

Network security: Networking Principles

COMPUTER SECURITY
MARKULF KOHLWEISS

Some slides adapted from those by Myrto Arapinis, Kami Vaniea, and Roberto Tamassia



Network Communication

- Communication in modern networks is characterized by the following fundamental principles
 - Packet switching
 - Stack of layers
 - Encapsulation

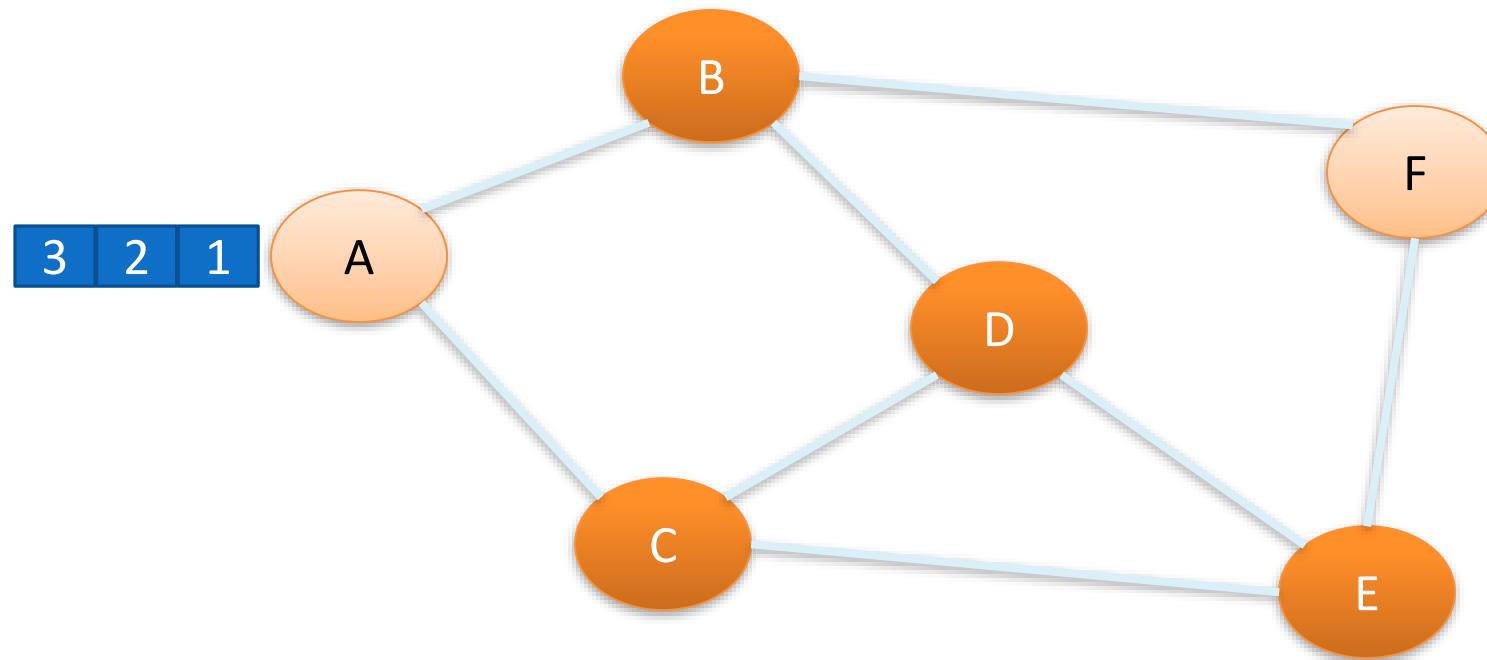


Packet Switching

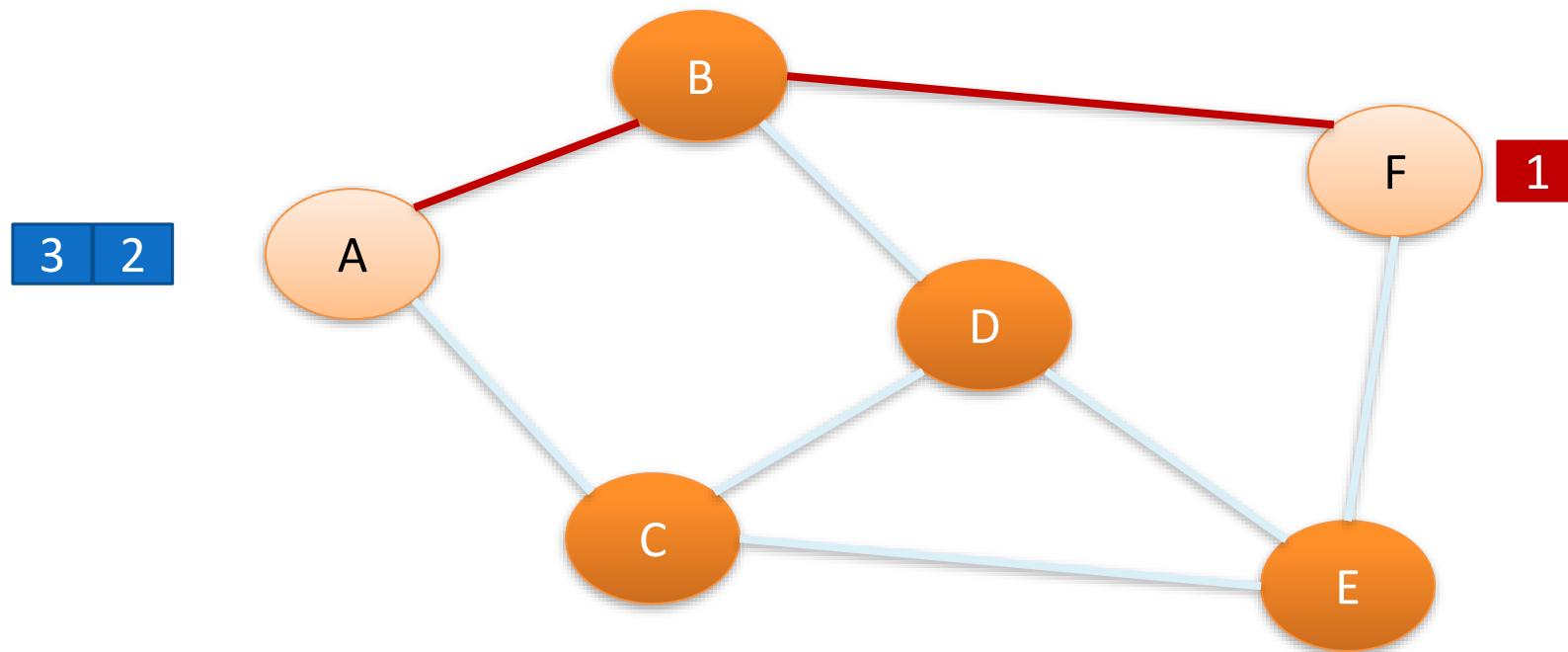
- Data split into **packets**
- Each packet is
 - Transported **independently** through network
 - Handled on a **best efforts** basis by each device
- Packets may
 - Follow different routes between the same endpoints
 - Be dropped by an intermediate device and never delivered



