

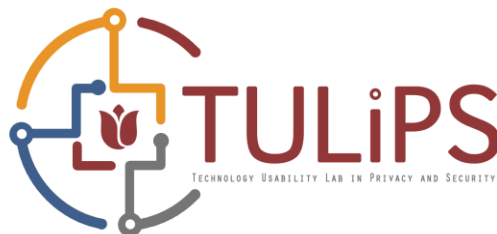
ECE458/ECE750T27: Computer Security Networking Threats

Dr. Kami Vaniea
Electrical and Computer Engineering
kami.vaniea@uwaterloo.ca



UNIVERSITY OF
WATERLOO

FACULTY OF
ENGINEERING



MIDTERM NEXT WEEK

THREAT MODELING

“A system which is unspecified can never be wrong, it can only be surprising.”

Security expects specifications

- There is no such thing as a fully secure system that is protected from everything.
- Systems can be protected against specified threats, allow specified actions, and accept specified risks.
- So what level or type of attacker is a system designed to protect against?
 - Local in-home attacker (spouse, child, visitor)
 - Remote attacker with limited infrastructure
 - Advertiser who wants to profile you to sell you more stuff
 - Curious system admin, no malice, but lots of “I wonder what bob is doing today” curiosity
 - Government-level attacker with ability to order companies to comply

CIAAA specifies the properties we want

Security properties to ensure

Confidentiality No improper information gathering

Integrity Data has not been (maliciously) altered

Availability Data/services can be accessed as desired

Accountability Actions are traceable to those responsible

Authentication User or data origin accurately identifiable

STRIDE: Specifies system threat types

- Spoofing
- Tampering
- Repudiation
- Information disclosure
- Denial of service
- Elevation of privilege

Threat	Desired property	Threat Definition
Spoofing	Authenticity (Authentication)	Pretending to be something or someone other than yourself
Tampering	Integrity	Modifying something on disk, network, memory, or elsewhere
Repudiation	Accountability	Claiming that you didn't do something or were not responsible; can be honest or false
Information disclosure	Confidentiality	Someone obtaining information they are not authorized to access
Denial of service	Availability	Exhausting resources needed to provide service
Elevation of privilege	Authorization	Allowing someone to do something they are not authorized to do

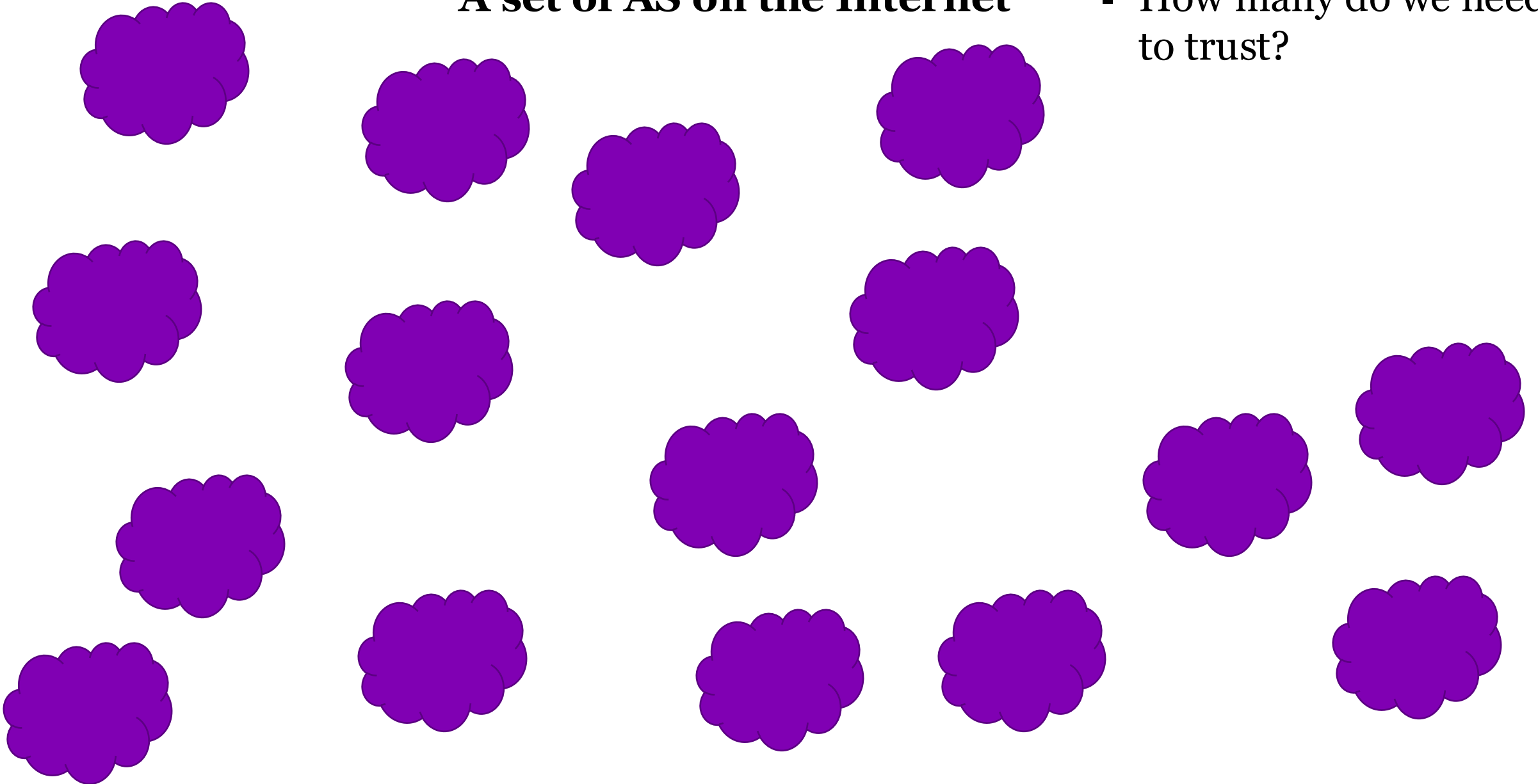
HARMS: Specifies human-level threat types

Table 1. Summary of the Human HARMS Model

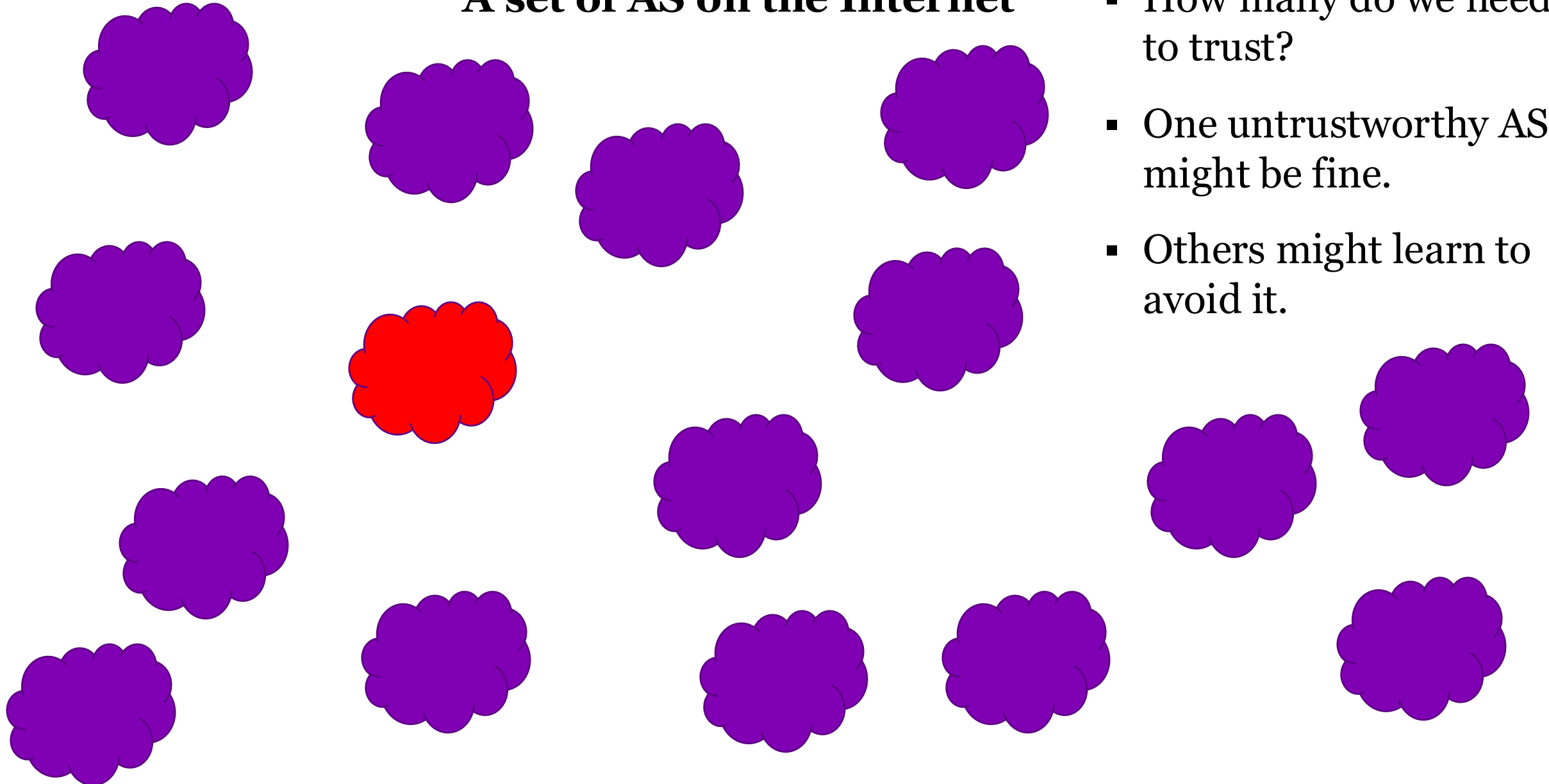
Term	Definition	Examples
Harassment	Causing distress through interactions	Sending hateful messages or playing loud sounds
Access/Infiltration	Obtaining or extending access	Increasing own privileges, or adding an external user to a system
Restrictions	Reducing access of existing user	Removing legitimate user's access, or inhibiting specific functionality
Manipulation/Tampering	Controlling other users	Blackmailing users with information from the system, or creating fake evidence
Surveillance	Observing others without their knowledge	Using cameras and microphones to observe users, or investigating logs of past activity

A set of AS on the Internet

- How many do we need to trust?

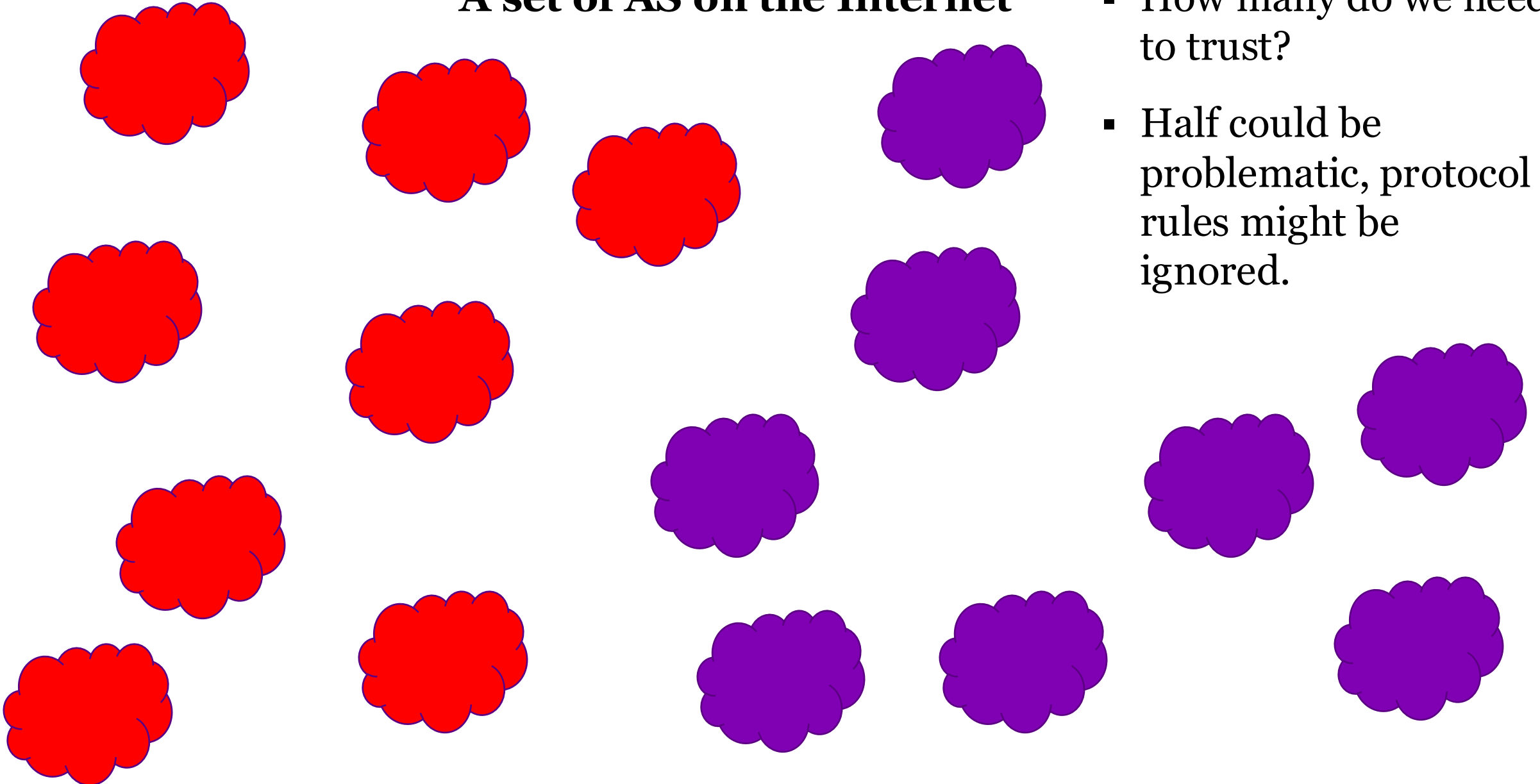


A set of AS on the Internet



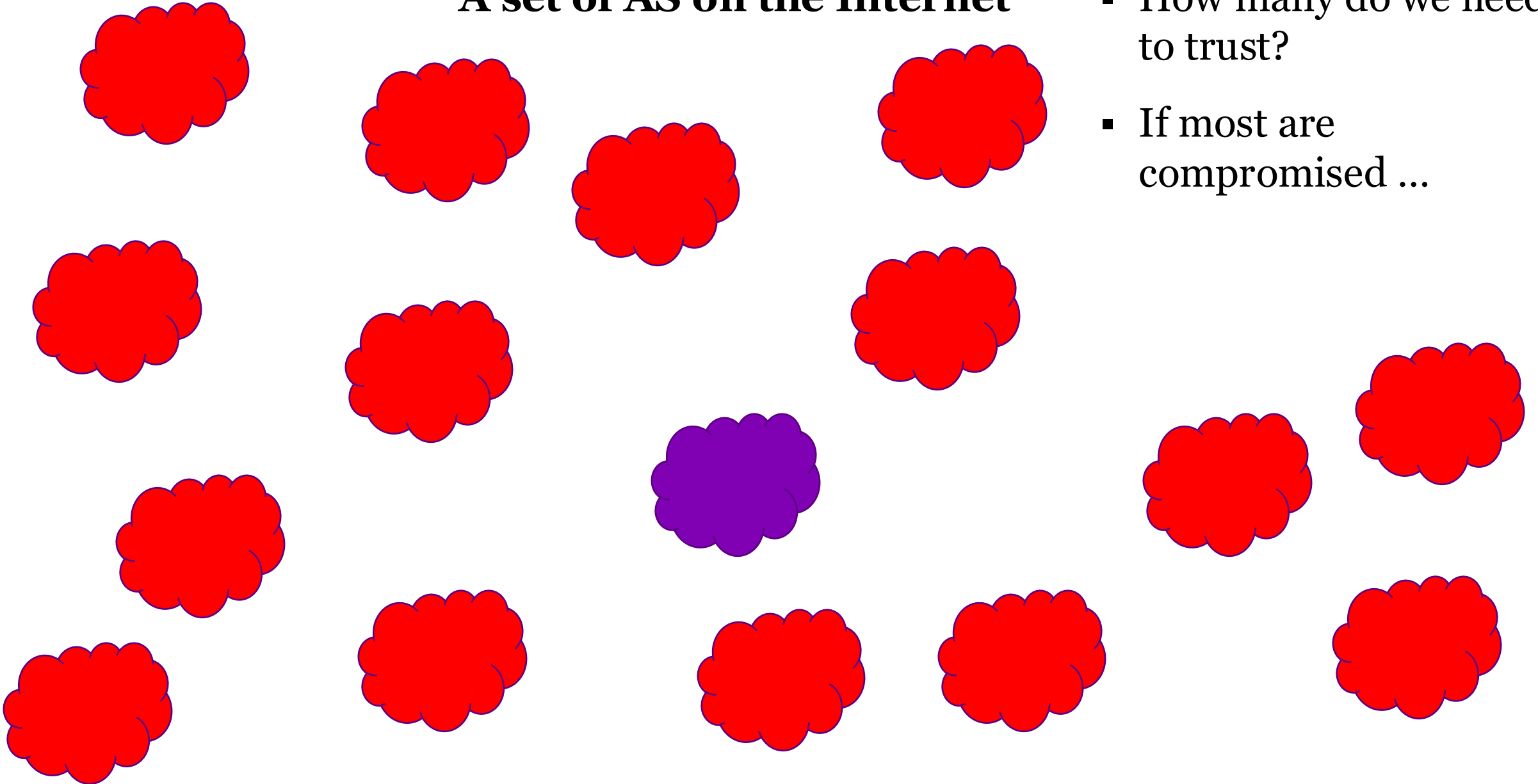
- How many do we need to trust?
- One untrustworthy AS might be fine.
- Others might learn to avoid it.

