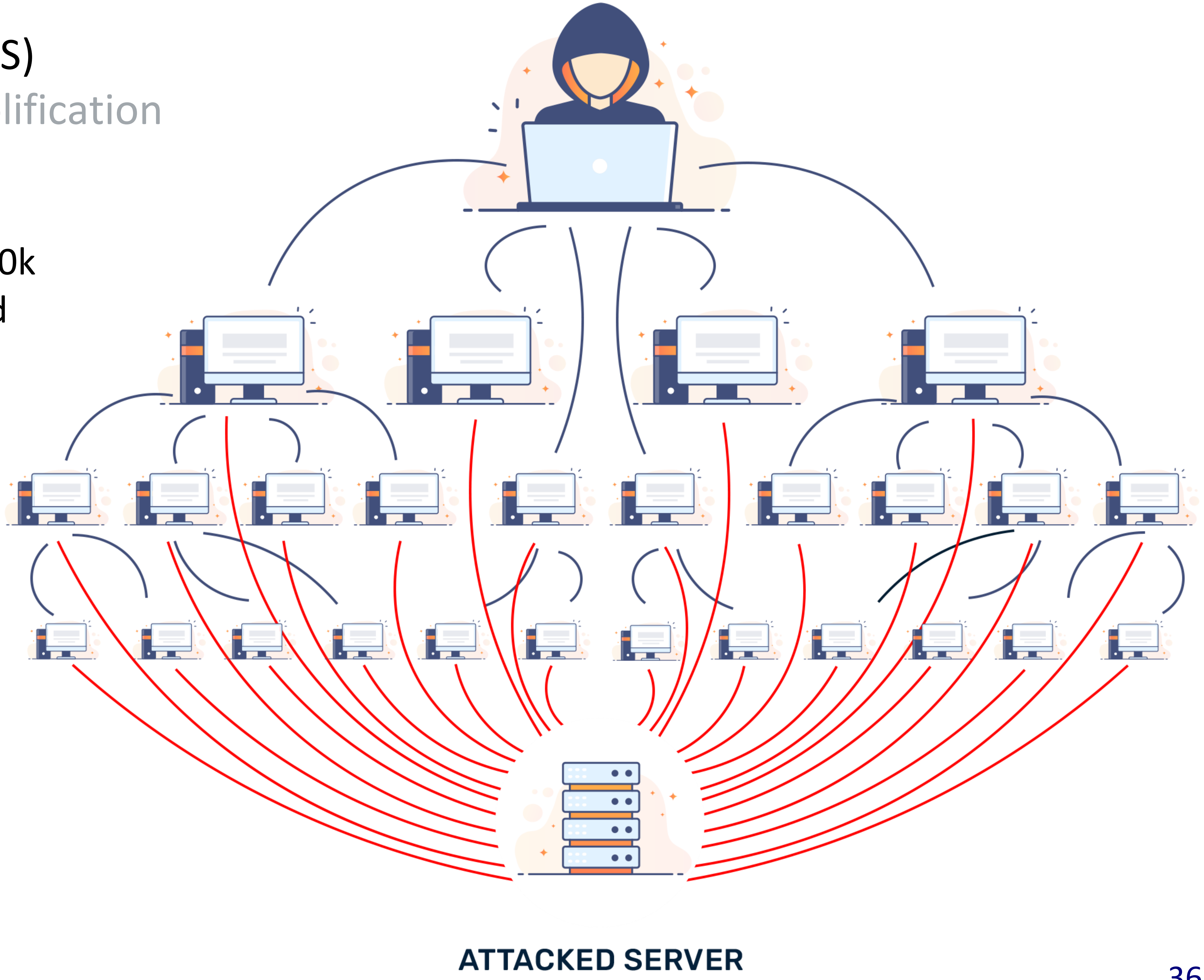
Amplification Attacks



Distributed DoS (DDoS)

Another form of Amplification

Bot-networks of 80k to 100k have been seen in the wild



Amplification Attacks

A simple example!

- TCP SYN-ACK amplification!

- Attacker pretends to be from the target's network IP address
- Sends SYN packet to a preselected reflection IP addresses or services

- Now, lots of SYN-ACK packet go toward the target network!

- Target hosts unaware of these connections, drop the SYN-ACKs
- But the retransmission of the SYN-ACKs continuous!