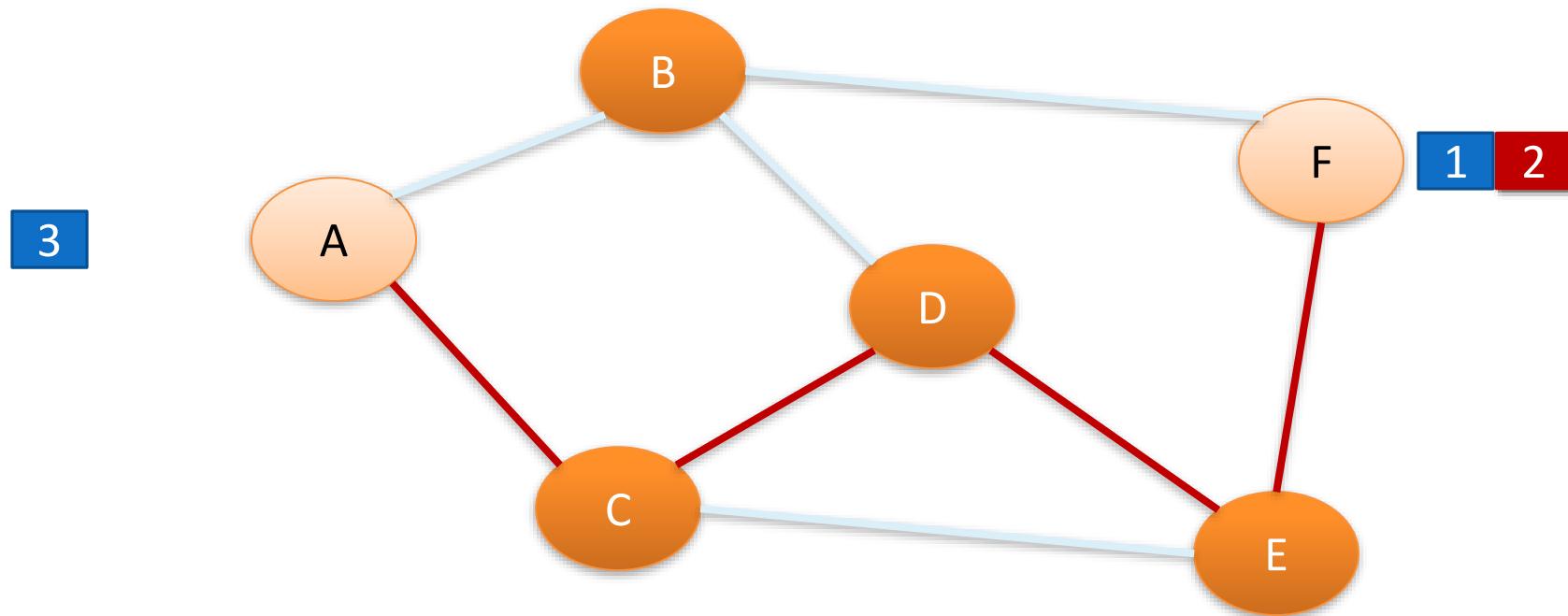
Packet Switching



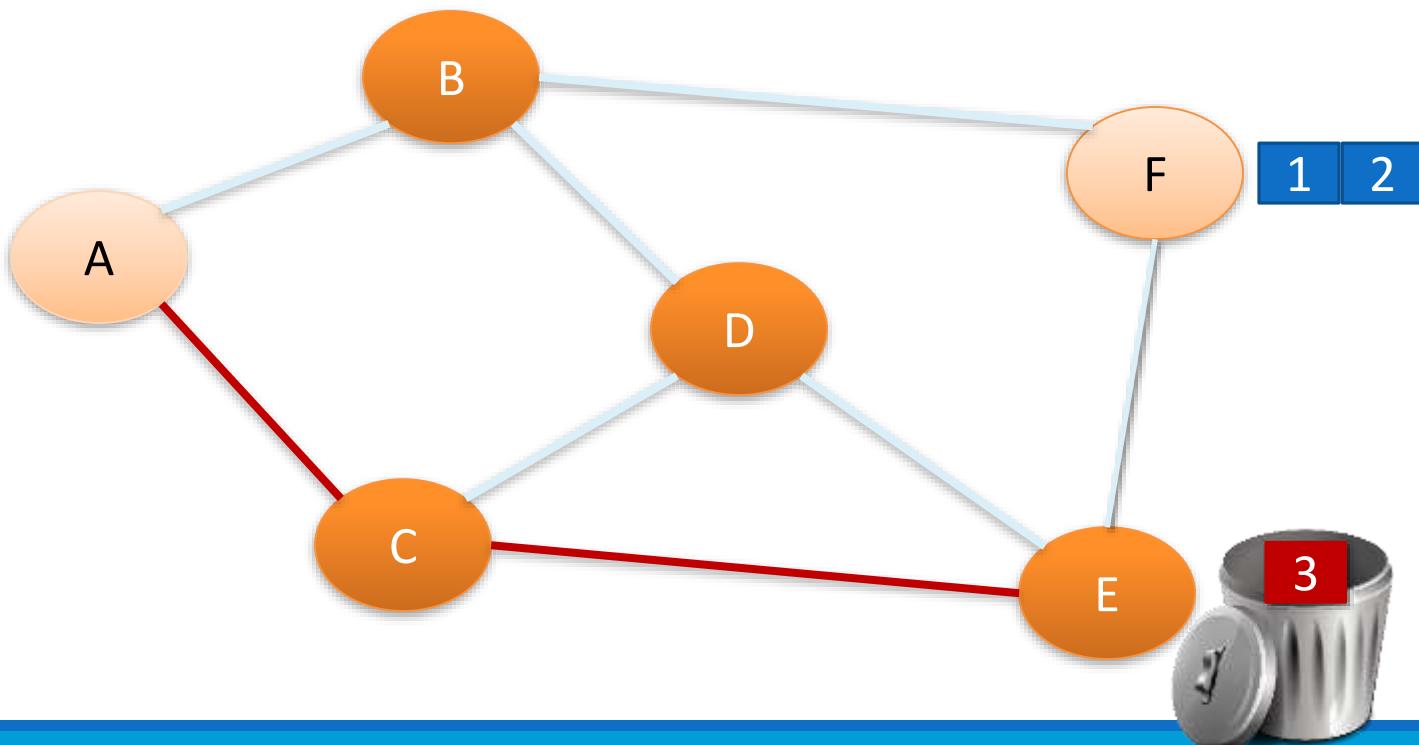
Packet Switching



Packet Switching



Packet Switching

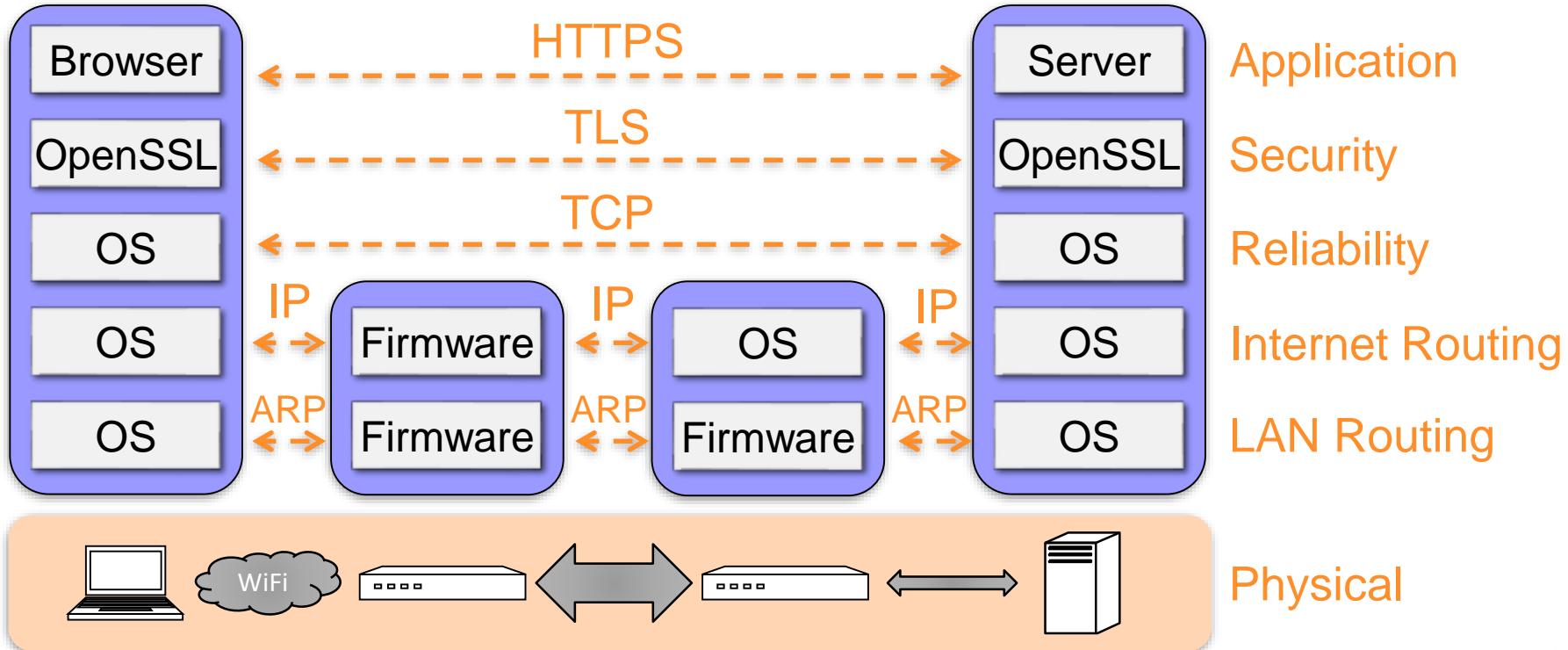


Stack of Layers

- Network communication models use a **stack of layers**
 - Higher layers use services of lower layers
 - Physical channel at the bottommost layer
- A network device implements several layers
- A communication channel between two devices is established for each layer
 - **Actual** channel at the bottom layer
 - **Virtual** channel at higher layers

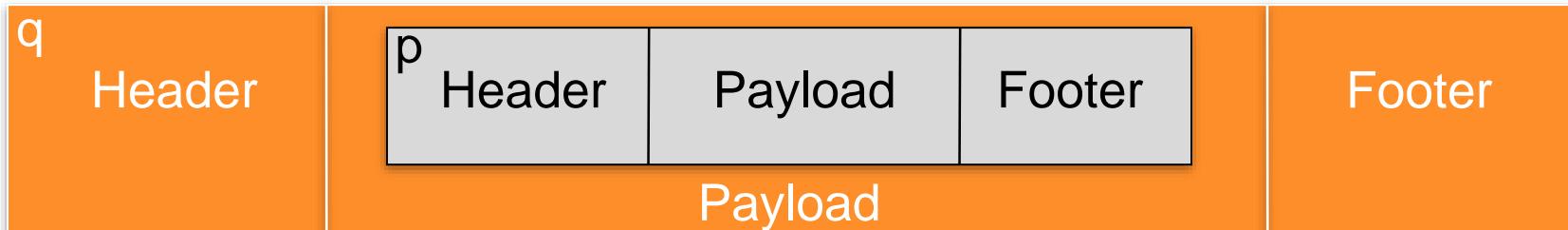


Internet Stack (simplified)