A set of AS on the Internet



A set of AS on the Internet

- How many do we need to trust?
- If most are compromised ...



Protocols are mutually agreed behavior not rules

- If everyone follows the BGP protocol then the Internet works as expected
- There is advantage in cooperation
- There is no way to force all AS to correctly follow a specified protocol



For more than two hours on Thursday, June 6, a large chunk of European mobile traffic was rerouted through the infrastructure of China Telecom, China's third-largest telco and internet service provider (ISP).

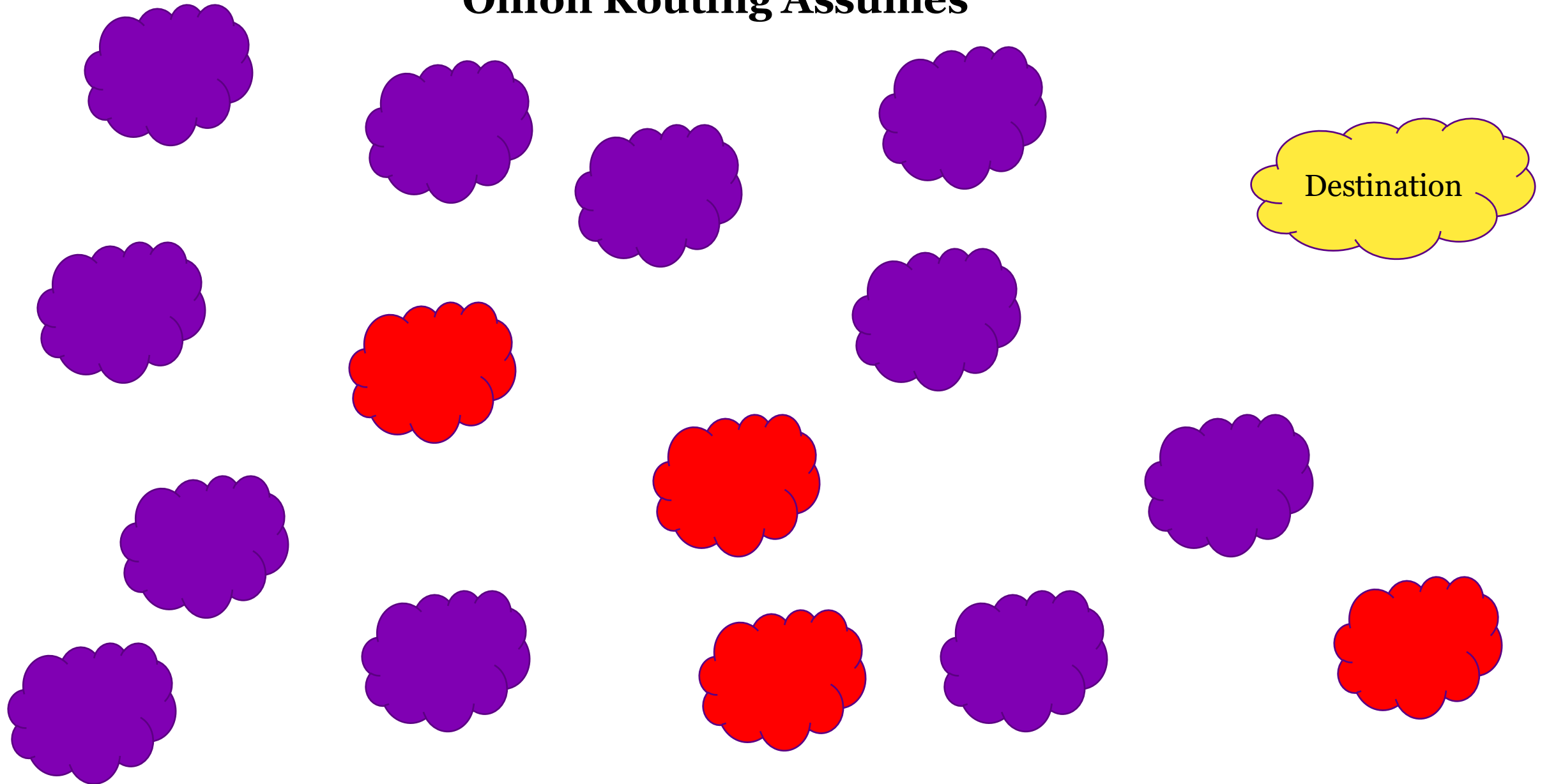
The incident occurred because of a BGP route leak at Swiss data center colocation company Safe Host, which accidentally leaked over 70,000 routes from its internal routing table to the Chinese ISP.



ONION ROUTING

aka Tor

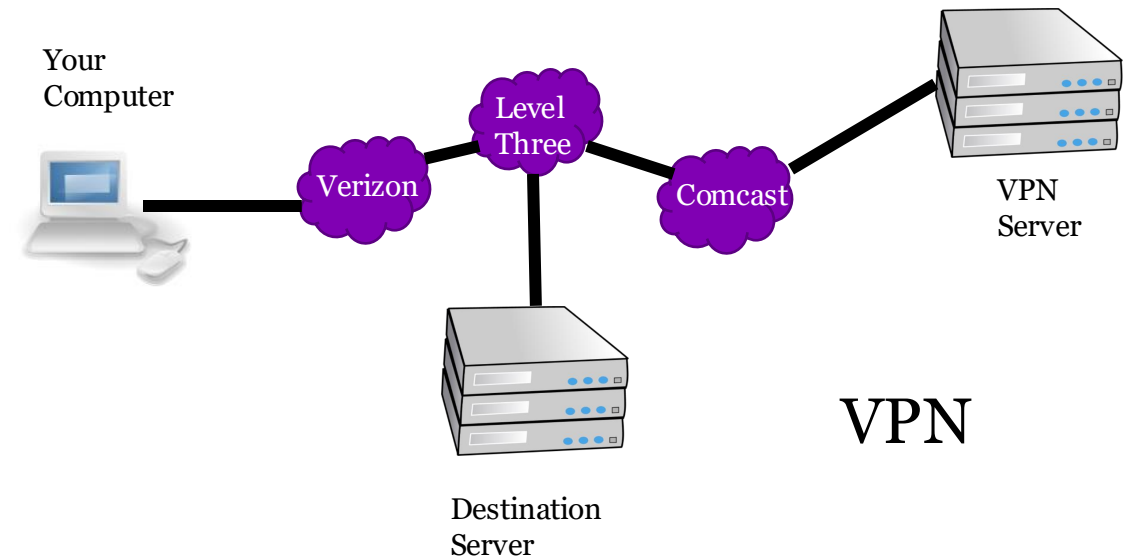
Onion Routing Assumes



Problem: Anonymity

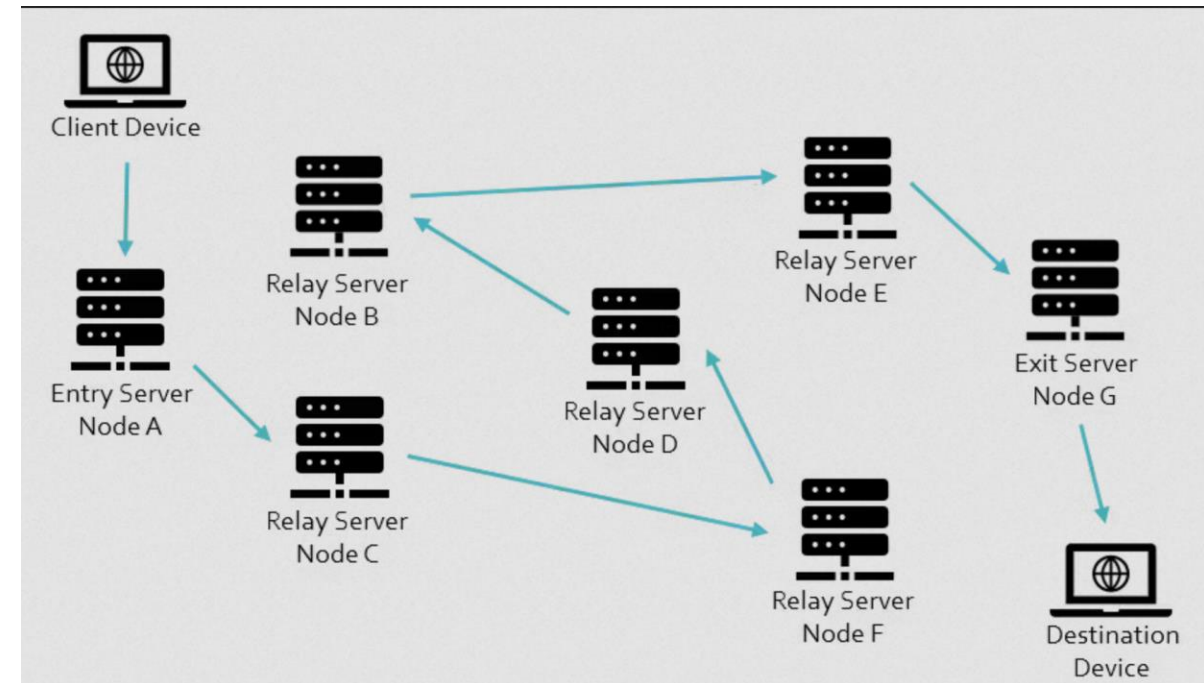
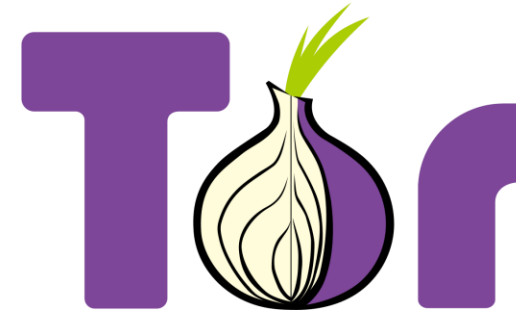
- The user wants to access parts of the internet without being tracked by:
 - Governments
 - Destination server
 - First-hop router
- Avoid being “tracked” means:
 - User’s IP address not associated with traffic
 - User’s identity cannot be associated with their traffic

- VPN solves some but not all
 - VPN knows who the user is (authentication) and what the traffic is
 - Destination server knows who the VPN is and could theoretically be compelled to provide data (Government attacker)



Onion Routing

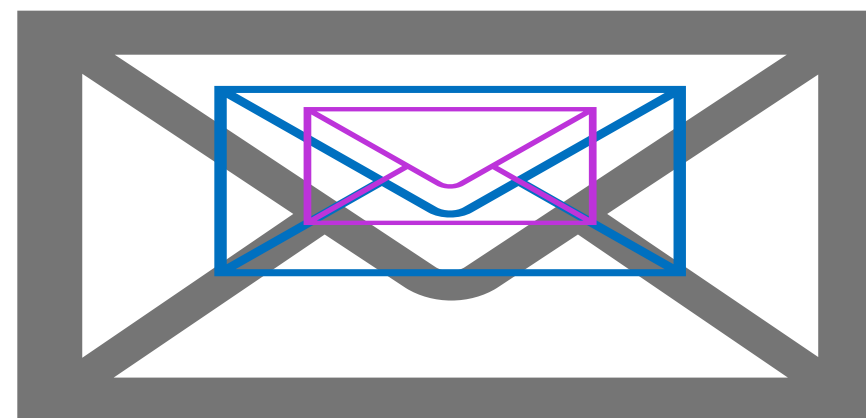
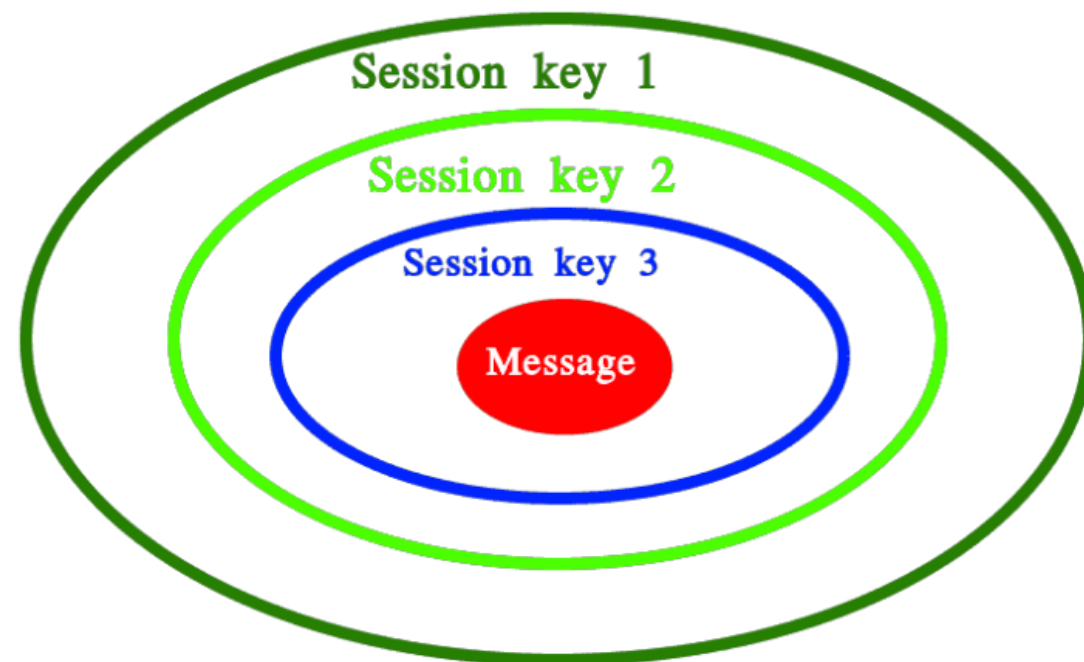
- Goal: allow users to access the internet such that their identity and traffic are not linked
- Each server knows where it got traffic and where it sent traffic, but it doesn't know the whole path, just its neighbors
- This approach protects the client from connecting their real IP address with their traffic
 - First node (Node A) has the real IP address
 - Last node (Node G) has the real traffic
 - Nodes A and G do not know they are carrying the same person's traffic



<https://privacyhq.com/documentation/onion-routing-explained/>

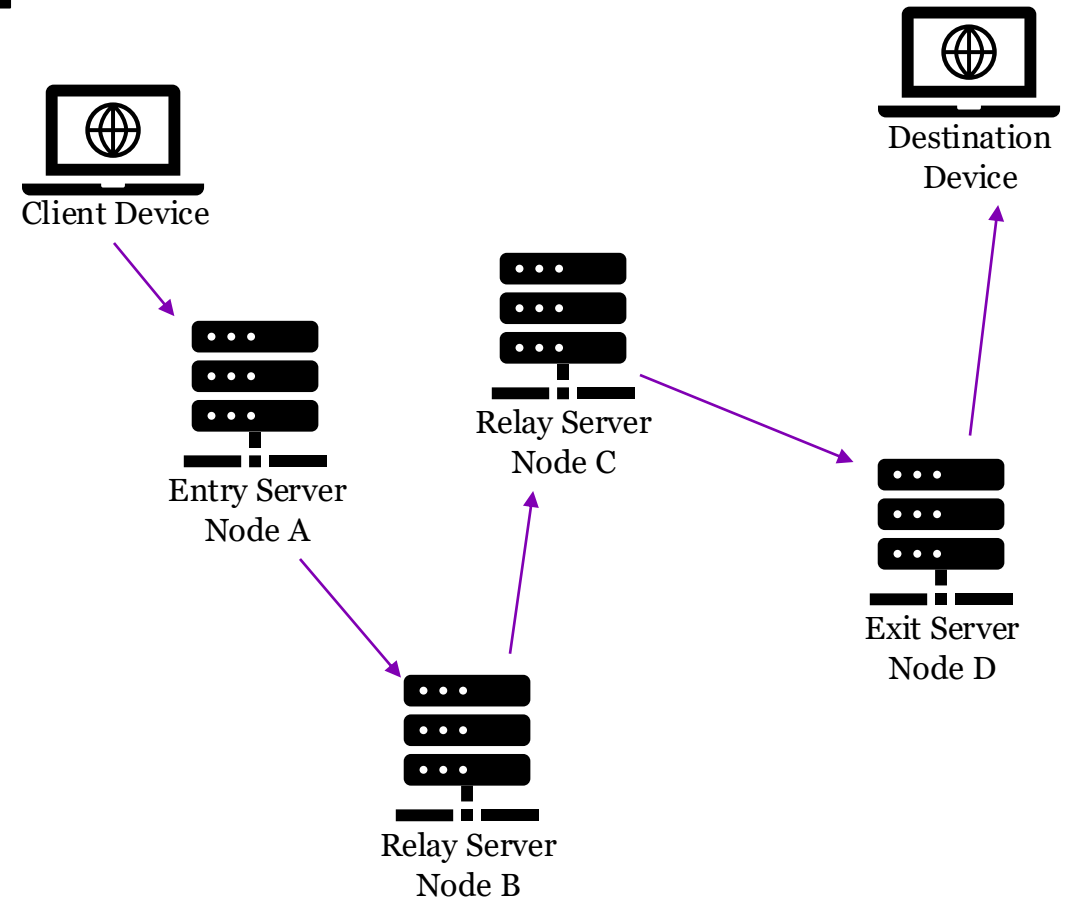
Onion Routing

- Encryption is also important to make all this work
- The client has a list of all onion routers in the network, they select a set and encrypt the message in concentric layers
- Each layer:
 - Encrypted with current node's public key
 - Address of next destination