Common UDP Amplifiers

- **DNS:** ANY query returns all records server has about a domain
- **NTP:** MONLIST (a debugging command!) returns list of last 600 clients who asked for the time recently
- Only works if you can receive a big response by sending a single packet
  - otherwise spoofing doesn't help you!

DNS: Domain Name Server
NTP: Network Time Protocol

Amplification Attacks

- **2013:** DDoS attack generated 300 Gbps (DNS)
  - 31,000 misconfigured open resolvers, each at 10 Mbps
  - Source: 3 networks that allowed IP spoofing
- **2014:** 400 Gbps DDoS attacked used 4500 NTP servers

THE WALL STREET JOURNAL.

October 21, 2016

Cyberattack Knocks Out Access to Websites

Popular sites such as Twitter, Netflix and PayPal were unreachable for part of the day

twitter amazon web services PayPal NETFLIX SOUNDCLOUD

Spotify GitHub reddit New York Times

https://www.wsj.com/articles/denial-of-service-web-attack-affects-amazon-twitter-others-1477056080

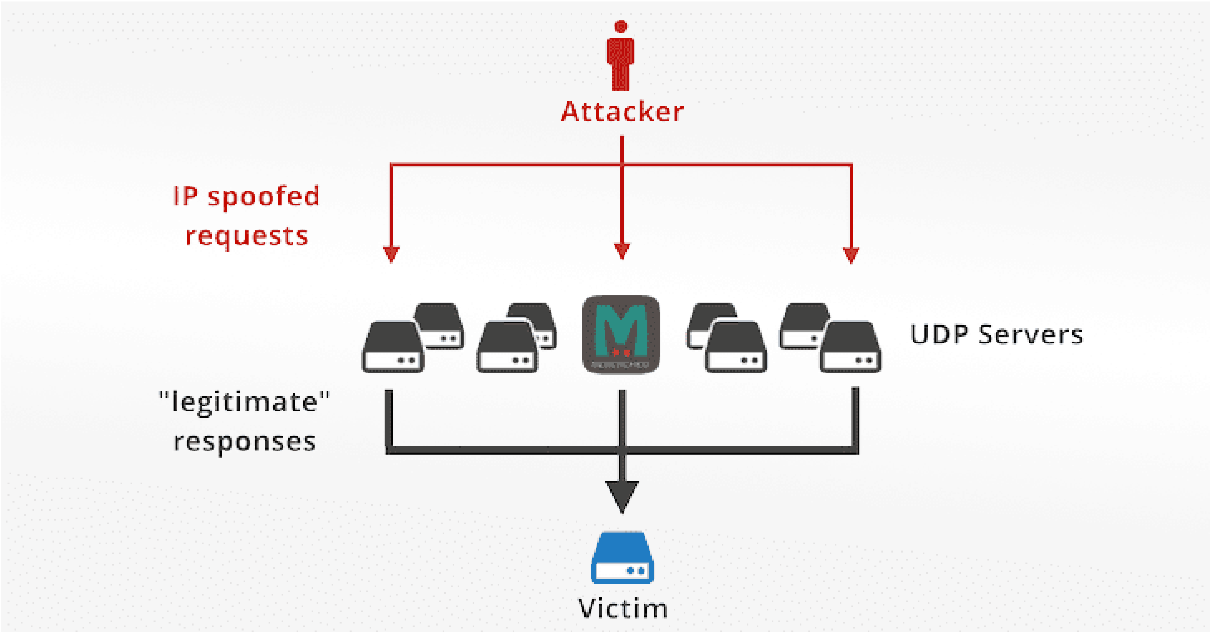
Cause:
DNS Amplification Attack

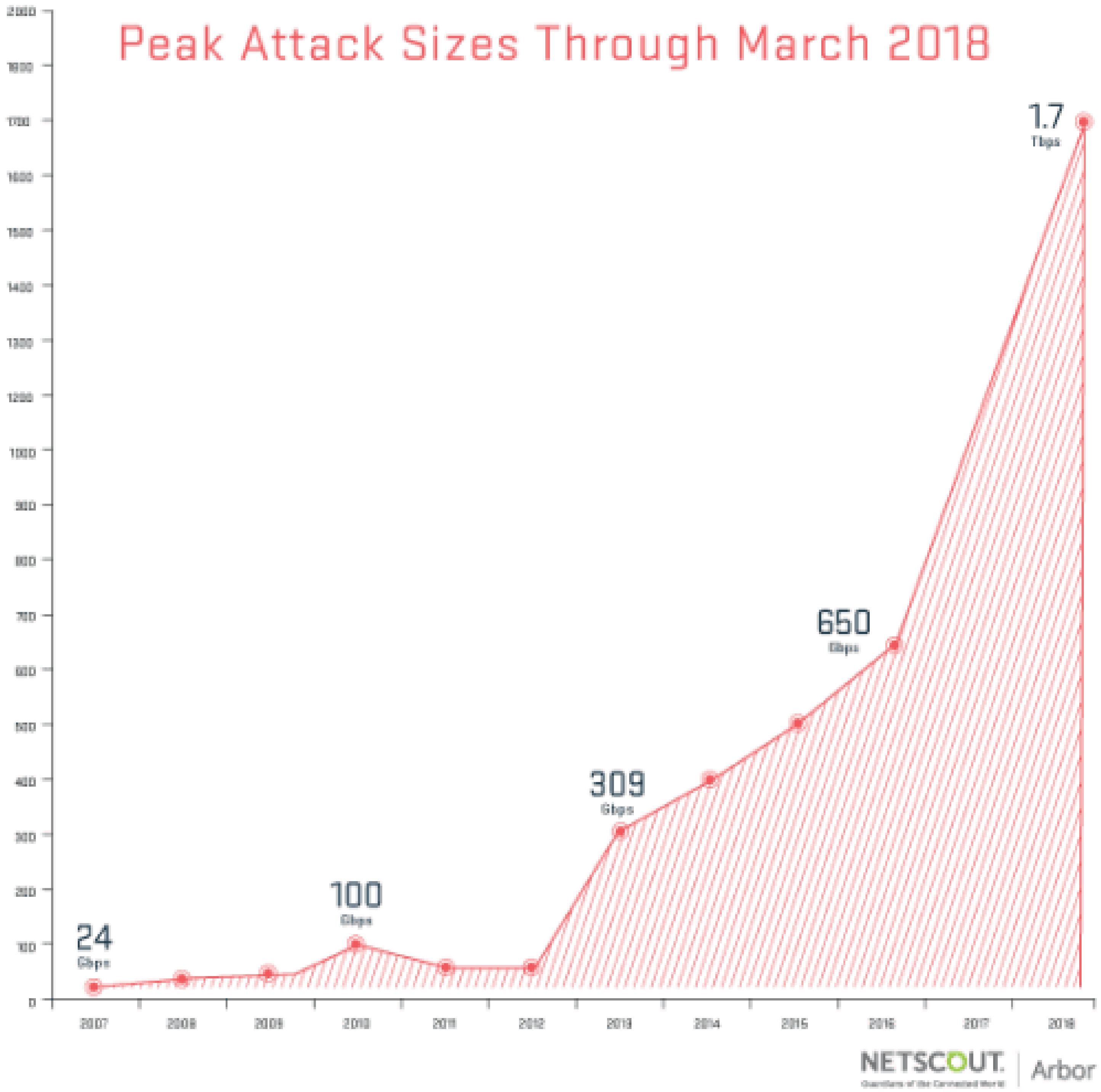
- A Botnet of IoT Devices
- Attack came from "tens of millions" of addresses on infected machines (with **Mirai** malware)
- Caused an attack with a magnitude of 1.2 Tbps!

Amplification Attacks
Memcached

- Record of amplification!
  - 1.7 Tbps amplification attack

- An amplification by a factor of 51,000 using thousands of misconfigured Memcached servers exposed on the Internet





Moving Up Stack: GET Floods

- Command bot army to:
  - Complete real TCP connection
  - Complete TLS Handshake
  - GET large image or other content
- Will bypass flood protections…. but attacker can no longer use random source IPs
- Victim site can block or rate limit bots

Reconnaissance

- To attack a victim, first discover available resources
- Many commonly used reconnaissance techniques
  - Port scanning
  - Host/application fingerprinting
  - Traceroute
  - DNS (reverse DNS scanning, Zone transfer)
  - SNMP
- These are meant for use by admins to diagnose network problems!
  - Trade-off between the ability to diagnose a network and reveal security sensitive information