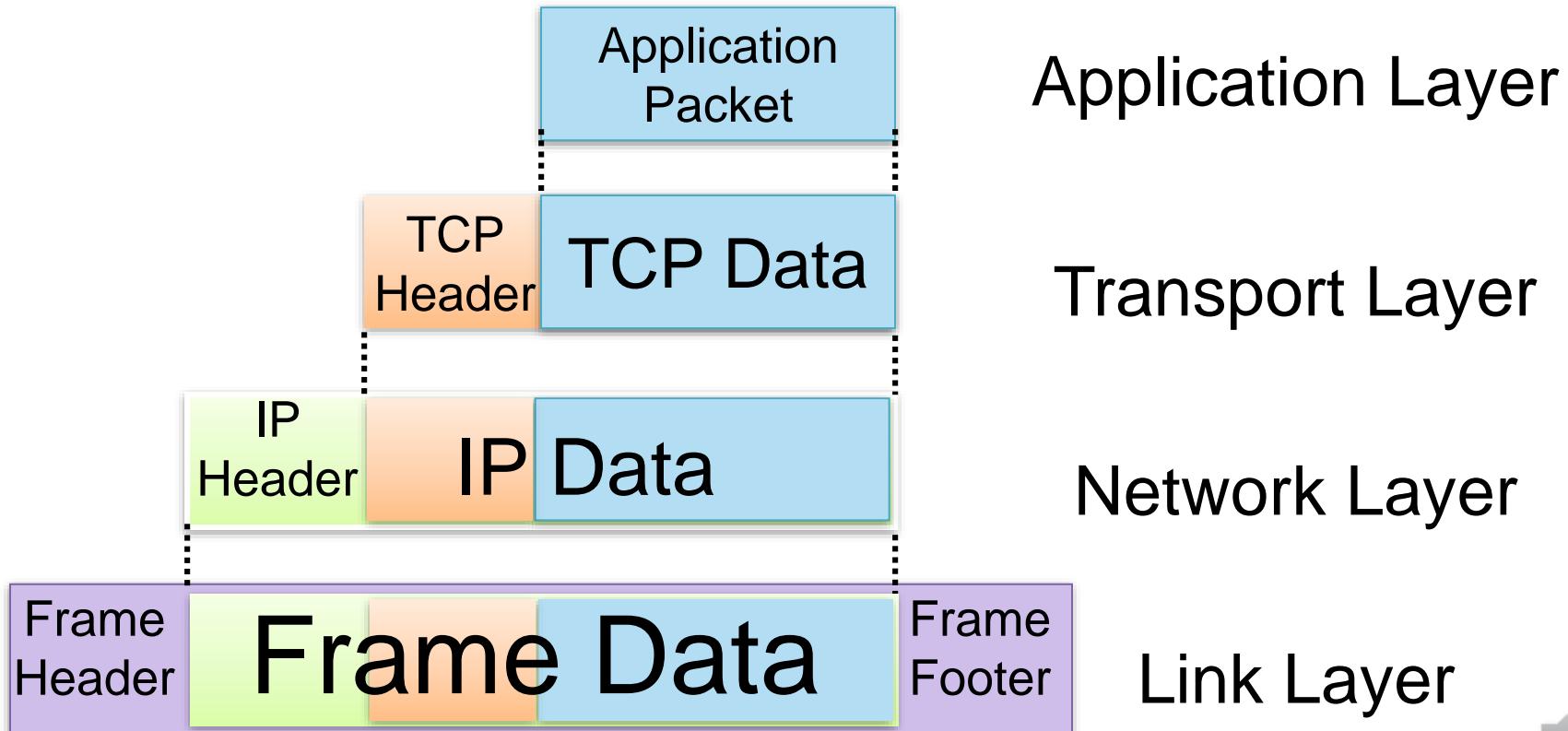


Encapsulation

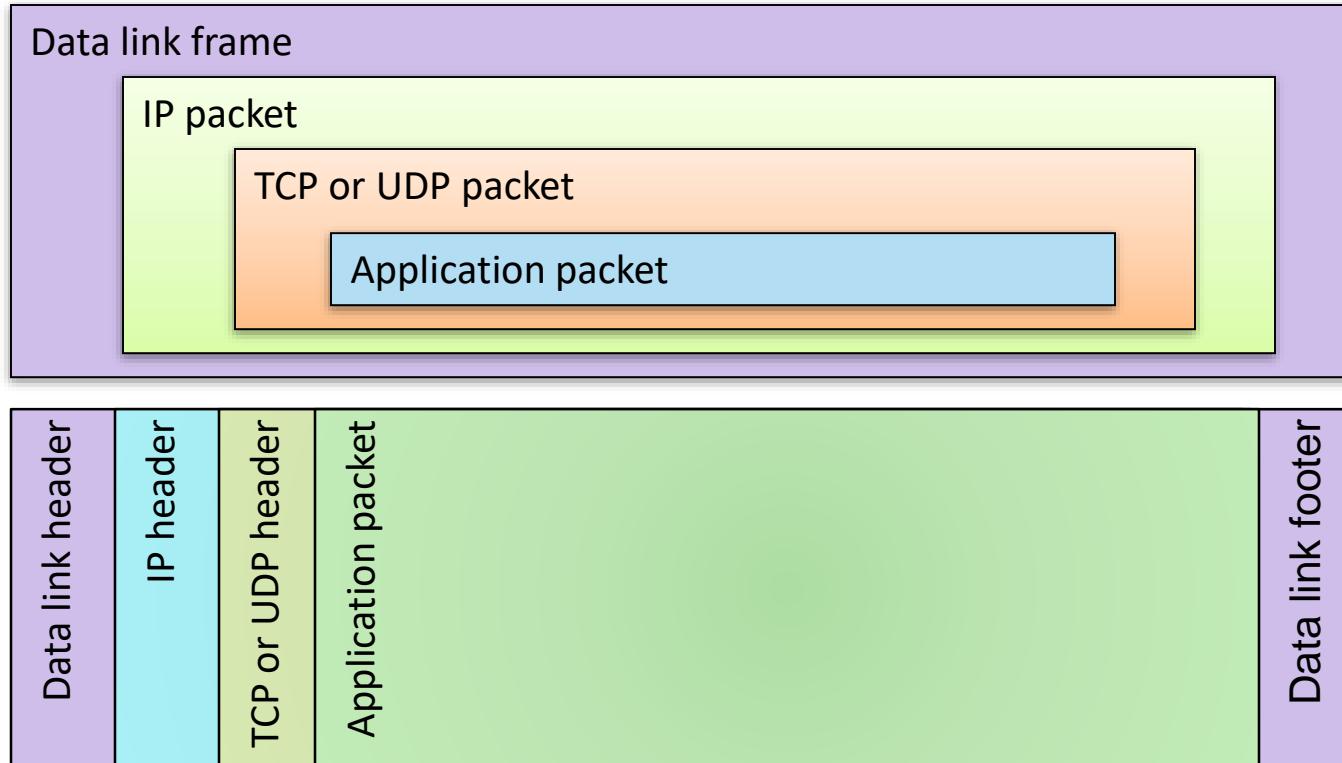
- A packet typically consists of
 - Control information: **header** and **footer**
 - Data: **payload**
- A protocol P uses the services of another protocol Q through **encapsulation**
- A packet p of P is encapsulated into a packet q of Q
- The payload of q is p