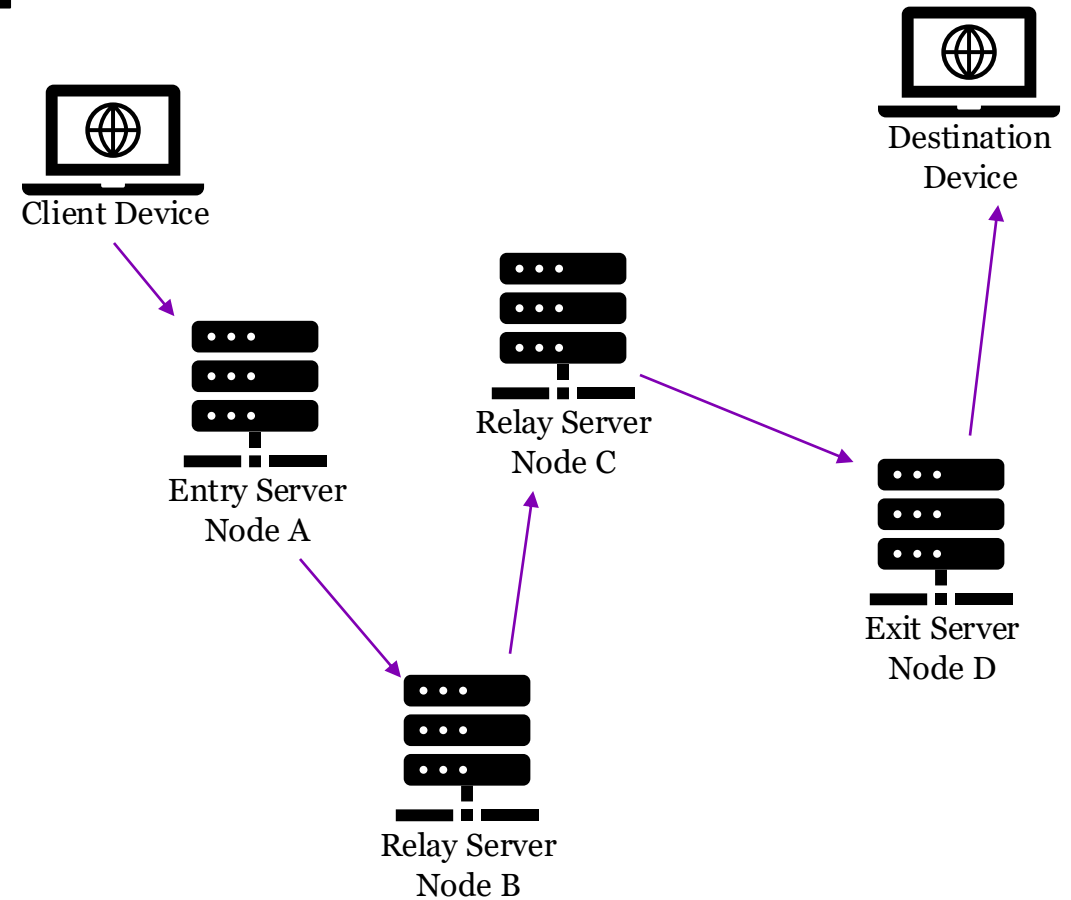
Onion Routing: Client->Destination

- M = Message
- $M_1 = E(M, \text{Destination}_{\text{addr}}, D_{\text{pub}})$
- $M_2 = E(M_1, D_{\text{addr}}, C_{\text{pub}})$
- $M_3 = E(M_2, C_{\text{addr}}, B_{\text{pub}})$
- $M_4 = E(M_3, B_{\text{addr}}, A_{\text{pub}})$
- Client sends M_4 to Node A



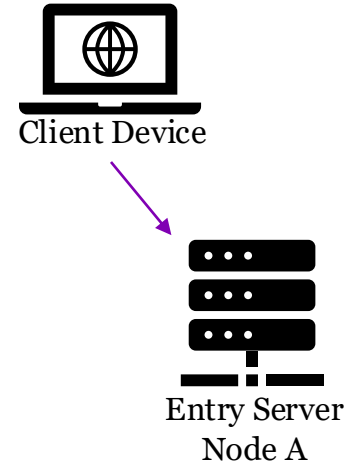
Onion Routing: Client->Destination

- Route iteratively built by client.
- For each node:
 - Send an initiation request including creating a new session (symmetric) key.



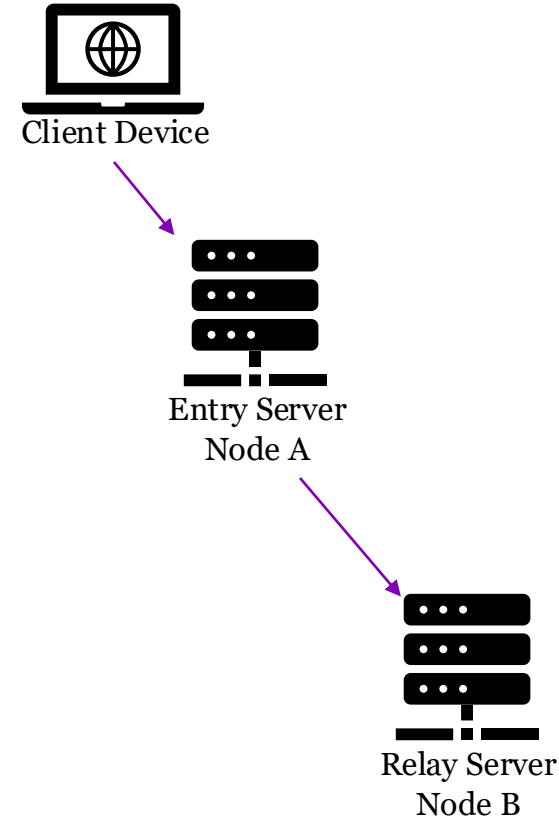
Onion Routing: Client->Destination

- Client selects a set of nodes and picks a set of session keys such that each node has a different associated session key
 - Session keys could also be negotiated via Diffie-Hillman type approach, pre-selected just easier to explain
- Client then builds the path one node at a time
- $M = \text{Start onion route using } A_{\text{session}}$
- $M_1 = \text{PublicKeyE}(M, A_{\text{pub}})$
- Client sends M_1 to Node A
- Node A decrypts using A_{priv} and now has A_{session} key for future communication



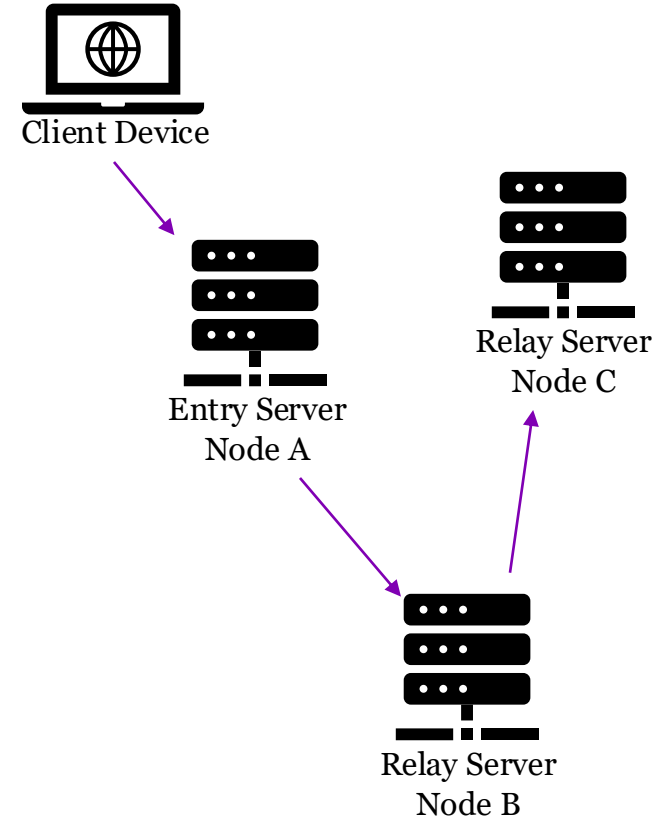
Onion Routing: Client->Destination

- Client then extends the path to the next planned node B
- $M = \text{Start onion route using } B_{\text{session}}$
- $M_1 = \text{PublicKeyE}(M, B_{\text{pub}})$
- $M_2 = \text{SymmetricE}(M_1 + B_{\text{addr}}, A_{\text{session}})$
- Client sends M_2 to Node A
- Node A decrypts using A_{session} that was sent earlier and forwards M_1 to B_{addr}
- Node B decrypts using B_{priv} and now has B_{session}



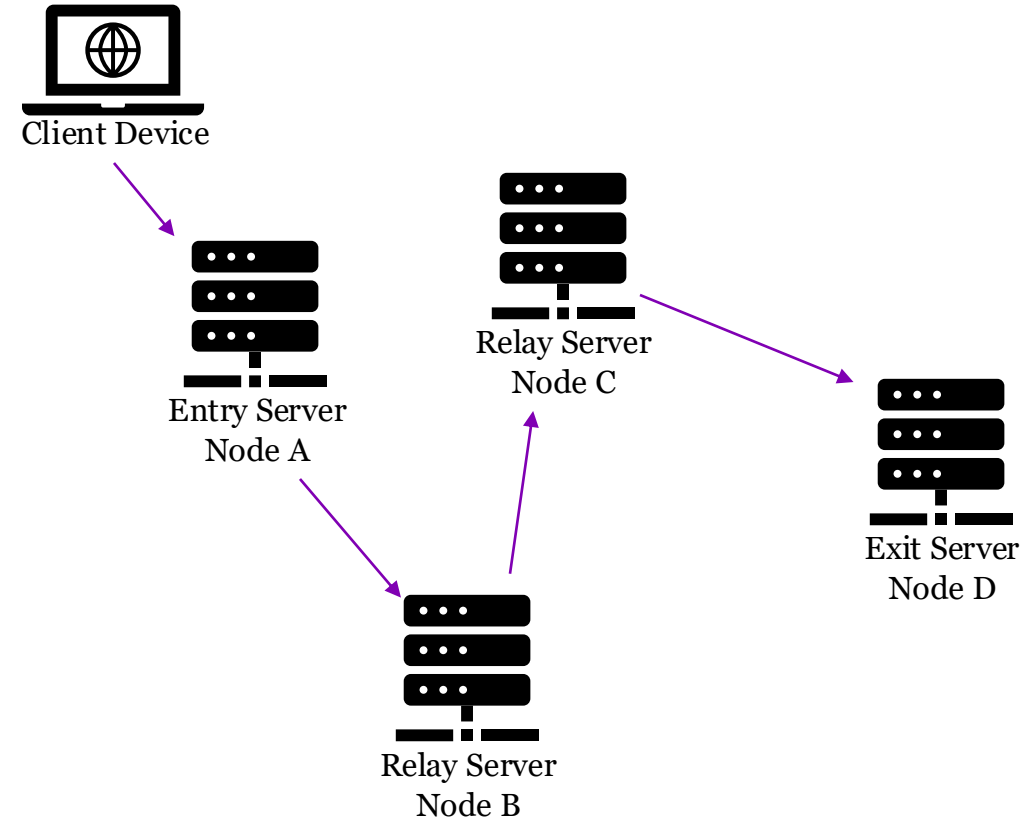
Onion Routing: Client->Destination

- Client then extends the path to the next planned node C
- $M = \text{Start onion route using } C_{\text{session}}$
- $M_1 = \text{PublicKeyE}(M, C_{\text{pub}})$
- $M_2 = \text{SymmetricE}(M_1 + C_{\text{addr}}, B_{\text{session}})$
- $M_3 = \text{SymmetricE}(M_2 + B_{\text{addr}}, A_{\text{session}})$
- Client sends M_3 to Node A
- Node A decrypts using A_{session} and forwards M_2 to B_{addr}
- Node B decrypts using B_{session} and forwards M_1 to C_{addr}
- Node C decrypts using C_{priv} and now has C_{session}



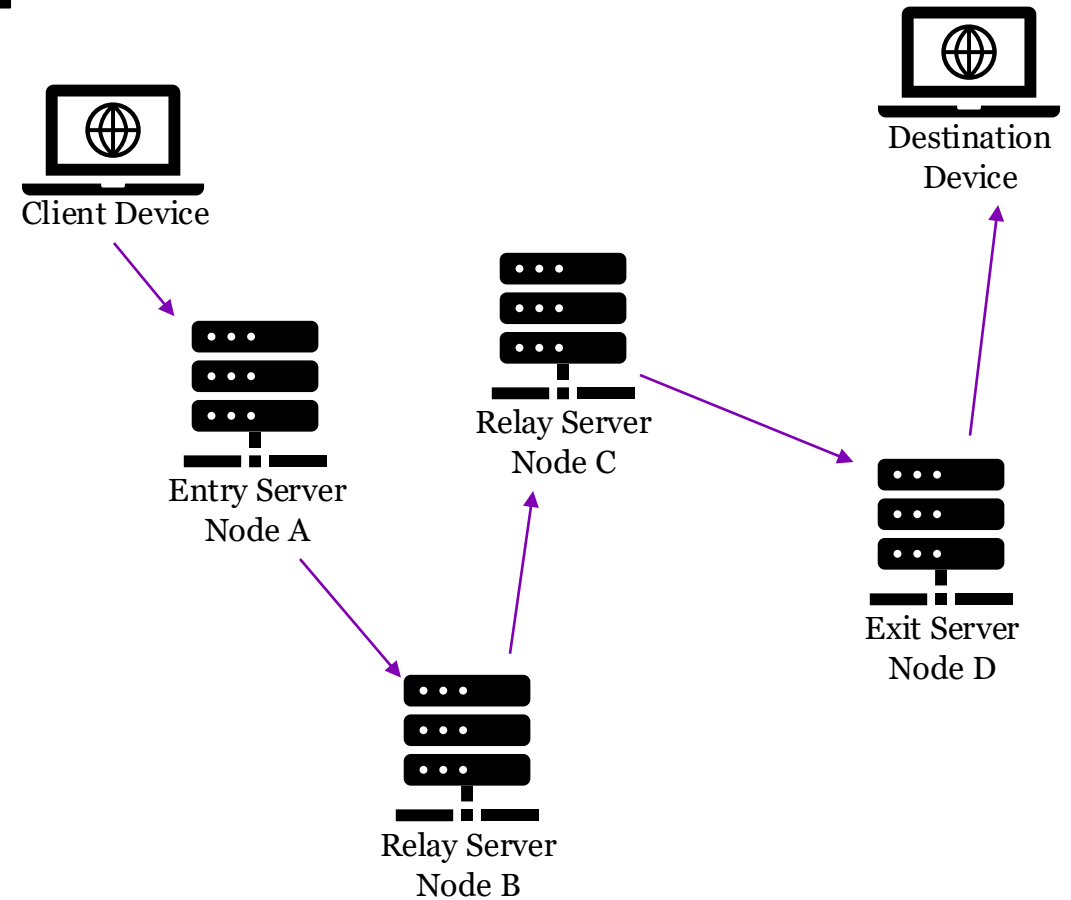
Onion Routing: Client->Destination

- Client then extends the path to the next planned node D
- $M = \text{Start onion route using } D_{\text{session}}$
- $M_1 = \text{PublicKeyE}(M, D_{\text{pub}})$
- $M_2 = \text{SymmetricE}(M_1 + D_{\text{addr}}, C_{\text{session}})$
- $M_3 = \text{SymmetricE}(M_2 + C_{\text{addr}}, B_{\text{session}})$
- $M_4 = \text{SymmetricE}(M_3 + B_{\text{addr}}, A_{\text{session}})$
- Client sends M_4 to Node A
- Node A decrypts using A_{session} and forwards M_3 to B_{addr}
- Node B decrypts using B_{session} and forwards M_2 to C_{addr}
- Node C decrypts using C_{session} and forwards M_1 to D_{addr}
- Node D decrypts using D_{priv} and now has D_{session}