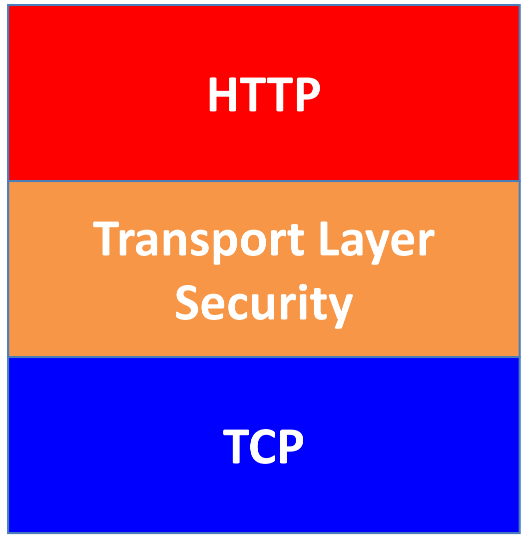
Anecdotes …

- Large bot networks exist that scan the Internet daily looking for vulnerable hosts
- Old worms still endemic on Internet
  - Seem to come and go in mass
  - Surreptitious scanning effort?

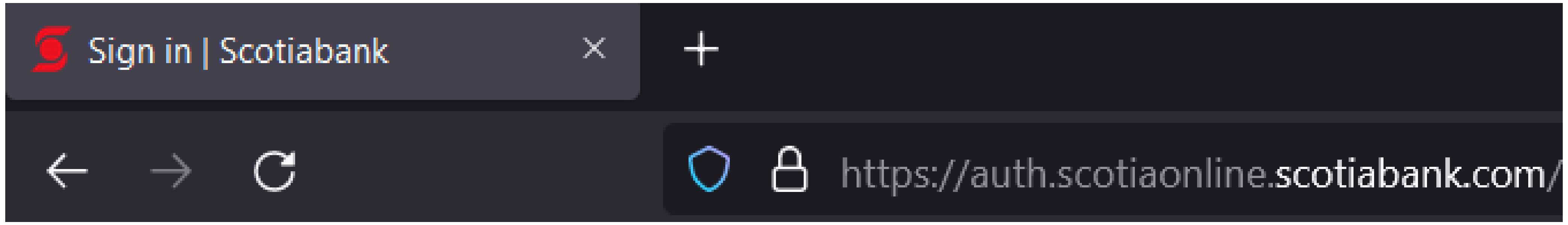
Network Defenses

Assume network is malicious!

- The network is out to get you!
- **Solution:** Always use TLS if you want any protection against large-scale eavesdropping (e.g., intelligence agencies), or guarantee that data hasn't been modified or corrupted by an on-path attacker
- **Note!** HTTPS and TLS aren't just for sensitive material! There have been attacks where malicious Javascript or malware is injected into websites.
  - E.g., 1.35 Tbps attack against Github