- The control information of q is derived from that of p



Internet Packet Encapsulation

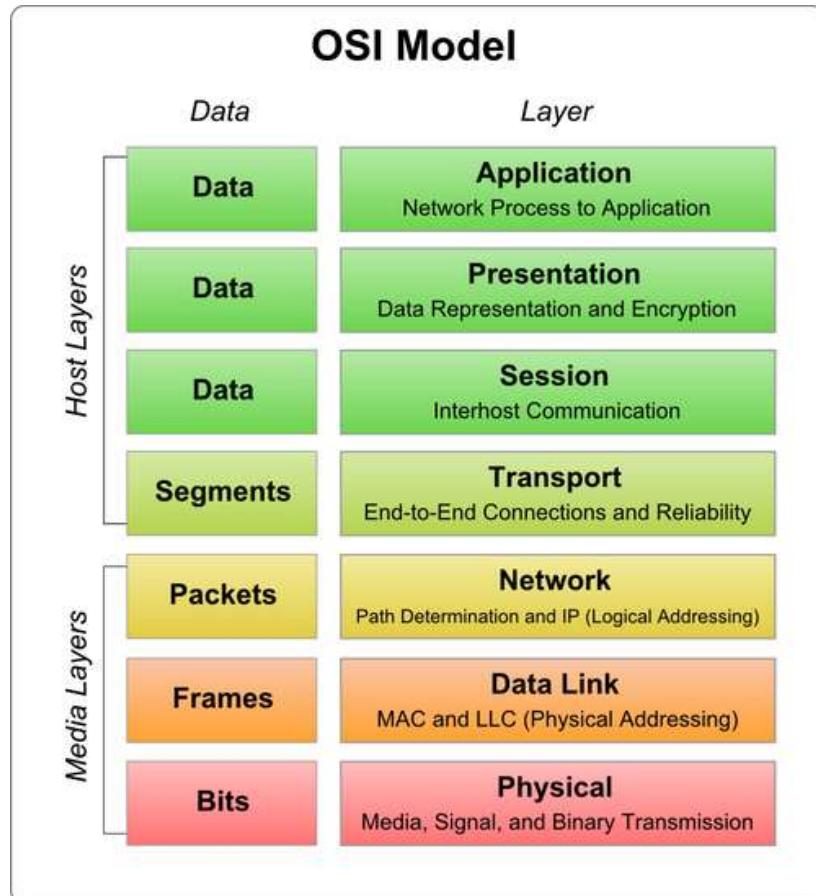


Internet Packet Encapsulation

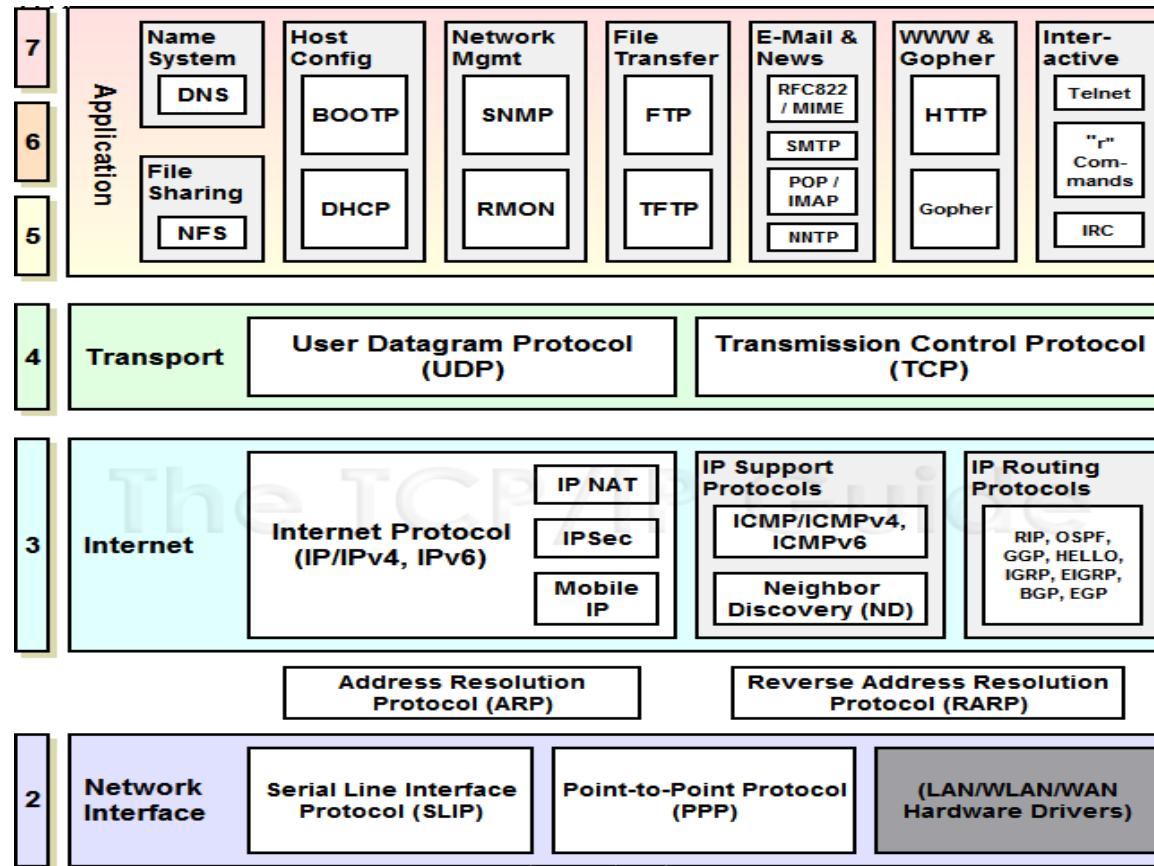


The OSI Model

- The **OSI** (Open System Interconnect) Reference Model is a network model consisting of seven layers
- Created in 1983, OSI is promoted by the International Standard Organization (**ISO**)