Onion Routing: Client->Destination

- Client then uses the established path to communicate with destination server
- M = Message
- $M_1 = \text{SymmetricE}(M + \text{Destination}_{\text{addr}}, D_{\text{session}})$
- $M_2 = \text{SymmetricE}(M_1 + D_{\text{addr}}, C_{\text{session}})$
- $M_3 = \text{SymmetricE}(M_2 + C_{\text{addr}}, B_{\text{session}})$
- $M_4 = \text{SymmetricE}(M_3 + B_{\text{addr}}, A_{\text{session}})$
- Client sends M_4 to Node A
- Node A decrypts using A_{session} and forwards M_3 to B_{addr}
- Node B decrypts using B_{session} and forwards M_2 to C_{addr}
- Node C decrypts using C_{session} and forwards M_1 to D_{addr}
- Node D decrypts using D_{session} and forwards M to the Destination device

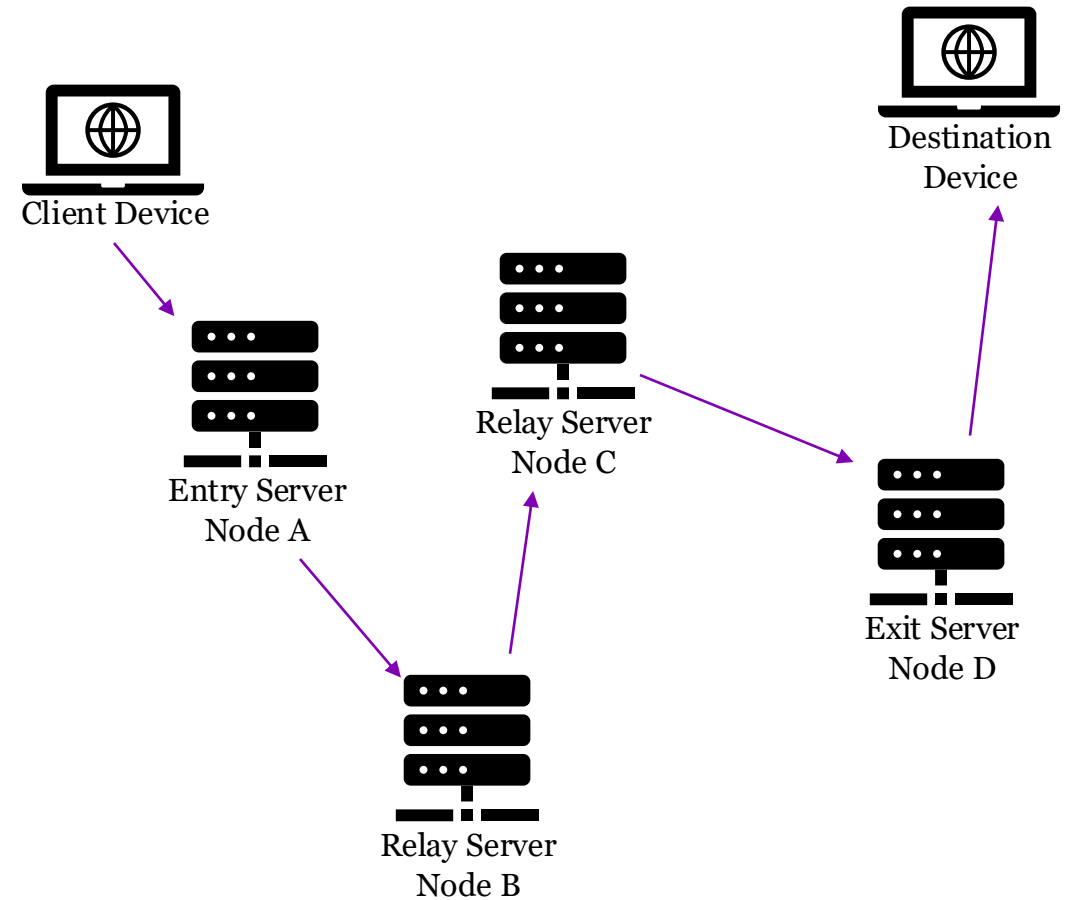


Think-pair-share

- What would the encryption look like for Destination -> Client

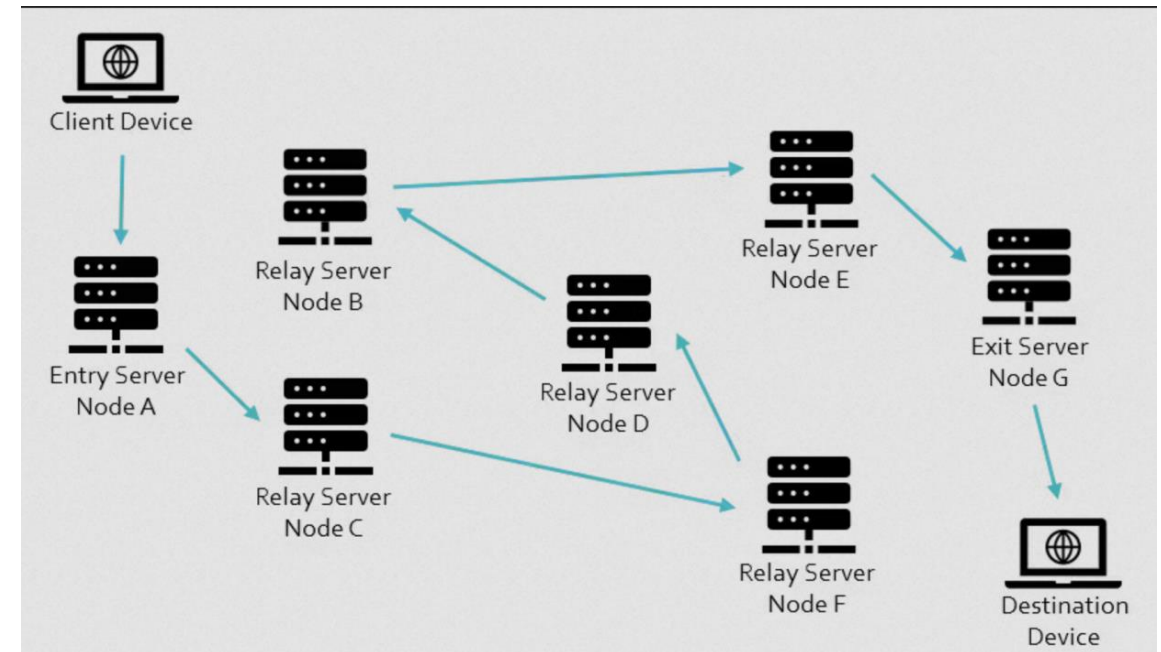
Client->Destination

- M = Message
- $M_1 = \text{SymmetricE}(M + \text{Destination}_{\text{addr}}, D_{\text{session}})$
- $M_2 = \text{SymmetricE}(M_1 + D_{\text{addr}}, C_{\text{session}})$
- $M_3 = \text{SymmetricE}(M_2 + C_{\text{addr}}, B_{\text{session}})$
- $M_4 = \text{SymmetricE}(M_3 + B_{\text{addr}}, A_{\text{session}})$
- Client sends M_4 to Node A
- Node A decrypts using A_{session} and forwards M_3 to B_{addr}
- Node B decrypts using B_{session} and forwards M_2 to C_{addr}
- Node C decrypts using C_{session} and forwards M_1 to D_{addr}
- Node D decrypts using D_{session} and forwards M to the Destination device



Onion Routing

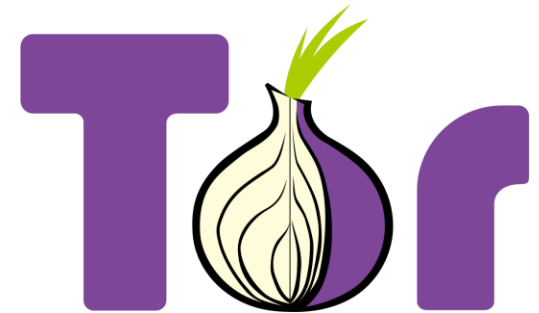
- Each Node only knows the address of where it got the packet and the address of where the packet is going
- They can only decrypt one layer of the packet
- Called onion routing because it is done in layers, with layers constructed by the client and then stripped off by the nodes



<https://privacyhq.com/documentation/onion-routing-explained/>

Tor

- Tor is popular software that uses onion routing
- There are many ways a user can still show who they are even if using Tor
 - Example: log into Facebook
- Routing everything across Tor could be bad because all exiting traffic could be connected together, so if one bit of traffic leaks your identity, they identity known for all traffic
- Tor is often bundled with a carefully setup browser



DENIAL OF SERVICE

Denial of Service (DoS): An attack that prevents valid users from accessing a service.

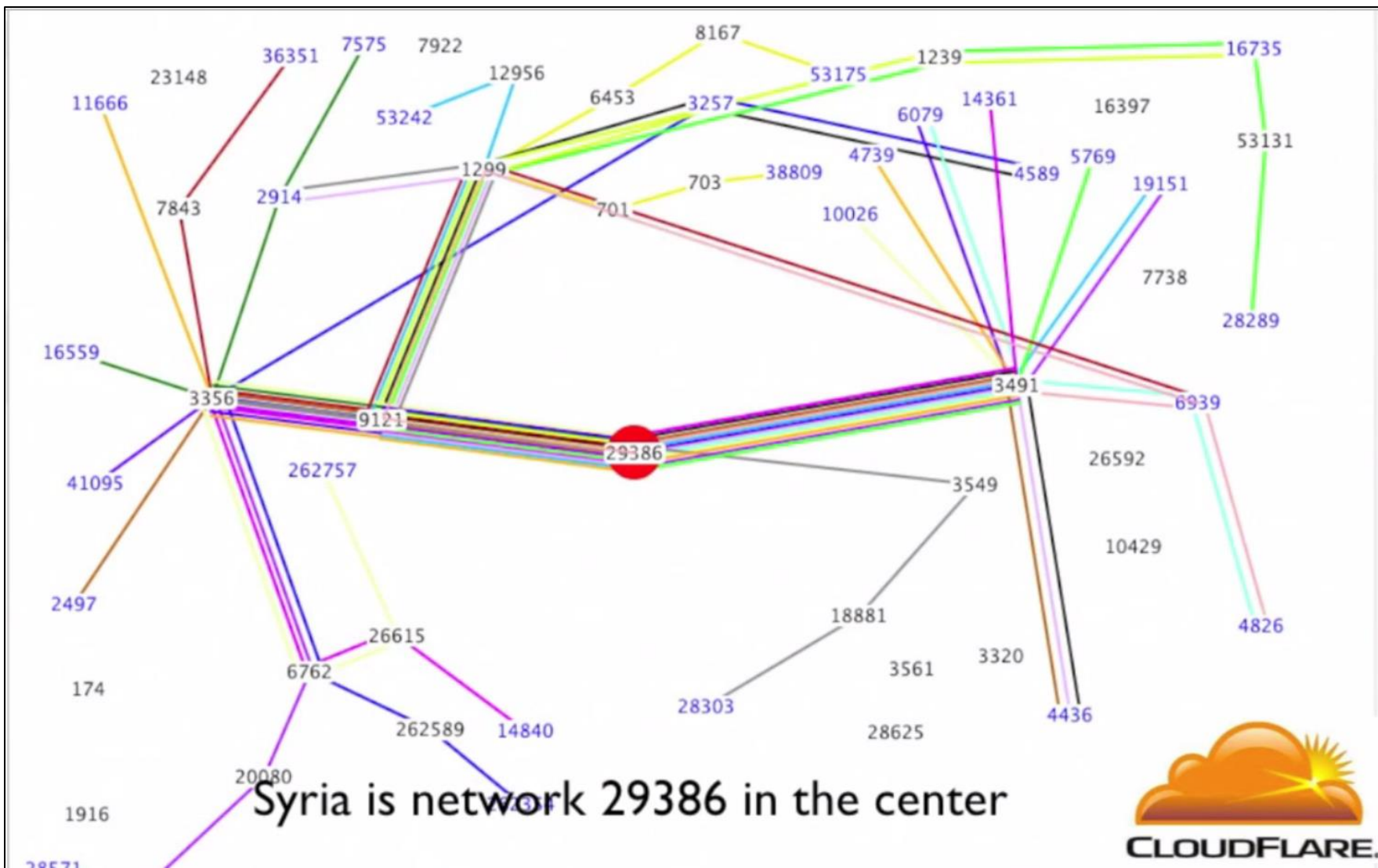
Common examples:

- Cutting power, cables, etc.
- Overloading a server with invalid traffic
- Removing a user account
- Changing the DNS to point to the wrong page

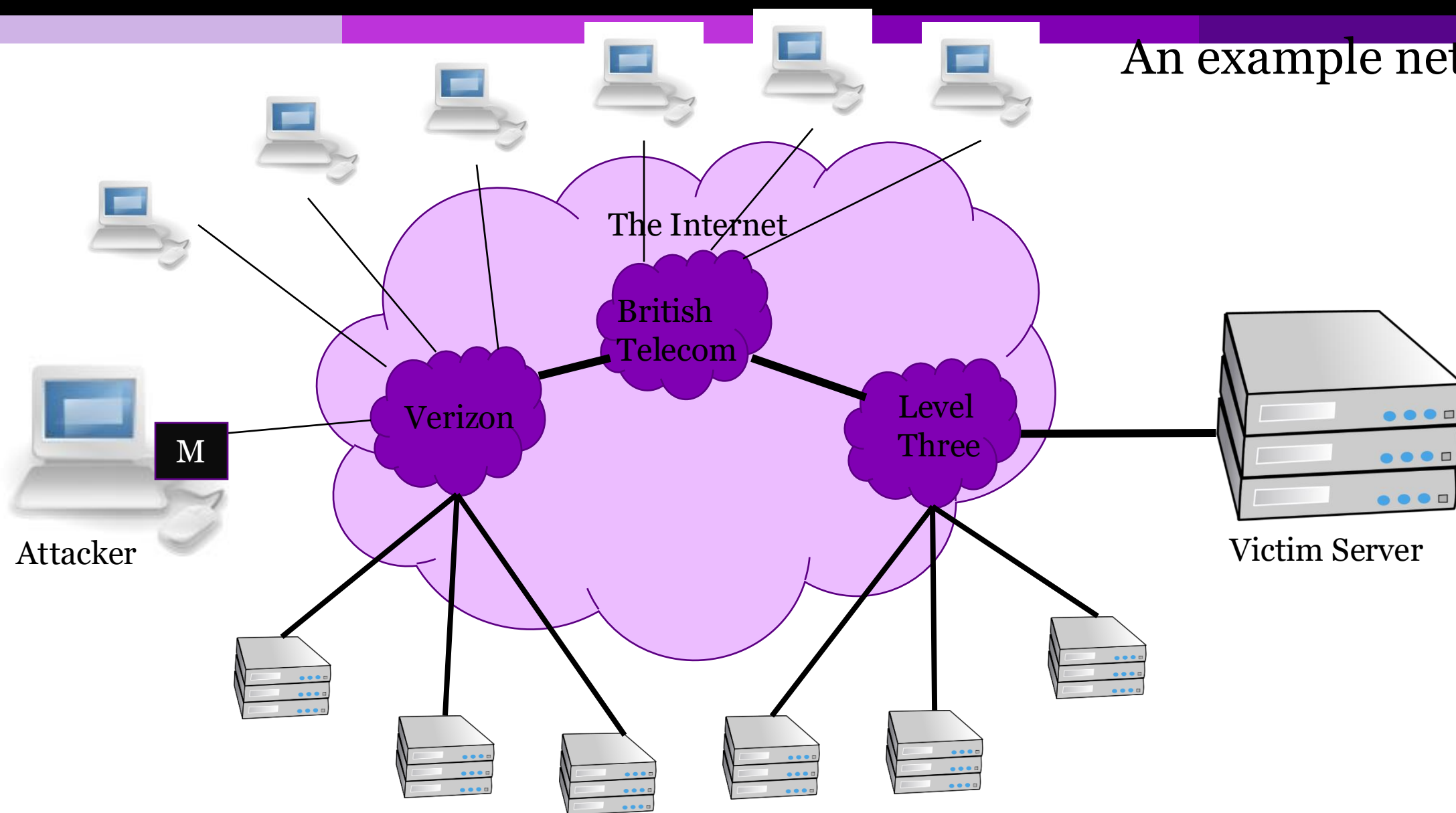
Attacks:

- SYN flooding
- Spoofing
- Smurfing

Syria's network shutdown is a DoS



An example network



SYN Flooding

Send tons of requests at the victim and overload them.