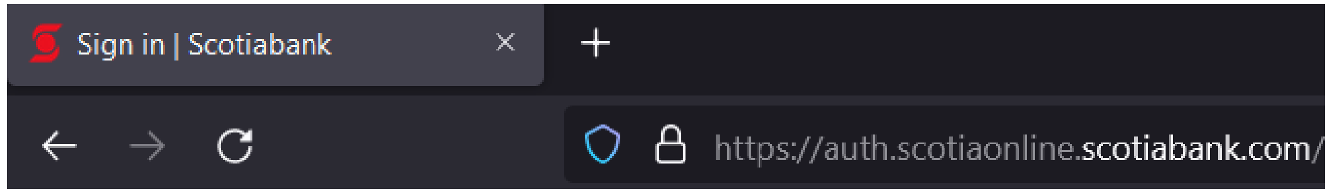


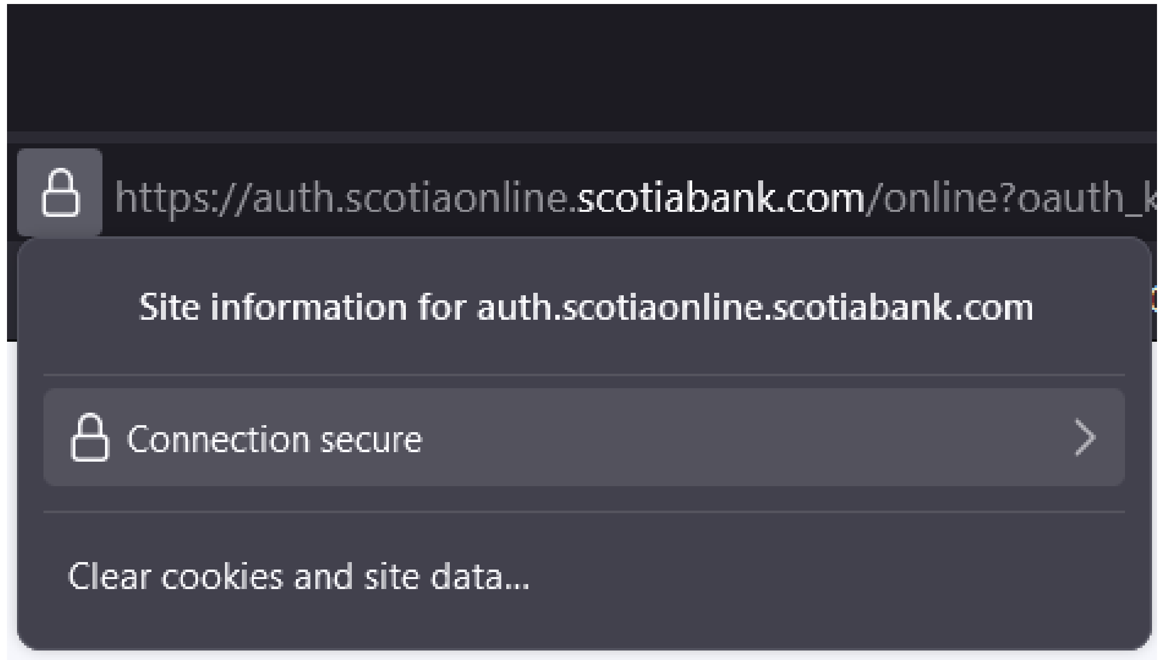
HTTPS

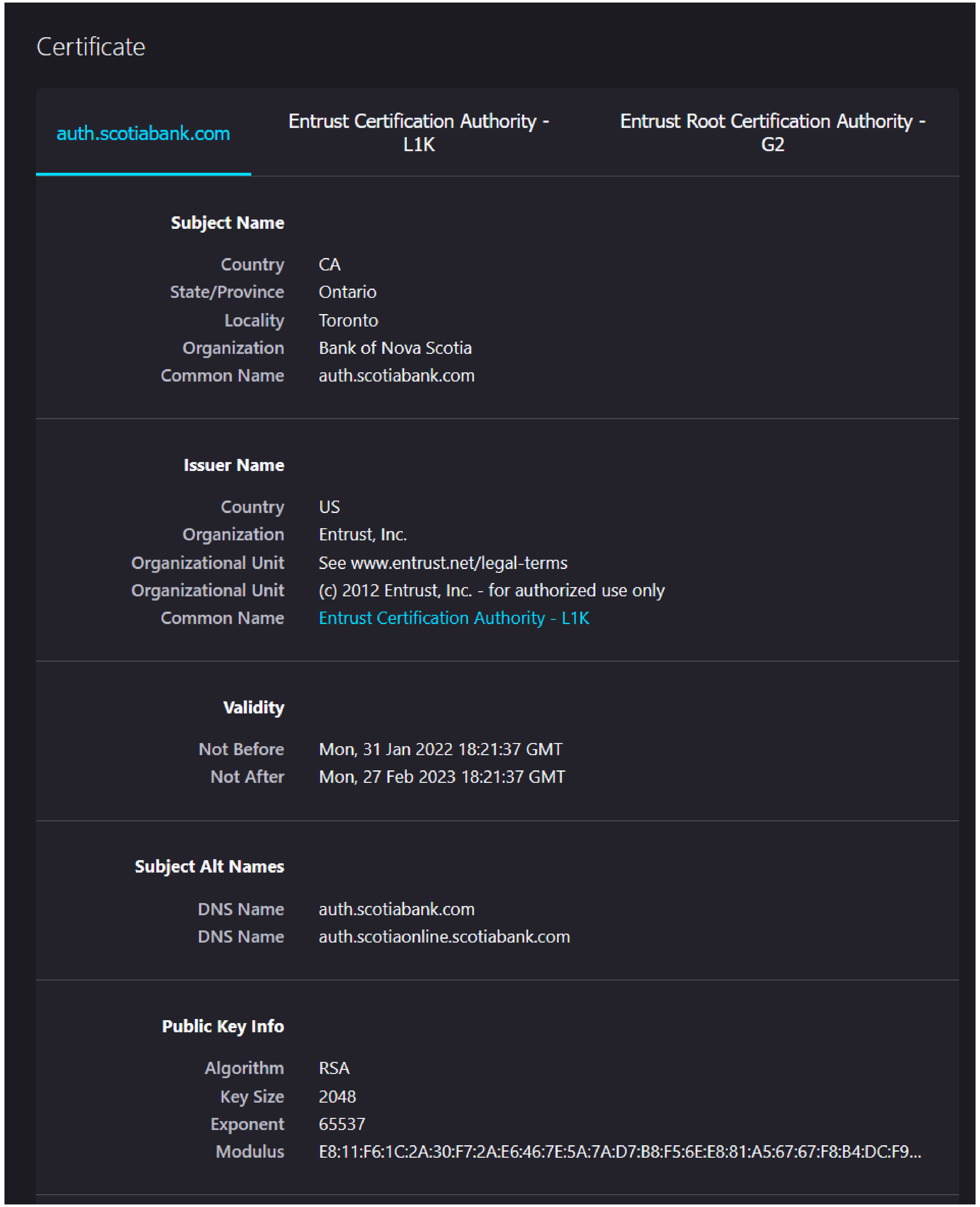


- What is that lock?
  - Securely binds domain name to public key (PK)
  - If PK is authenticated, then any message signed by that PK cannot be forged by non-authorized party

PK Certificate







Port Scanning

- Send a SYN or application-specific UDP packet to a port to see if any service is listening

- **Vertical Scan:** Try large number of ports on a single host. Typically use Nmap.

- **Horizontal Scan:** Try a single port on a large number of hosts. Typically ZMap

Firewalls

- Keep out unwanted traffic
- Can be done in the network (e.g., network perimeter) or at the host
- Many mechanisms
  - Packet filters (discussed last time)
  - Stateful packet filters (discussed last time)
  - Proxies, gateways

Proxies

- Want to look “deeper” into packets
  - Application type
  - Content
- Full TCP termination in the network
- Often done transparently (e.g., HTTP proxies)
- Allows access to objects passed over network
  - E.g., files, streams etc.
  - Can do lots of other fun things
    - E.g., content caching

- Proxies duplicate per-flow state held by clients
- How does this break end-to-end semantics of TCP?
  - E.g., what if proxy crashes right after reading from client?
    - lost data!

Intrusion Detection Systems (IDS)

- Software/device to monitor network traffic for attacks or policy violations
- Violations are reported to a central security information and event management (SIEM) system where analysts can later investigate

- **Signature Detection:** maintains long list of traffic patterns (rules) associated with attacks
- **Anomaly Detection:** attempts to learn normal behavior and report deviations

Open Source IDS

- Three Major Open Source IDS (and lots of commercial products)