TCP/IP Model Mapped onto OSI



Network Interfaces

- Network interface: device connecting a computer to a network
 - Ethernet card
 - WiFi adapter
 - DSL modem
- A computer may have multiple network interfaces
- Packets transmitted between network interfaces
- Most local area networks, (including Ethernet and WiFi) broadcast frames



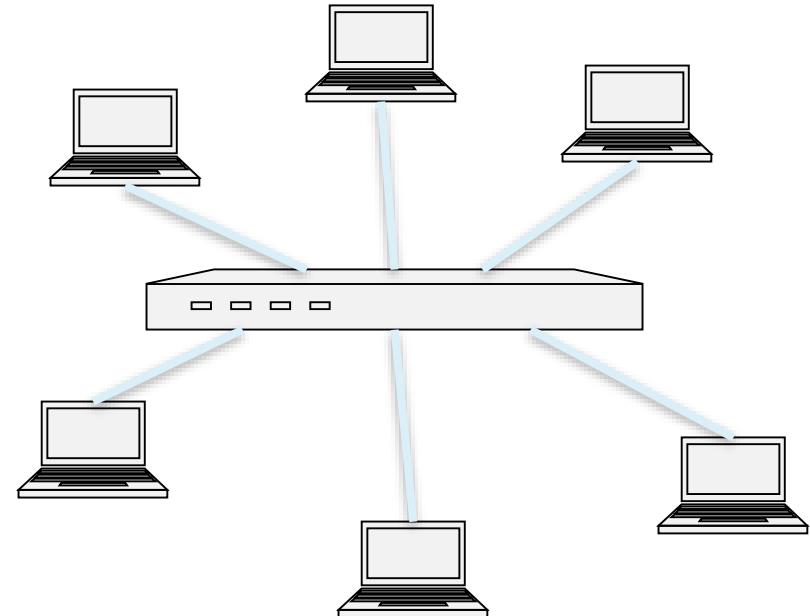
Media Access Control (MAC) Addresses

- Most network interfaces come with a predefined MAC address
- A MAC address is a 48-bit number usually represented in hex
 - E.g., 00-1A-92-D4-BF-86
- The first three octets of any MAC address are IEEE-assigned Organizationally Unique Identifiers
 - E.g., Cisco 00-1A-A1, D-Link 00-1B-11, ASUSTek 00-1A-92, 00-0a-95 ??????
- The next three can be assigned by organizations as they please, with uniqueness being the only constraint



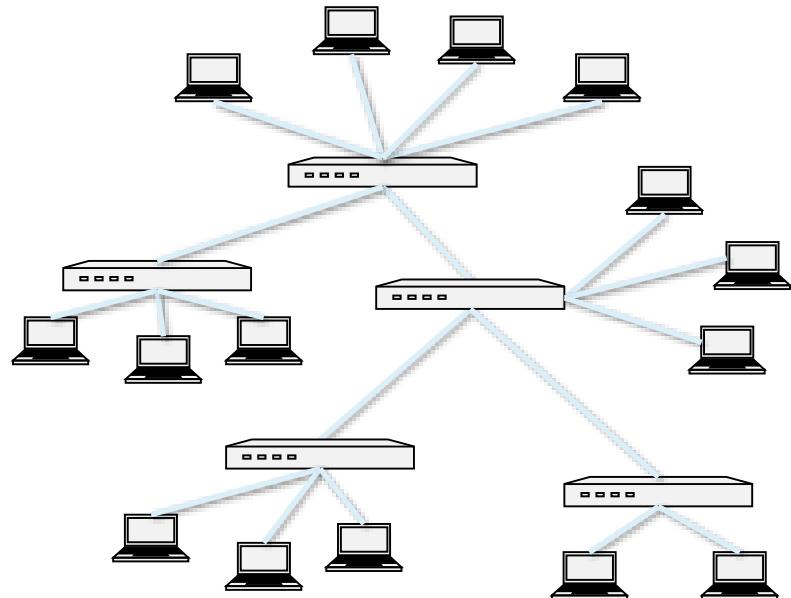
Switch

- A switch performs routing in a local area network
 - Operates at the link layer
 - Has multiple interfaces, each connected to a computer/segment
- Operation of a switch
 - Learn the MAC address of each computer connected to it
 - Forward frames only to the destination computer

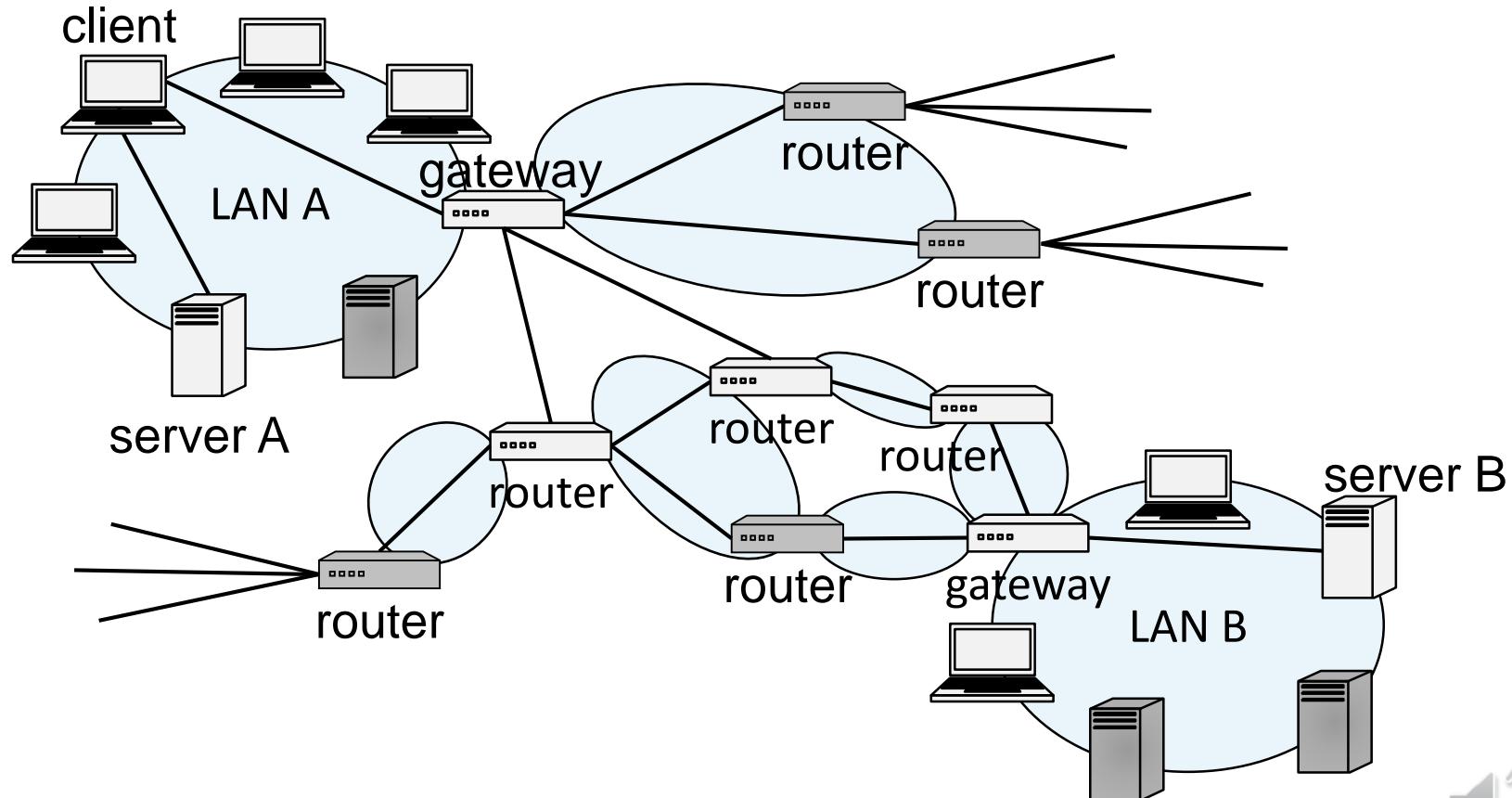


Combining Switches

- Switches can be arranged into a **tree**
- Each forwards frames for the MAC addresses of the machines in the segments (subtrees) connected to it
- Frames to unknown MAC addresses are broadcast
- Frames to MAC addresses in the same segment as the sender are ignored