- Basic three-part handshake used by Alice to initiate a TCP connection with Bob.

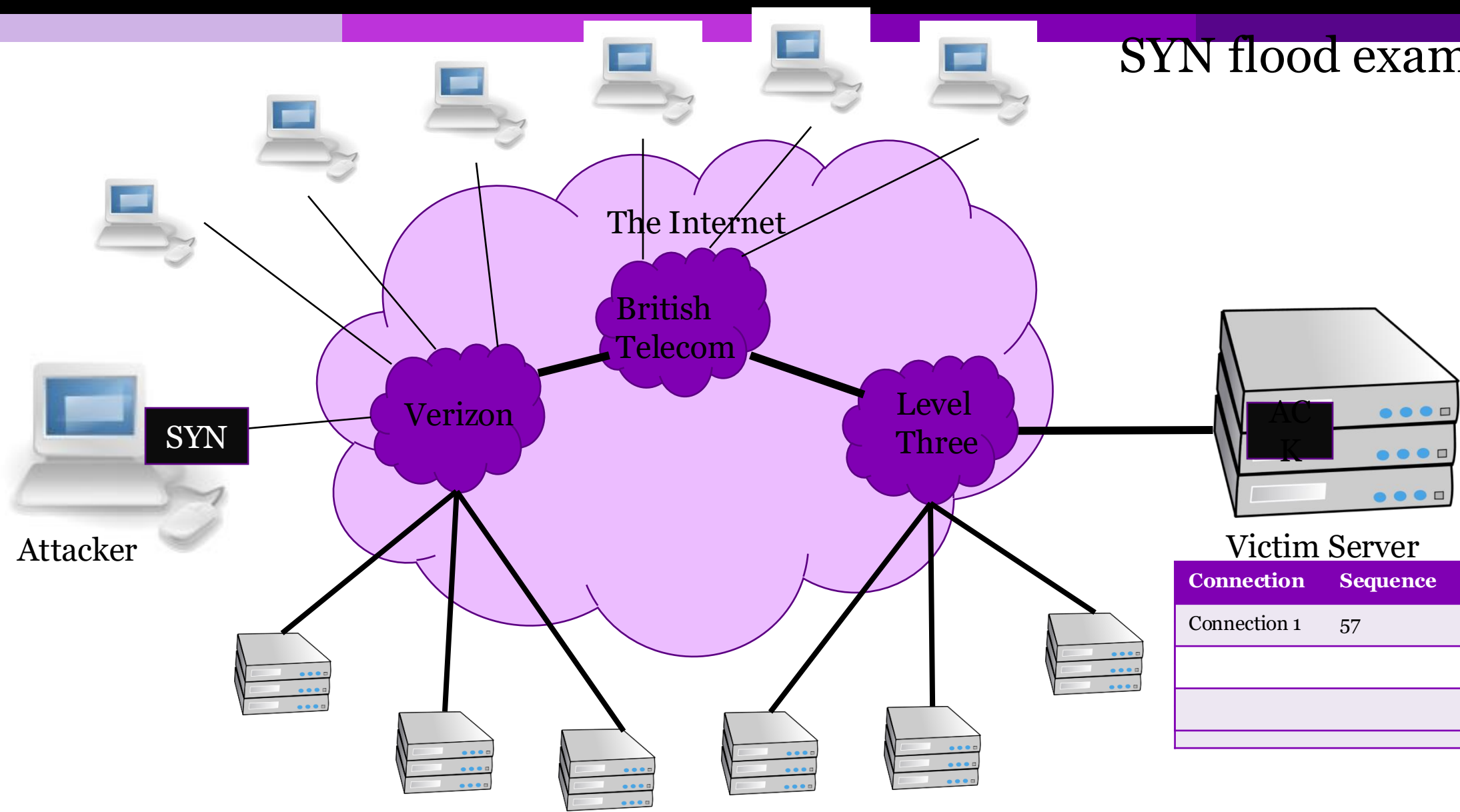
$A \rightarrow B : \text{ SYN, } X$

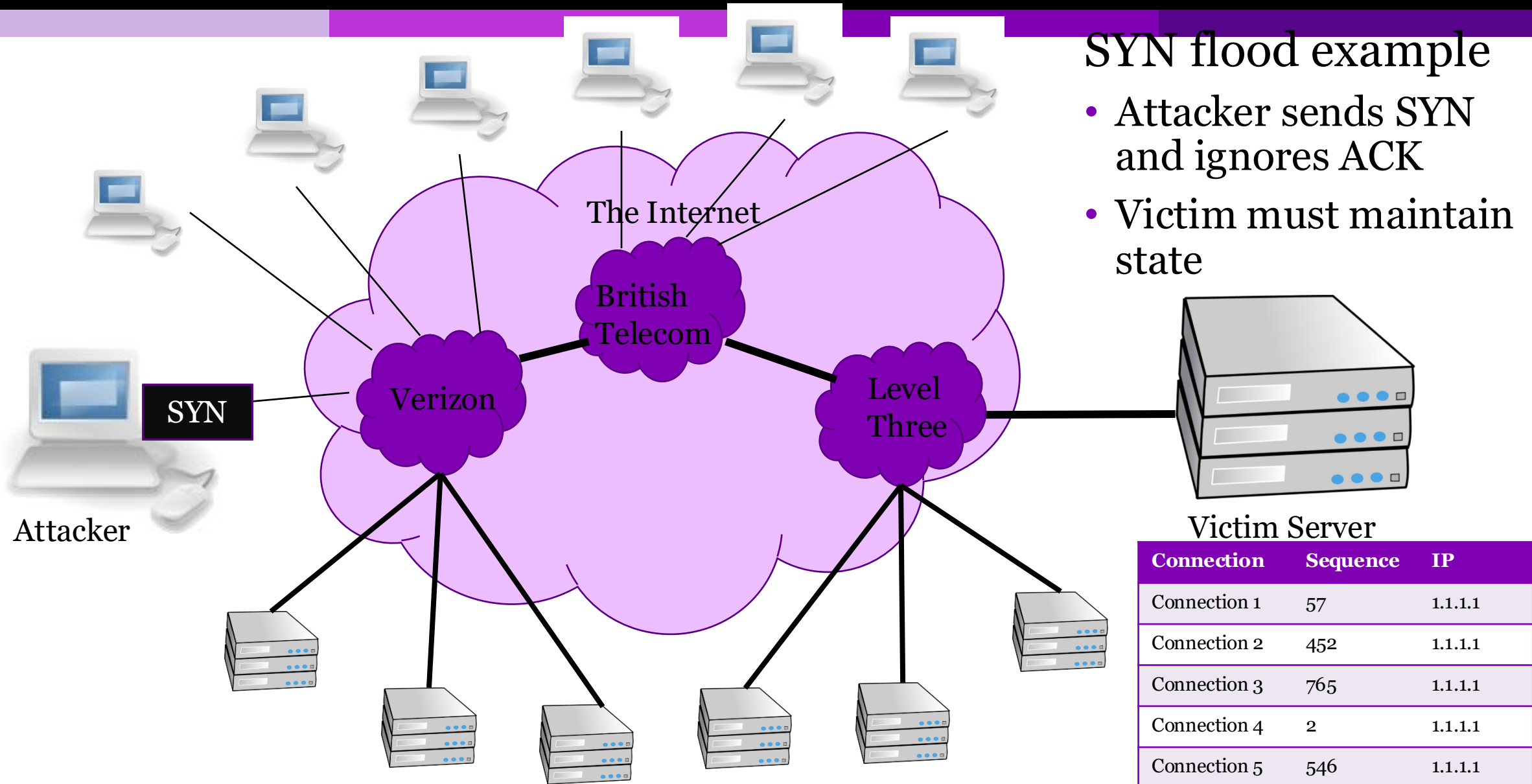
$B \rightarrow A : \text{ ACK, } X + 1; \text{ SYN, } Y$

$A \rightarrow B : \text{ ACK, } Y + 1$

- Alice sends many SYN packets, without acknowledging any replies. Bob accumulates more SYN packets than he can handle.

SYN flood example





Victim Server

Connection	Sequence	IP
Connection 1	57	1.1.1.1
Connection 2	452	1.1.1.1
Connection 3	765	1.1.1.1
Connection 4	2	1.1.1.1
Connection 5	546	1.1.1.1
Connection 6	97	1.1.1.1
Connection 7	56	1.1.1.1
Connection 8	15	1.1.1.1

SYN Flooding

- Problems
 - Attribution – attacker uses their own IP which could be traced
 - Bandwidth – attacker uses their own bandwidth which is likely smaller than a server's
- Effective against a small target
 - Someone running a game server in their home
- Not effective against a large target
 - Company website

Spoofing: forged TCP packets

- Same as SYN flooding, but forge the source of the TCP packet
- Advantages:
 - Harder to trace
 - ACKs are sent to a second computer, less attacker bandwidth used
- Problems:
 - Ingress filtering is commonly used to drop packets with source addresses outside their origin network fragment.

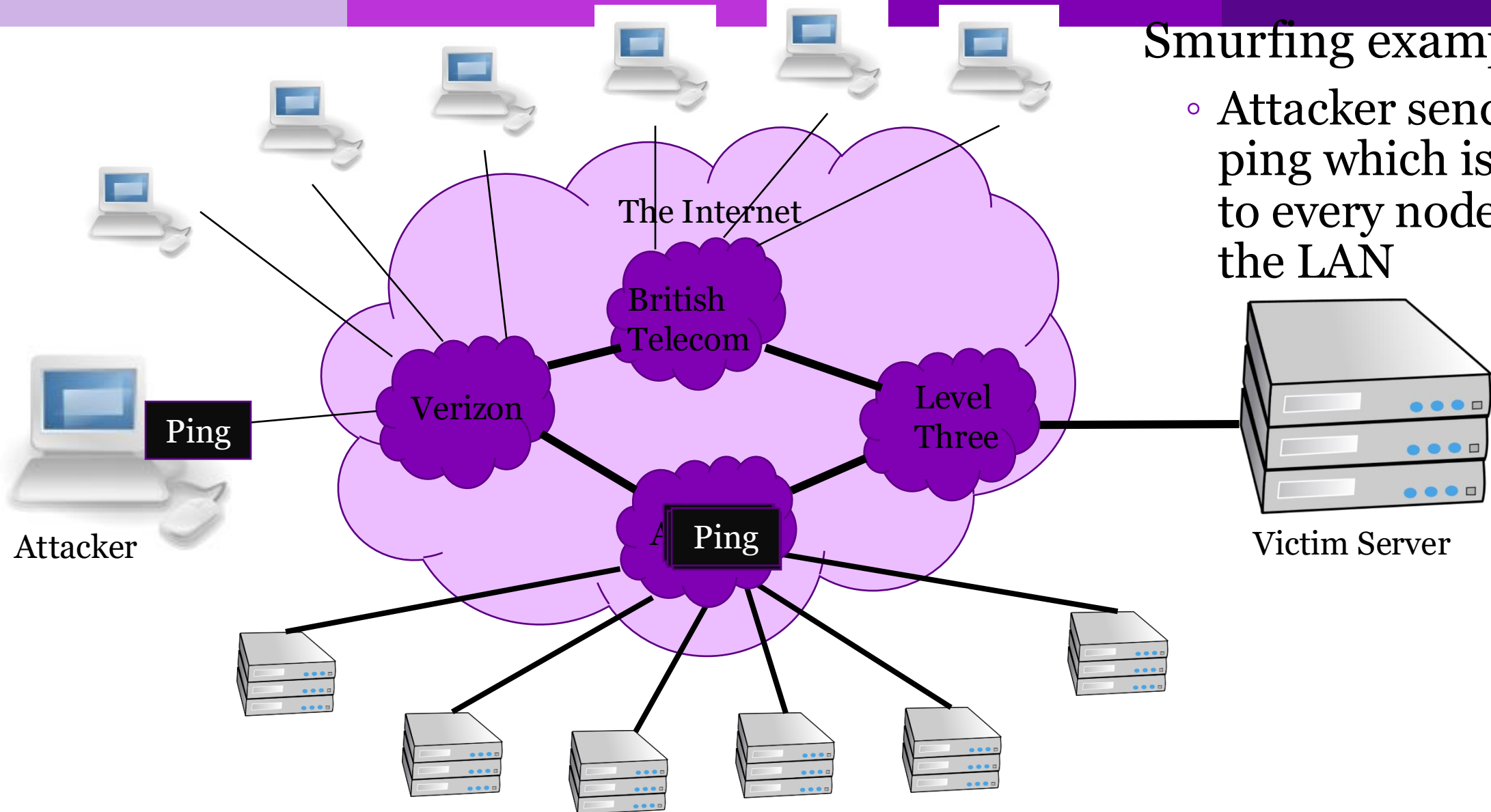
Friday Lecture Stopped Here

Smurfing (directed broadcast)

- The smurfing attack exploits the ICMP (Internet Control Message Protocol) whereby remote hosts respond to echo packets to say they are alive (ping).
- Some implementations respond to pings to broadcast addresses.
- Idea: Ping a LAN to find hosts, which then all respond to the ping.
- Attack: make a packet with a forged source address containing the victim's IP number. Send it to a smurf amplifier, who swamp the target with replies.

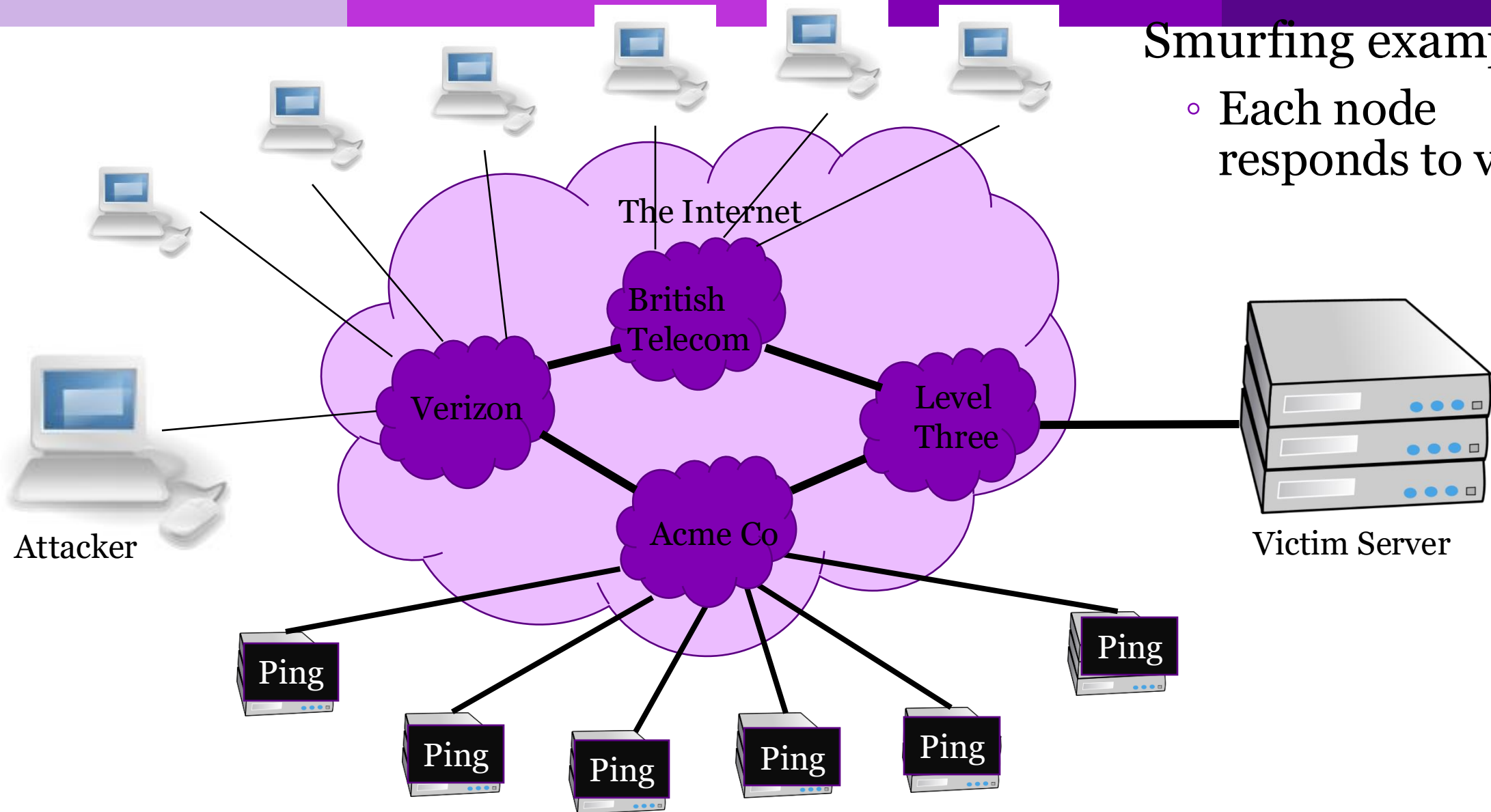
Smurfing example

- Attacker sends 1 ping which is sent to every node on the LAN



Smurfing example

- Each node responds to victim



**LANs that allow
Smurf attacks are
badly configured.
One approach is to
ban these LANs.**



Smurf Amplifier Registry (SAR)
<http://www.powertech.no/smurf/>

Current top ten smurf amplifiers (updated every 5 minutes)
(last update: 2016-01-17 23:31:02 CET)