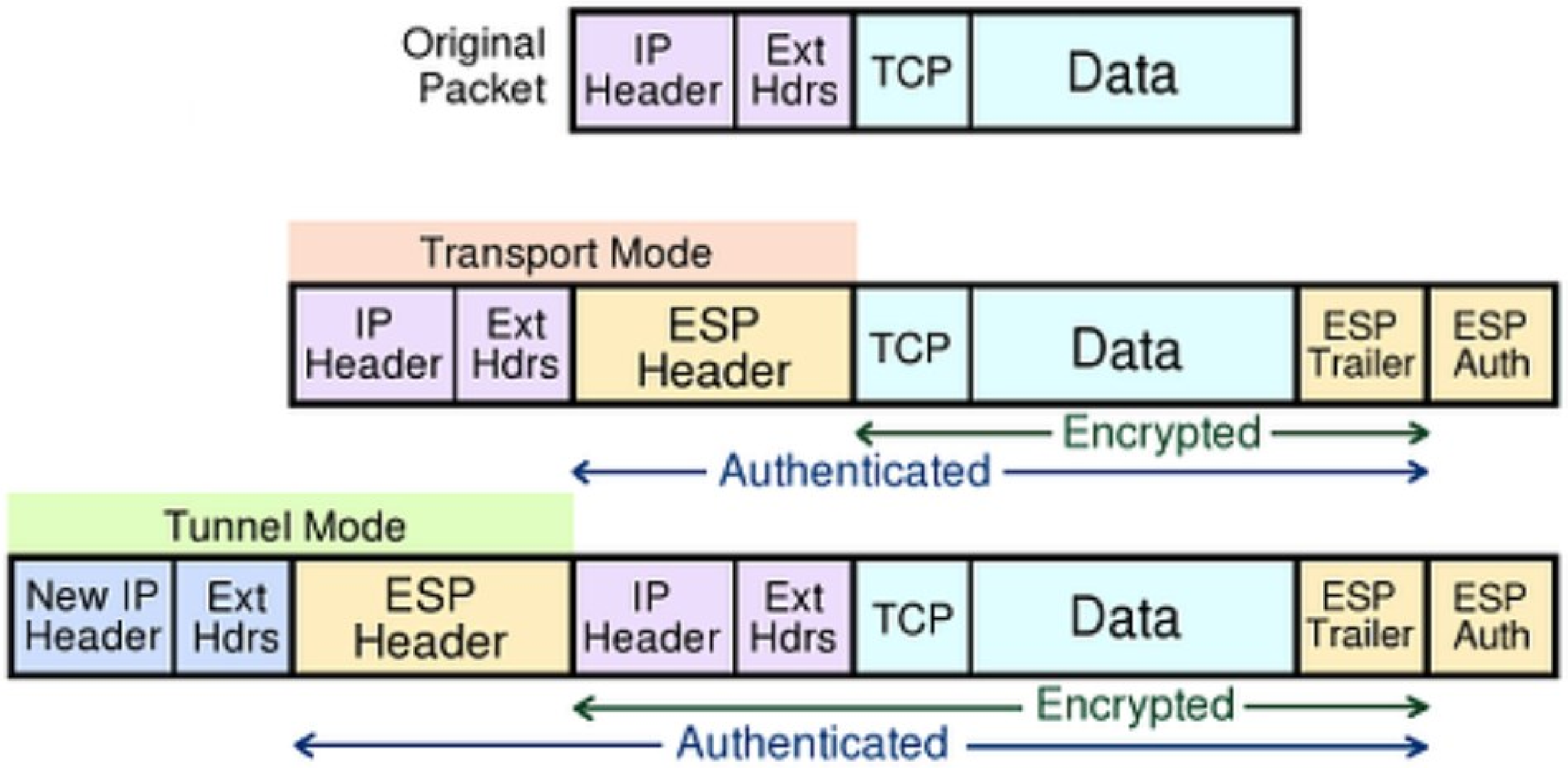
Snort

Zeek

Suricata

Virtual Private Networks (VPNs)

- Several VPN protocols exist (PPTP, L2TP, IPsec, OpenVPN)
- Most popular is IPsec.
- General IP Security framework
- Allows one to provide Access control, integrity, authentication, originality, and confidentiality
- Applicable to different settings



ESP: Encapsulating Secure Payload Protocol

Gooey Middle

- VPNs support the idea of having a secure internal network and untrusted public Internet.
- Attacker has a ton of access once the network perimeter is breached.
- Internal networks **aren't** that secure. Computers are compromised all the time and attackers have free rein.

Zero Trust Security (BeyondCorp)

- **Google:** assume internal network is also out to get you.
- Remove privileged intranet and put all corporate applications on the Internet.
- Access depends solely on device and user credentials, regardless of a user’s network location
- Protect applications, not the network
- Enable secure remote work without the need for a traditional VPN

Wrapping Up

- Internet **not** designed for security!
- Many, many attacks
  - Defense is difficult
  - Attackers are smart; Broken network aids them!
- Retrofitting solutions often break original design principles
  - Some of these solutions work, some of the time
  - Some make the network inflexible, brittle
- Time for new designs/principles?