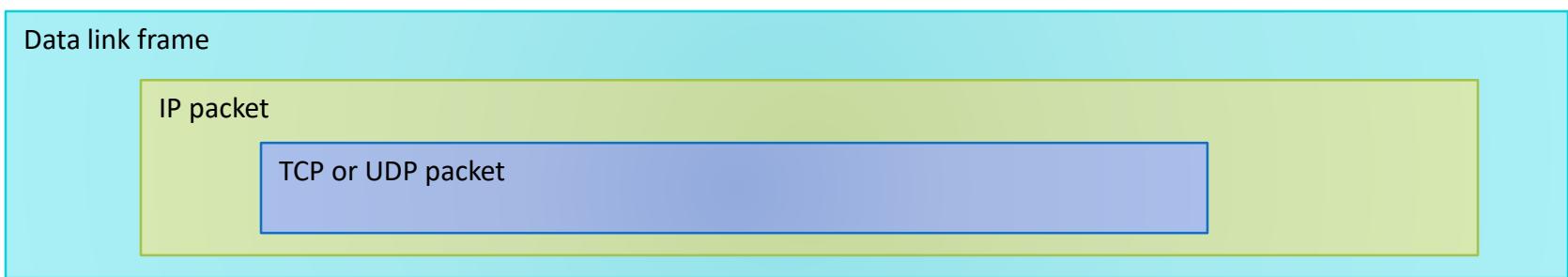


The Internet



Internet Protocol Functions

- **Addressing:** In order to deliver data, IP needs to be aware of where to deliver data to, and hence includes addressing systems
- **Routing:** IP might be required to communicate across networks, and communicate with networks not directly connected to the current network



IP Addresses and Packets

- IP addresses
 - IPv4: 32-bit addresses
 - IPv6: 128-bit addresses
- Address subdivided into **network**, **subnet**, and **host**
 - E.g., **128.148.32.110**
- Broadcast addresses
 - E.g., **128.148.32.255**
- Private networks
 - not routed outside of a LAN
 - **10.0.0.0/8**
 - **172.16.0.0/12**
 - **192.168.0.0/16**
- IP header includes
 - Source address
 - Destination address
 - Packet length (up to 64KB)
 - Time to live (up to 255)
 - IP protocol version
 - Fragmentation information
 - Transport layer protocol information (e.g., TCP)

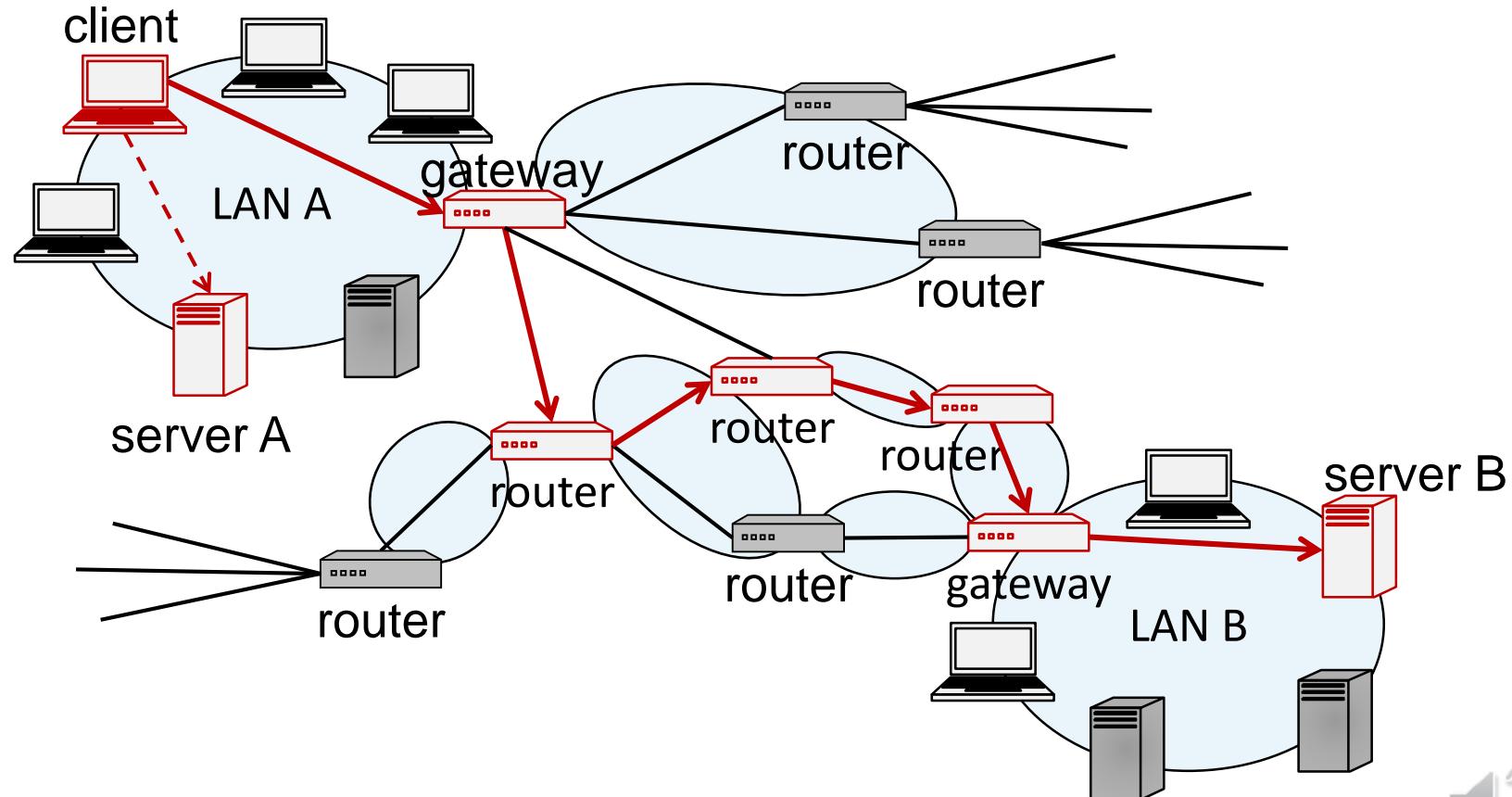


IP Routing

- A router bridges two or more networks
 - Operates at the network layer
 - Maintains tables to forward packets to the appropriate network
 - Forwarding decisions based solely on the destination address
- Routing table
 - Maps ranges of addresses to LANs or other gateway routers