Network	#Dups	#Incidents	Registered at	Home AS
212.1.130.0/24	38	0	1999-02-20 09:41	AS9105
204.158.83.0/24	27	0	1999-02-20 10:09	AS3354
209.241.162.0/24	27	0	1999-02-20 08:51	AS701
159.14.24.0/24	20	0	1999-02-20 09:39	AS2914
192.220.134.0/24	19	0	1999-02-20 09:38	AS685
204.193.121.0/24	19	0	1999-02-20 08:54	AS701
198.253.187.0/24	16	0	1999-02-20 09:34	AS22
164.106.163.0/24	14	0	1999-02-20 10:11	AS7066
12.17.161.0/24	13	0	2000-11-29 19:05	not-analyzed
199.98.24.0/24	13	0	1999-02-18 11:09	AS6199

2457713 networks have been probed with the SAR
56 of them are currently broken
193885 have been fixed after being listed here

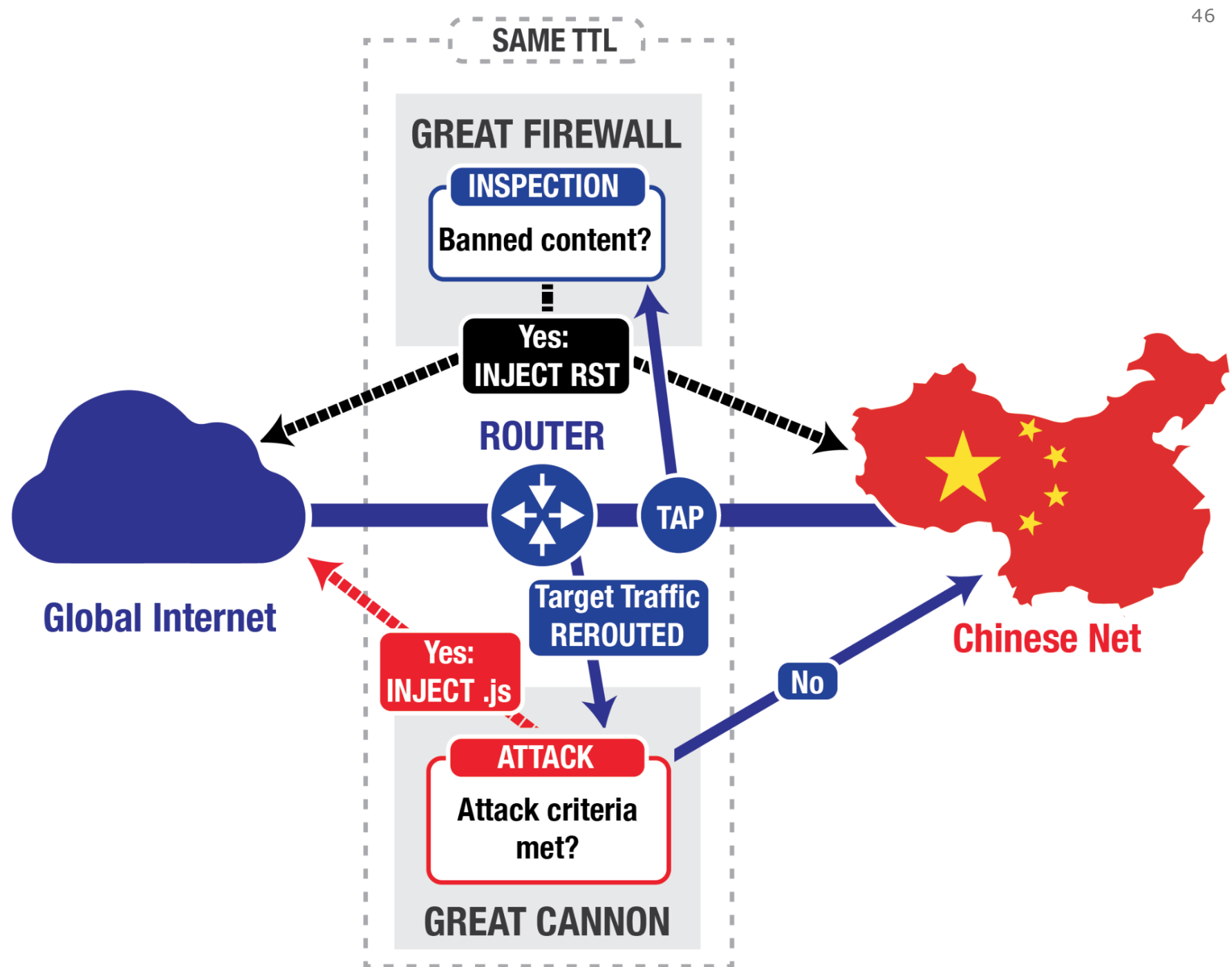
Distributed Denial of Service (DDoS)

A large number of machines work together to perform an attack that prevents valid users from accessing a service.

Common examples:

- Slashdot effect – a large number of valid users all try and access at once.
- Botnets
- Amazon web services

Great Cannon of China is a DDoS attack caused by a MITM attack

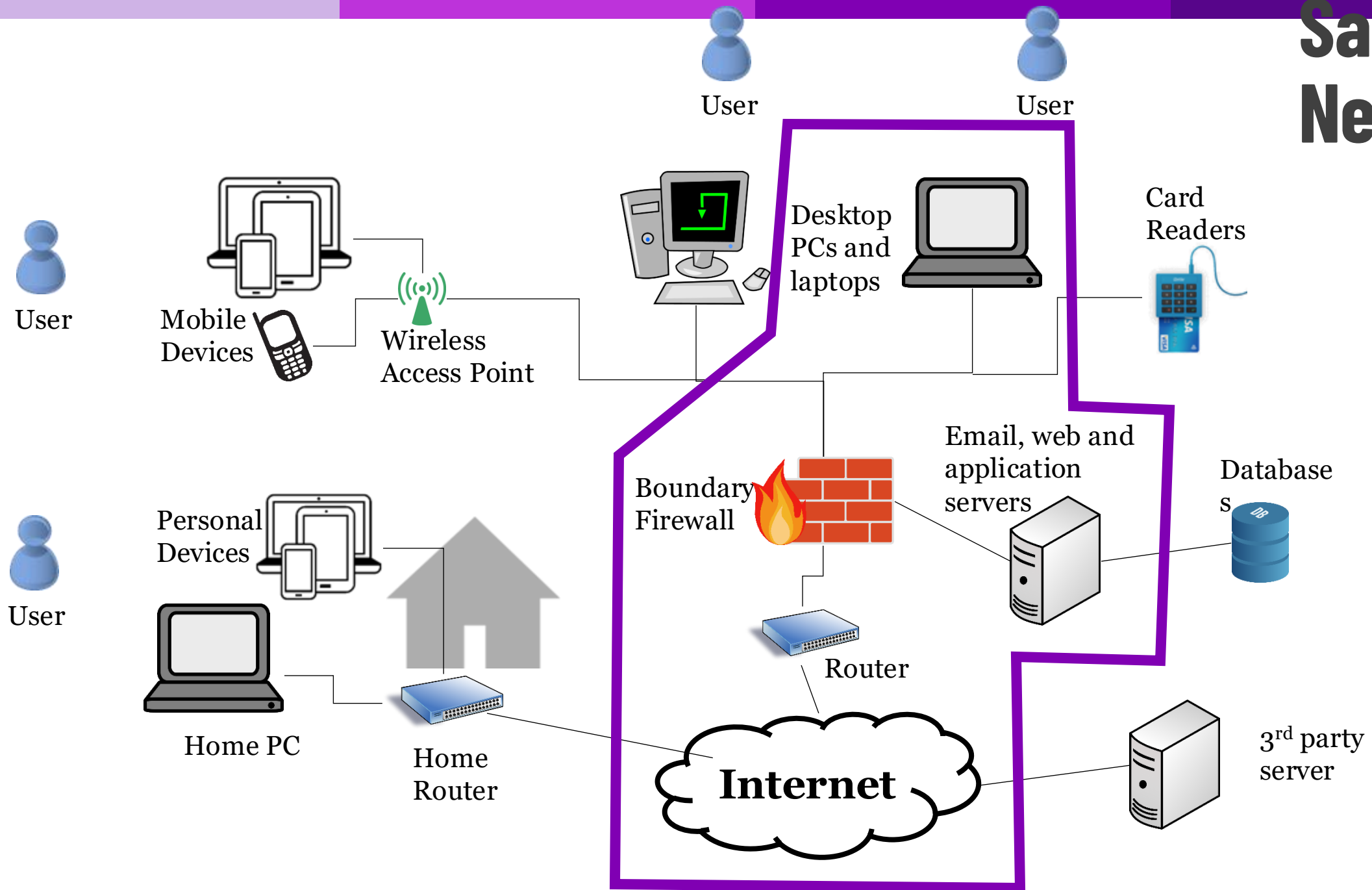


FIREWALLS

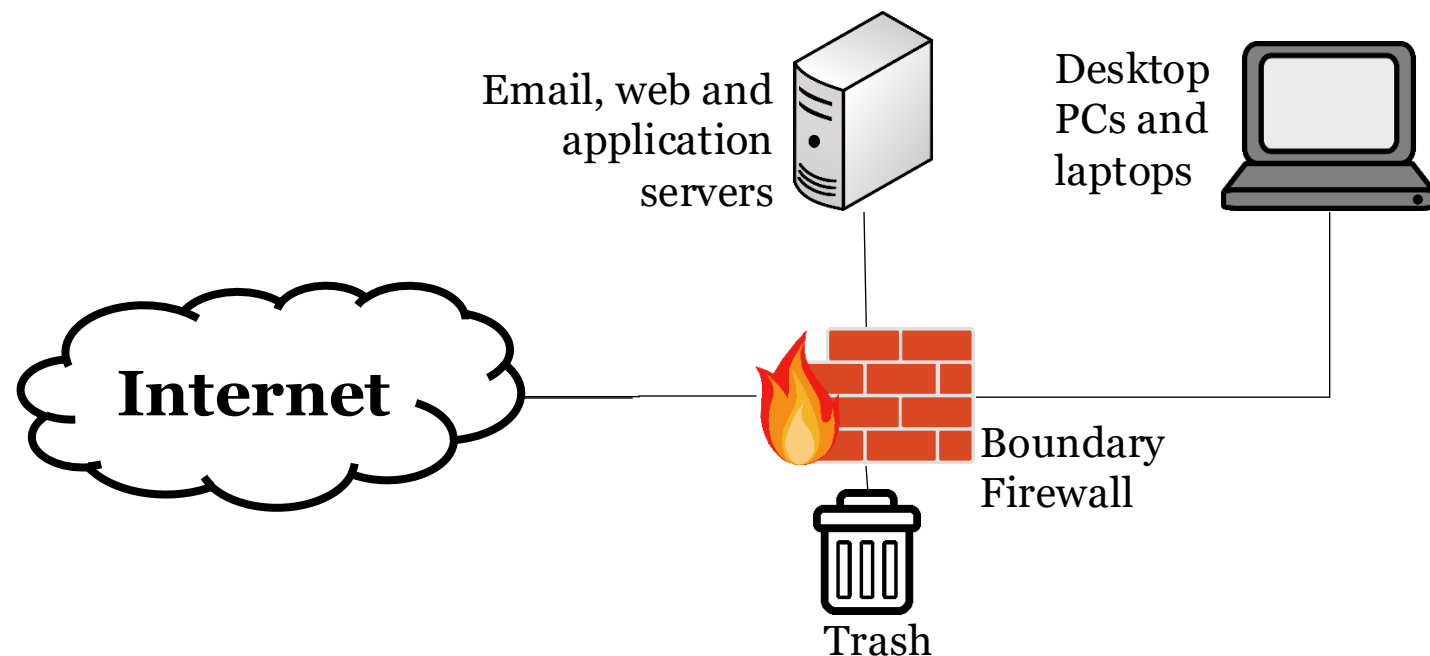
Firewalls

- Firewalls divide the untrusted outside of a network from the more trusted interior of a network
- Often they run on dedicated devices
 - Less possibilities for compromise – no compilers, linkers, loaders, debuggers, programming libraries, or other tools an attacker might use to escalate their attack
 - Easier to maintain few accounts
 - Physically divide the inside from outside of a network

Sample Network



- Questionable things come from the internet AND from the local network
- Firewall applies a set of rules
- Based on rules, it allows or denies the traffic
- Firewalls can also act as routers deciding where to send traffic




Rule	Type	Source Address	Destination Address	Destination Port	Action
1	TCP	*	192.168.1.*	22	Permit
2	UDP	*	192.168.1.*	69	Permit
3	TCP	192.168.1.*	*	80	Permit
4	TCP	*	192.168.1.18	80	Permit
5	UDP	*	192.168.1.*	*	Deny



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<!--DOCTYPE needs to be the very first thing on the page, or IE 6 goes
into quirks mode, rather than standards mode -->
<!--DOCUMENT STARTS-->
<!--START:ssi/doctype.inc-->
<html>
<head>
<!--END:ssi/doctype.inc-->
<!--TITLE HERE-->

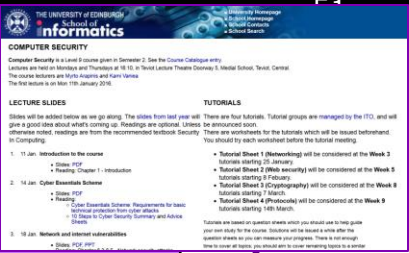
<!--START:ssi/bin/metadata-->
<!-- Metadata information automatically generated -->
<META NAME="DC.Title" CONTENT="Computer Security Course - University of Ed
<META NAME="DC.Creator" CONTENT="Neil Brown">
<META NAME="DC.Creator.Address" CONTENT="neilb@inf.ed.ac.uk">
```

Sender:
Apache

7	<div><div>Application</div></div> <div>Network process to application</div>
6	<div>Presentation</div> <div>Data representation and encryption</div>
5	<div>Session</div> <div>Interhost communication</div>
4	<div>Transport</div> <div>End-to-end connection and reliability</div>
3	<div>Network</div> <div>Path determination and IP (Logical Addressing)</div>
2	<div>Data Link</div> <div>MAC and LLC (Physical Addressing)</div>
1	<div>Physical</div> <div>Media, signal, and binary transmission</div>

Recipient:
Firefox user

7	<div><div>Application</div></div> <div>Network process to application</div>
6	<div>Presentation</div> <div>Data representation and encryption</div>
5	<div>Session</div> <div>Interhost communication</div>
4	<div>Transport</div> <div>End-to-end connection and reliability</div>
3	<div>Network</div> <div>Path determination and IP (Logical Addressing)</div>
2	<div>Data Link</div> <div>MAC and LLC (Physical Addressing)</div>
1	<div>Physical</div> <div>Media, signal, and binary transmission</div>





Sender:
Apache

7	Application Network process to application
6	Presentation Data representation and encryption
5	Session Interhost communication
4	Transport End-to-end connection and reliability
3	Network Path determination and IP (Logical Addressing)
2	Data Link MAC and LLC (Physical Addressing)
1	Physical Media, signal, and binary transmission

A firewall takes in network traffic and compares it to a set of rules. It must process several OSI levels to reach the data it needs.

For example, to filter out all traffic from IP 216.34.181.45 the packet needs to be processed through level 3 where IP addresses can be read.

Firewall

3	Network Path determination and IP (Logical Addressing)
2	Data Link MAC and LLC (Physical Addressing)
1	Physical Media, signal, and binary transmission



Recipient:
Firefox user

7	Application Network process to application
6	Presentation Data representation and encryption
5	Session Interhost communication
4	Transport End-to-end connection and reliability
3	Network Path determination and IP (Logical Addressing)
2	Data Link MAC and LLC (Physical Addressing)
1	Physical Media, signal, and binary transmission

Firewall ruleset from a custom home router

- Taken from an arstechnica article

```
root@ars-router: ~  
##### Service rules  
# OpenVPN  
-A INPUT -p udp -m udp --dport 1194 -j ACCEPT  
  
# ssh - drop any IP that tries more than 10 connections per minute  
-A INPUT -p tcp -m tcp --dport 22 -m state --state NEW -m recent --set --name DEFAULT --mask 255.255.255.255 --rsource  
-A INPUT -p tcp -m tcp --dport 22 -m state --state NEW -m recent --update --seconds 60 --hitcount 11 --name DEFAULT --mask 255.255.255.255 --rsource -j LOGDROP  
-A INPUT -p tcp -m tcp --dport 22 -j ACCEPT  
  
# www - accept from LAN  
-A INPUT -i p1p1 -p tcp -m tcp --dport 80 -j ACCEPT  
-A INPUT -i p1p1 -p tcp -m tcp --dport 443 -j ACCEPT  
  
# DNS - accept from LAN  
-A INPUT -i p1p1 -p tcp --dport 53 -j ACCEPT  
-A INPUT -i p1p1 -p udp --dport 53 -j ACCEPT  
  
# default drop because I'm awesome  
-A INPUT -j DROP  
  
##### forwarding ruleset
```

Image: <http://arstechnica.co.uk/gadgets/2016/01/numbers-dont-lie-its-time-to-build-your-own-router/>

There are many types of Firewalls

Key differences include:

- How implemented
 - Software – slower, easier to deploy on personal computers
 - Hardware – faster, somewhat safer, harder to add in
- Number of OSI levels of processing required
 - Packet size (level 1)
 - MAC (level 2) and IP (level 3) filtering
 - Port filtering (level 3)
 - Deep packet (level 4+)

Today we will talk about:

- Packet filtering gateway
- Stateful inspection firewall
- Application proxy
- Personal firewalls

Packet filtering gateway or screening router

- Simplest – compares information found in the headers to the policy rules
- Operate at OSI level 3
- Source addresses and ports can be forged, which a packet filter cannot detect
- Design is simple, but tons of rules are needed, so it is challenging to maintain

Stateful inspection firewall

- Maintains state from one packet to another
- Similar to a packet filtering gateway, but can remember recent events
- For example, if a outside host starts sending packets to many internal destination ports (aka a port scan) a stateful firewall would record the number of ports probed and once it is over the threshold specified in the policy it would block all further traffic

Port scan

- An attacker is looking for applications listening on ports
- A single IP address (right) is contacting many ports (left) to see if any respond

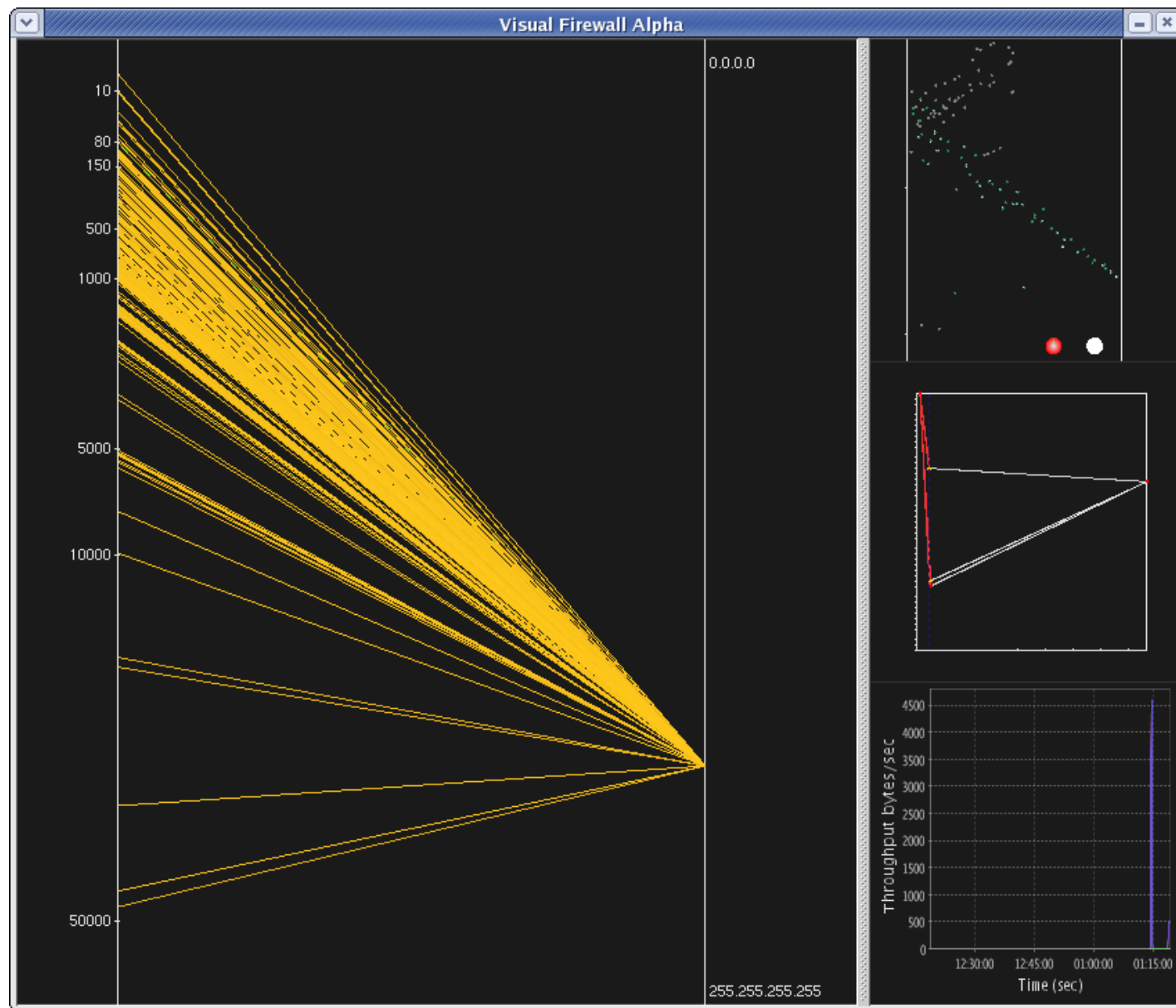
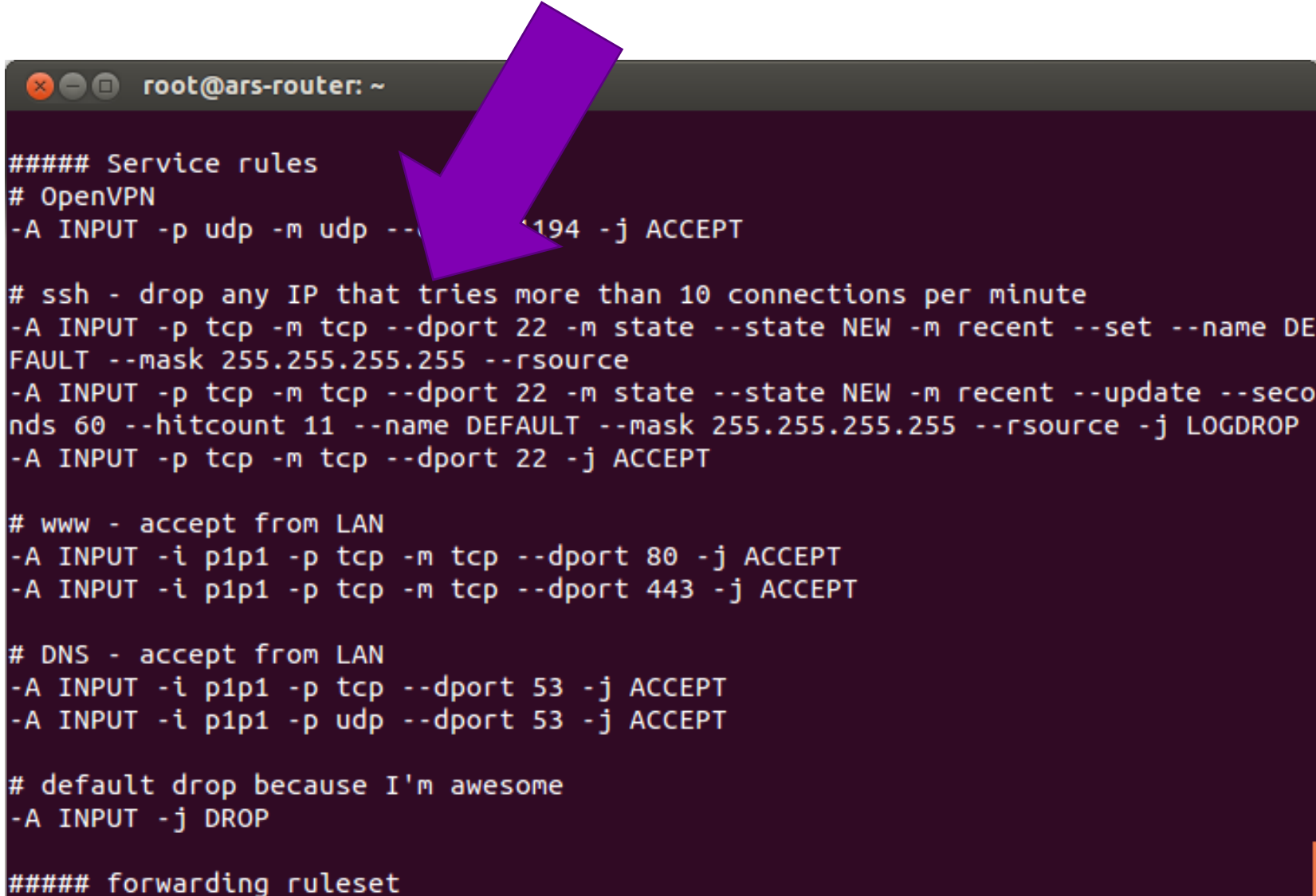


Image: <http://chrislee.dhs.org/projects/visualfirewall.html>

Firewall ruleset from a custom home router

- Taken from an ARSTechnica article



```
root@ars-router: ~  
##### Service rules  
# OpenVPN  
-A INPUT -p udp -m udp --dport 194 -j ACCEPT  
  
# ssh - drop any IP that tries more than 10 connections per minute  
-A INPUT -p tcp -m tcp --dport 22 -m state --state NEW -m recent --set --name DEFAULT --mask 255.255.255.255 --rsource  
-A INPUT -p tcp -m tcp --dport 22 -m state --state NEW -m recent --update --seconds 60 --hitcount 11 --name DEFAULT --mask 255.255.255.255 --rsource -j LOGDROP  
-A INPUT -p tcp -m tcp --dport 22 -j ACCEPT  
  
# www - accept from LAN  
-A INPUT -i p1p1 -p tcp -m tcp --dport 80 -j ACCEPT  
-A INPUT -i p1p1 -p tcp -m tcp --dport 443 -j ACCEPT  
  
# DNS - accept from LAN  
-A INPUT -i p1p1 -p tcp --dport 53 -j ACCEPT  
-A INPUT -i p1p1 -p udp --dport 53 -j ACCEPT  
  
# default drop because I'm awesome  
-A INPUT -j DROP  
  
##### forwarding ruleset
```

Image: <http://arstechnica.co.uk/gadgets/2016/01/numbers-dont-lie-its-time-to-build-your-own-router/>

Application proxy

- Simulates the (proper) effects of an application at OSI level 7
- Effectively a protective Man In The Middle that screens information at an application layer (OSI 7)
- Allows an administrator to block certain application requests.
- For example:
 - Block all web traffic containing certain words
 - Remove all macros from Microsoft Word files in email
 - Prevent anything that looks like a credit card number from leaving a database

Personal firewalls

- Runs on the workstation that it protects (software)
- Provides basic protection, especially for home or mobile devices
- Malicious software can disable part or all of the firewall
- Any rootkit type software can disable the firewall

Think-pair-share

Imagine you want to put a firewall in front of the email server

- Why is deep packet inspection easier to do on email than on normal network traffic?
- As a malicious actor, how might I go around your email firewall?

NETWORK ADDRESS TRANSLATION (NAT)

Looking at the IP
address of my
laptop which is
connected to
UWaterloo WIFI.

```
C:\Users\kamiv>ipconfig
```

```
Windows IP Configuration
```

```
Unknown adapter Local Area Connection:
```

```
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
```

```
Wireless LAN adapter Local Area Connection* 1:
```

```
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
```

```
Wireless LAN adapter Local Area Connection* 2:
```

```
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
```

```
Wireless LAN adapter Wi-Fi:
```

```
Connection-specific DNS Suffix . : uwaterloo.ca
IPv6 Address. . . . . : 2620:101:f000:700:3fff:ffff:1
8:cef7
```

```
Link-local IPv6 Address . . . . . : fe80::8218:8dee:f17e:d77%10
```

```
IPv4 Address. . . . . : 10.32.15.86
```

```
Subnet Mask . . . . . : 255.255.128.0
```

```
Default Gateway . . . . . : 10.32.0.1
```

```
Ethernet adapter Bluetooth Network Connection:
```

```
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
```

My computer as seen from a remote server

(<http://www.hashemian.com/whoami/>)

My computer claimed it was:
10.32.15.86

What happened?

by Robert Hashemian // **Your IP: 129.97.124.26**

What Is My IP Address? IPv4 Lookup, IPv6 Lookup, Country Lookup, Whoami (Who am I?) - Your/VPN/Proxy Online Information, Browser Headers, DNS, Whois, SSL/TLS, ISP ASN

Your/VPN/Proxy **IPv4** address: **129.97.124.26**

Your/VPN/Proxy **IPv6** address: **N/A**

Your/VPN/Proxy **Country**: **Canada**

Your local LAN IP:

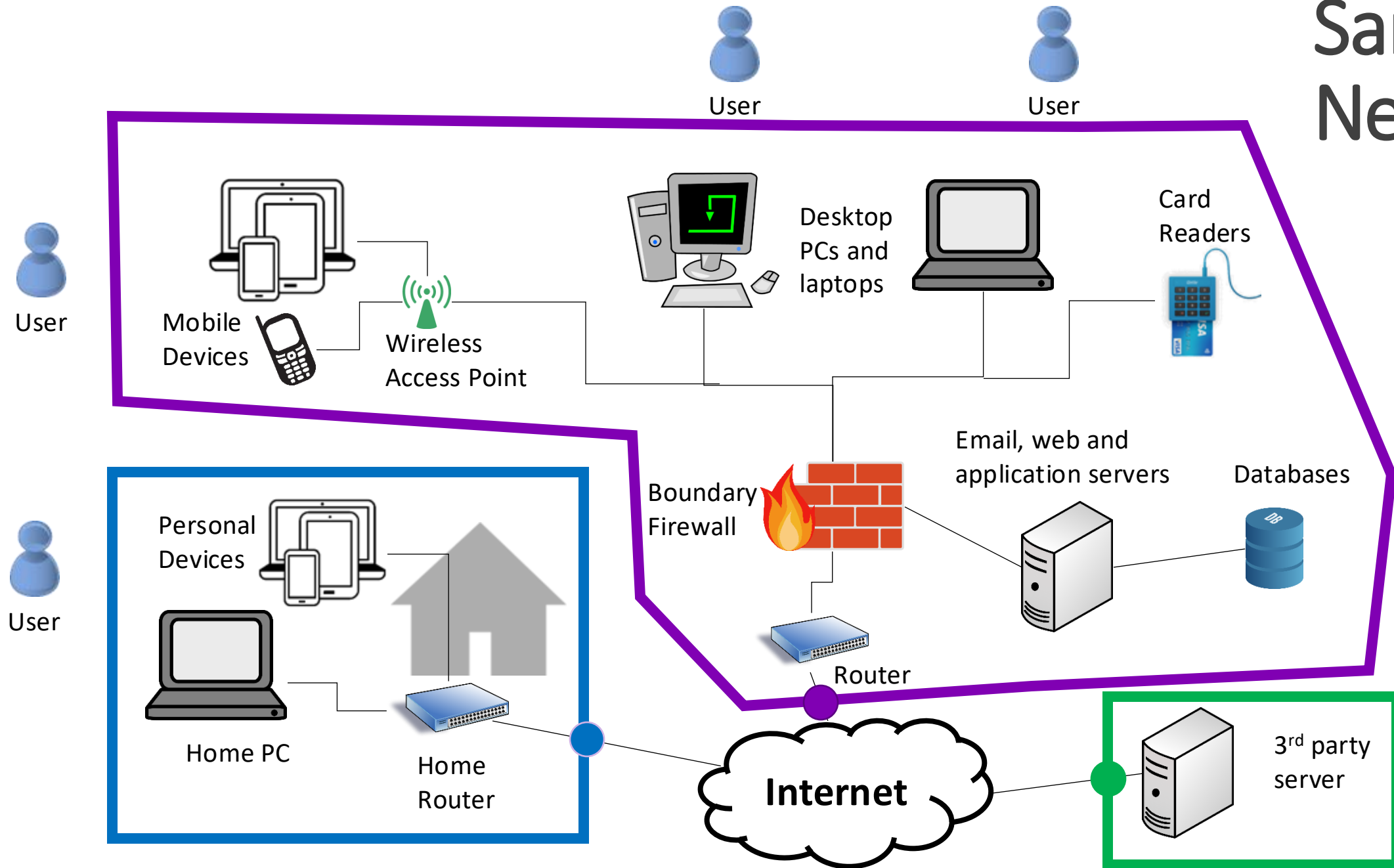
Your ISP's **Network Number / ASN** Info:

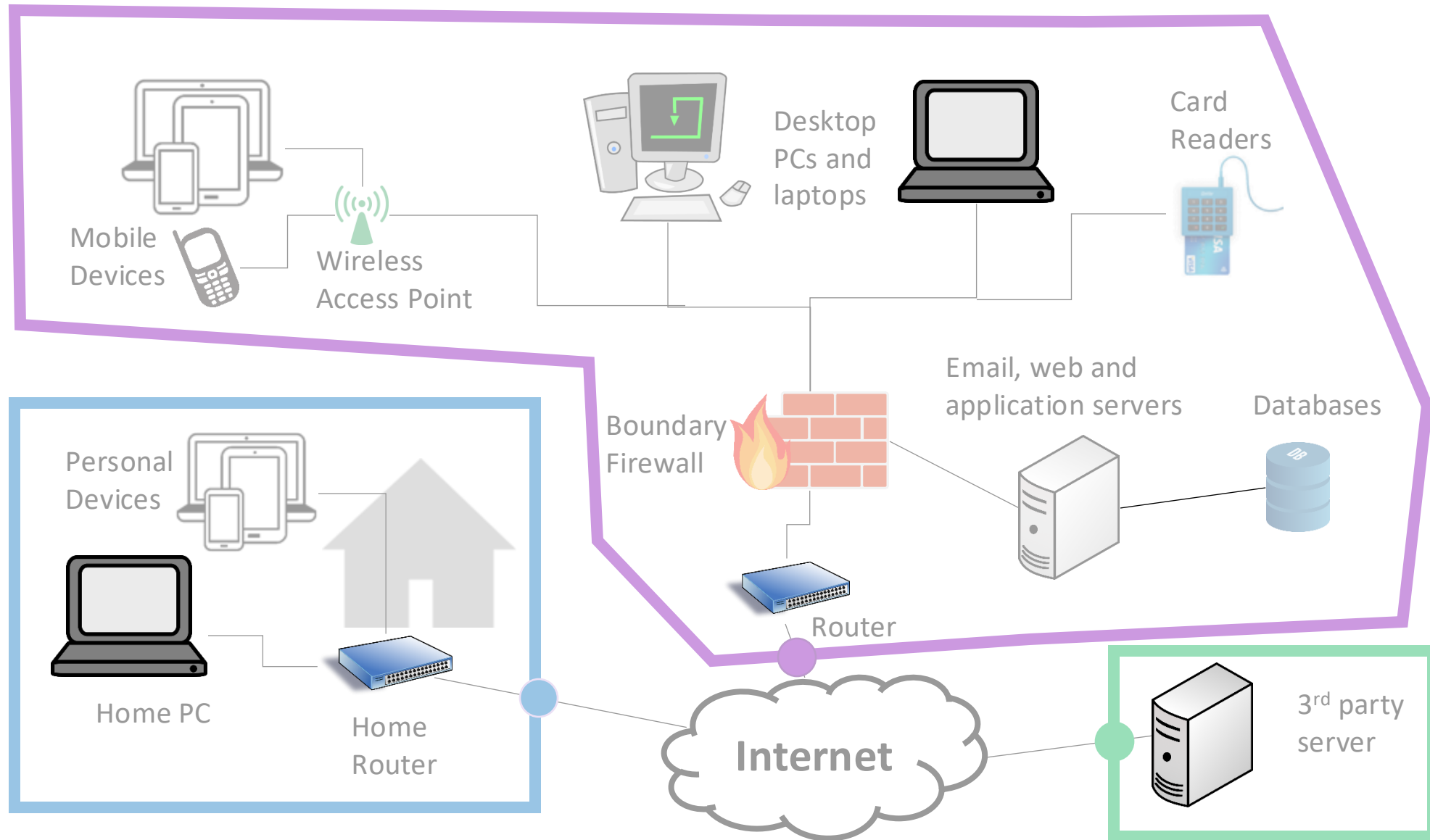
```
[Querying whois.cymru.com]
AS: 12093 ↗
IP: 129.97.124.26
BGPPrefix: 129.97.0.0/16
CC: CA
Registry: arin
Allocated: 1987-10-25
ASName: UWATERLOO,CA
```

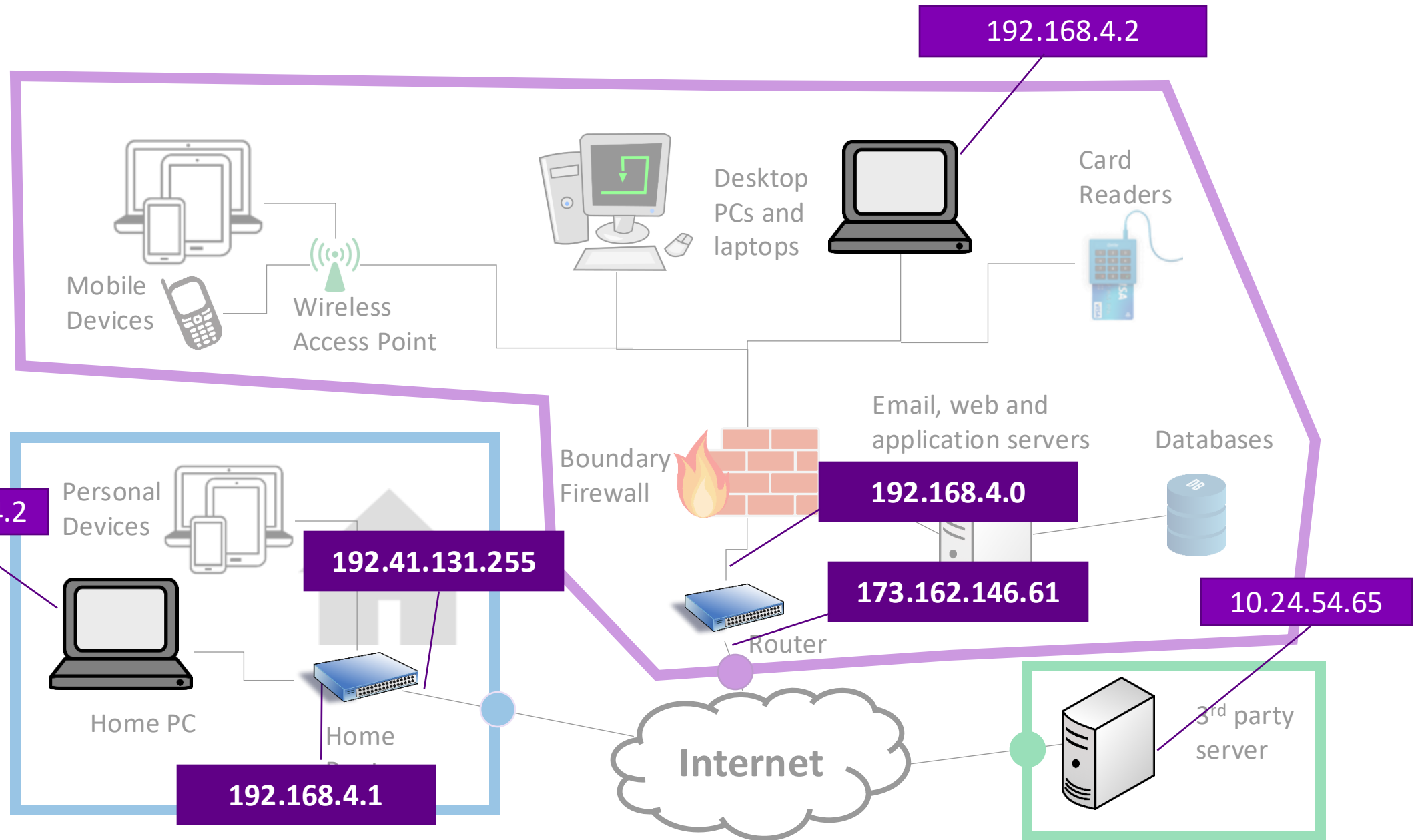
IPv4 and address space exhaustion

- Version 4 of the Internet Protocol
 - 192.168.2.6
- There are less than 4.3 billion IPv4 addresses available
- We do not have enough addresses for every device on the planet
- Answer: Network Address Translation
 - Internal IP different than external IP
 - Border router maps between its own IP and the internal ones

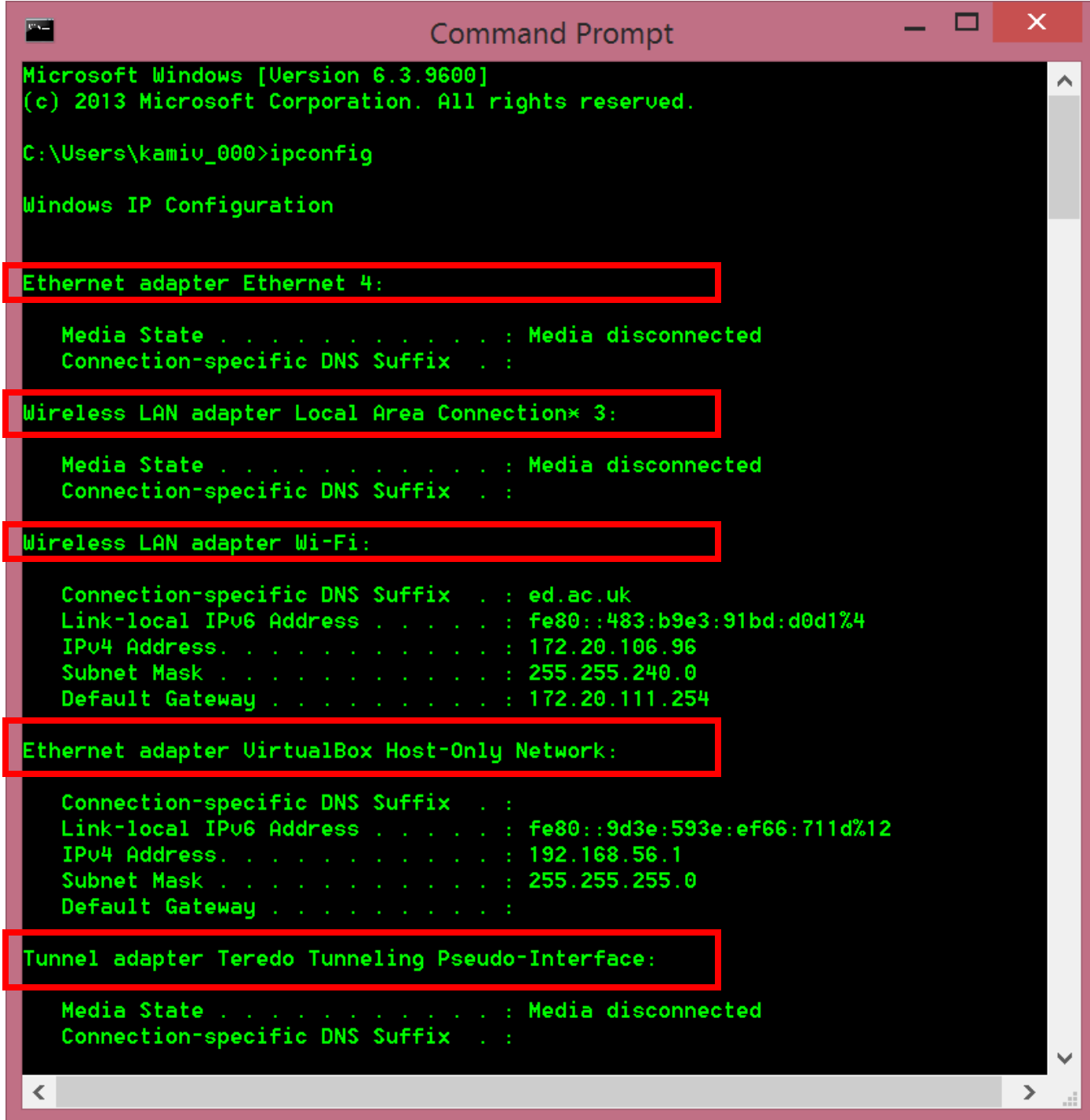
Sample Network







My laptop can have multiple IPs and bridge networks too. Here it shows IPs for both my VirtualBox and my WIFI.



```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\kamiu_000>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet 4:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 3:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Wi-Fi:

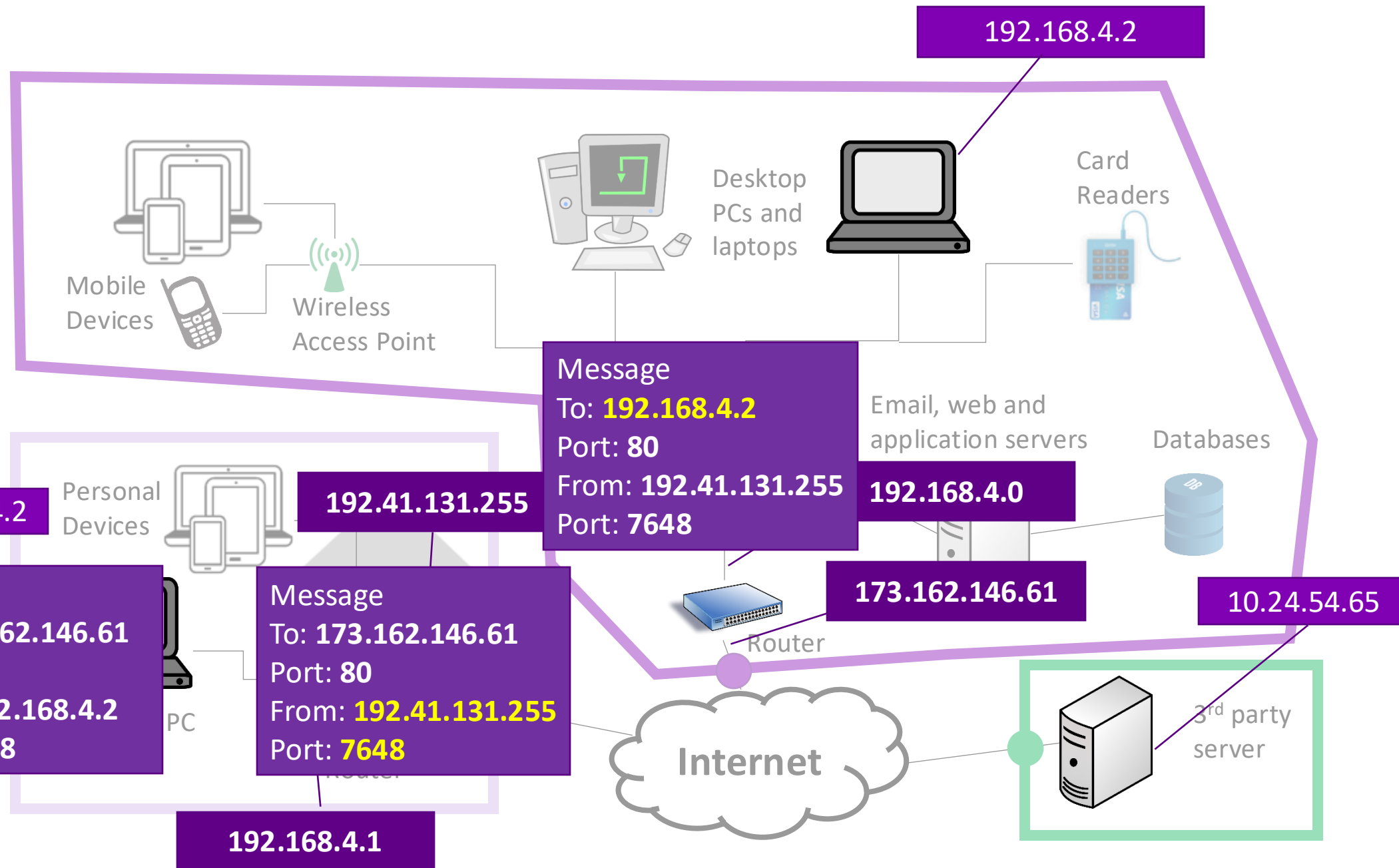
    Connection-specific DNS Suffix  . : ed.ac.uk
    Link-local IPv6 Address . . . . . : fe80::483:b9e3:91bd:d0d1%4
    IPv4 Address. . . . . : 172.20.106.96
    Subnet Mask . . . . . : 255.255.240.0
    Default Gateway . . . . . : 172.20.111.254

Ethernet adapter VirtualBox Host-Only Network:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::9d3e:593e:ef66:711d%12
    IPv4 Address. . . . . : 192.168.56.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . :

Tunnel adapter Teredo Tunneling Pseudo-Interface:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :
```



Think-pair-share

- Internet of Things (IoT) security cameras commonly advertise that you have the ability to see the video feed from anywhere using their app
- What would they need to do to technically implement this?
- Advanced: How do you think they are actually accomplishing this?

QUESTIONS