Routing Examples

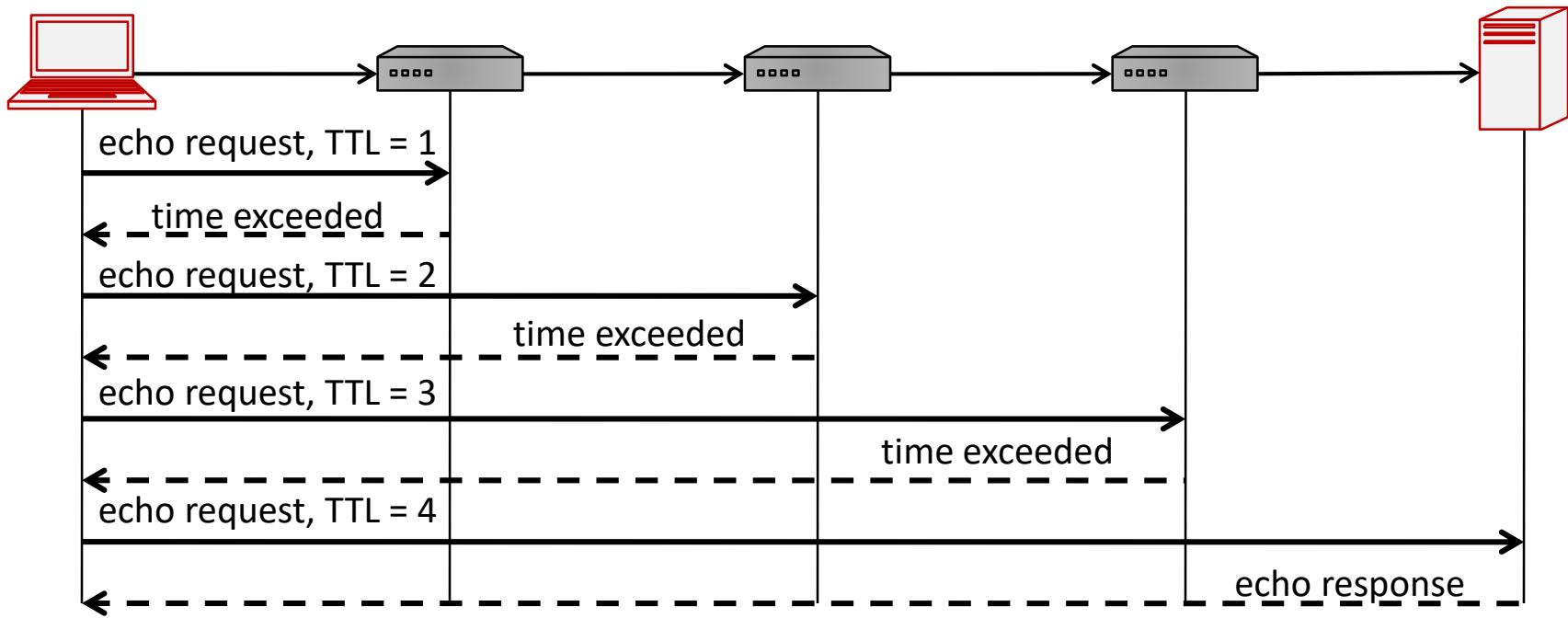


Internet Routes

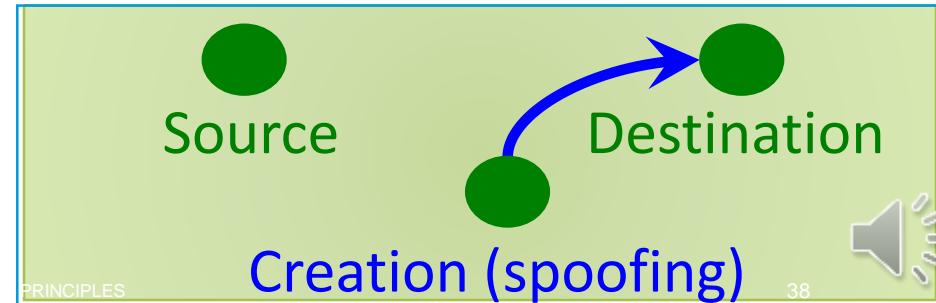
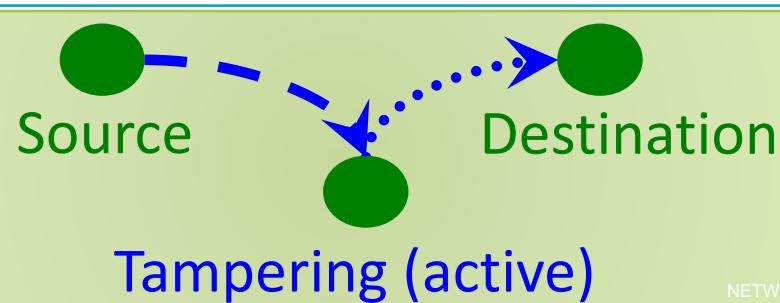
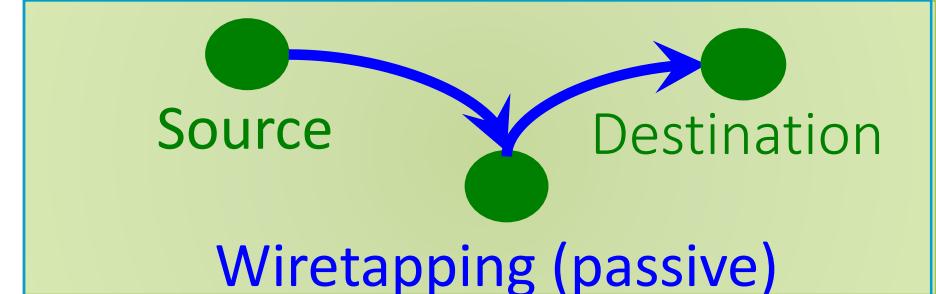
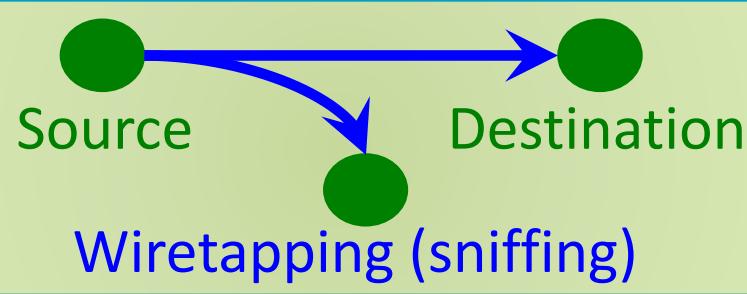
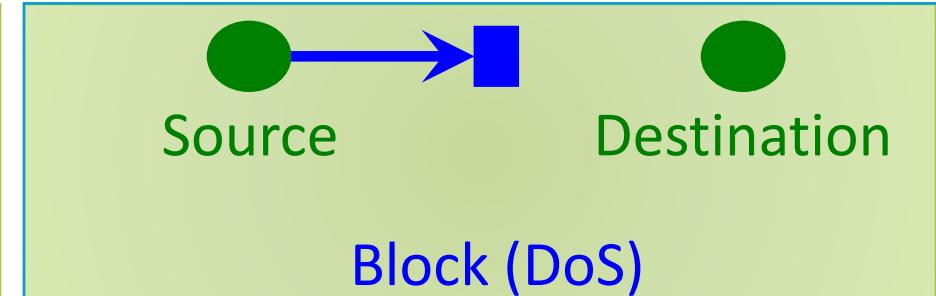
- Internet Control Message Protocol (**ICMP**)
 - Used for network testing and debugging
 - Simple messages encapsulated in single IP packets
 - Considered a network layer protocol
- Tools based on ICMP
 - Ping**: sends series of echo request messages and provides statistics on roundtrip times and packet loss
 - Traceroute**: sends series ICMP packets with increasing TTL value to discover routes



Traceroute



Network Attacks





Wireshark

- Packet sniffer and protocol analyzer
- Captures and displays network packets for analysis
- Supports plugins
- Usually requires administrator privileges because of security risks associated with the program
- When run in promiscuous mode, captures traffic across the network
- Freely available on www.wireshark.org



Network Security: ARP, IP, TCP, UDP

COMPUTER SECURITY
MARKULF KOHLWEISS

During normal operation:

- My laptop always has the same IP address.
 - False
- My laptop always has the same MAC wireless address.
 - True
- VPNs hide my laptops IP from the web site I am visiting.
 - True
- VPNs protect my data from modification between my computer and the destination website.
 - False – VPNs only protect to VPN endpoint
- My ISP (and my VPN) can add and change cookies sent to a website.
 - True – Unless the cookies are encrypted

IP and MAC Addresses

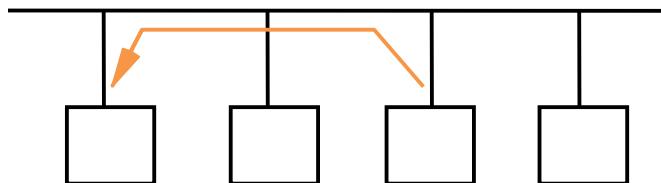
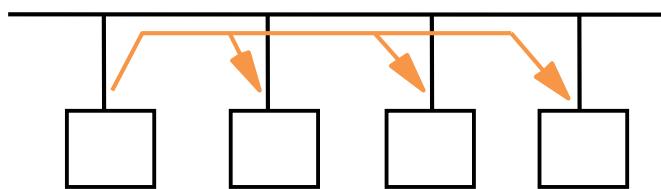
- Devices on a local area network have
 - IP addresses (network layer)
 - MAC addresses (data link layer)
- IP addresses are used for high level protocols
- MAC addresses are used for low level protocols
- How to translate IP Addresses into MAC addresses?

Address Resolution Protocol (ARP)

- Connects the network layer to the data link layer
- Maps IP addresses to MAC addresses
- Based on broadcast messages and local caching
- Does not support confidentiality, integrity, or authentication
- Defined as a part of **RFC 826**
(IETF, Request For Comments)

ARP Messages

- ARP **broadcasts** requests of type
who has <IP addressC>
tell <IP addressA>
- Machine with <IP addressC> responds
<IP addressC> is at <MAC address>
- Requesting machine caches response
- Network administrator configures IP address and subnet on each machine



ARP Cache

- The Linux, Windows and OSX command `arp - a` displays the ARP table

Internet Address	Physical Address	Type
128.148.31.1	00-00-0c-07-ac-00	dynamic
128.148.31.15	00-0c-76-b2-d7-1d	dynamic
128.148.31.71	00-0c-76-b2-d0-d2	dynamic
128.148.31.75	00-0c-76-b2-d7-1d	dynamic
128.148.31.102	00-22-0c-a3-e4-00	dynamic

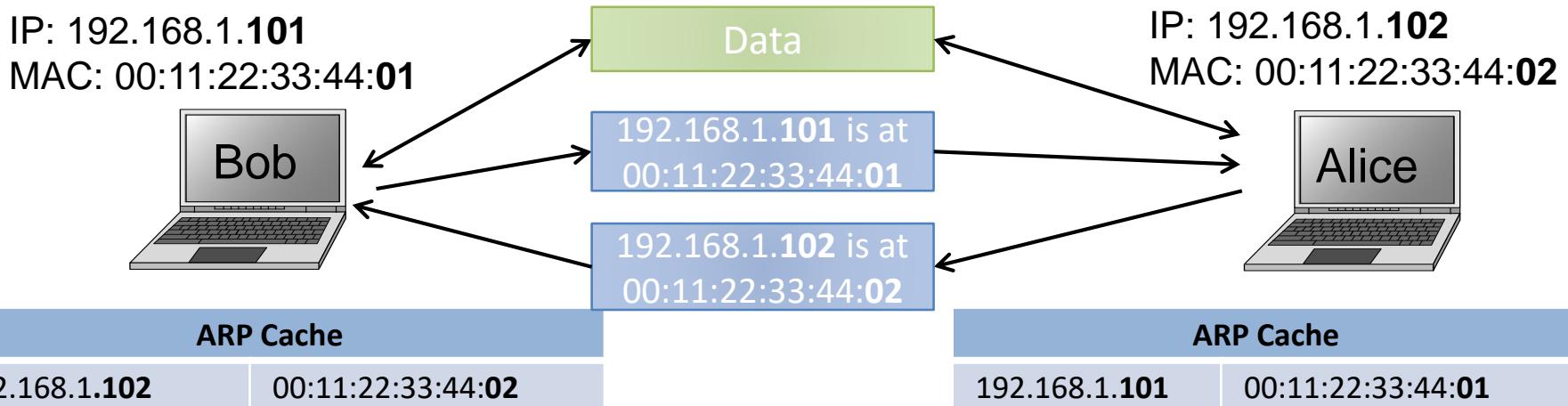
- Command `arp -a -d` flushes the ARP cache (Windows, Apple?)
- ARP cache entries are stored for a configurable amount of time

ARP Spoofing

- The ARP table is updated whenever an ARP response is received
- Requests are not tracked
- ARP announcements are not authenticated
- Machines trust each other
- A rogue machine can spoof other machines

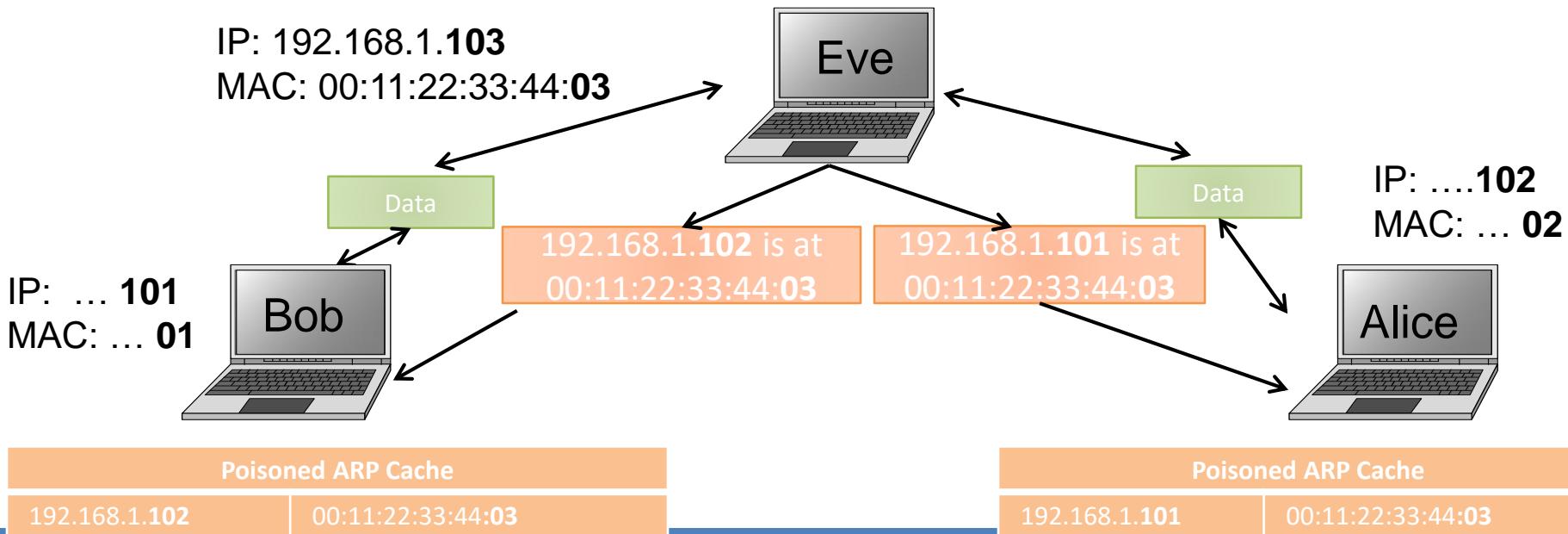
ARP Normal Operation

- Normal operation
 - Alice communicates with Bob



ARP Poisoning Attack

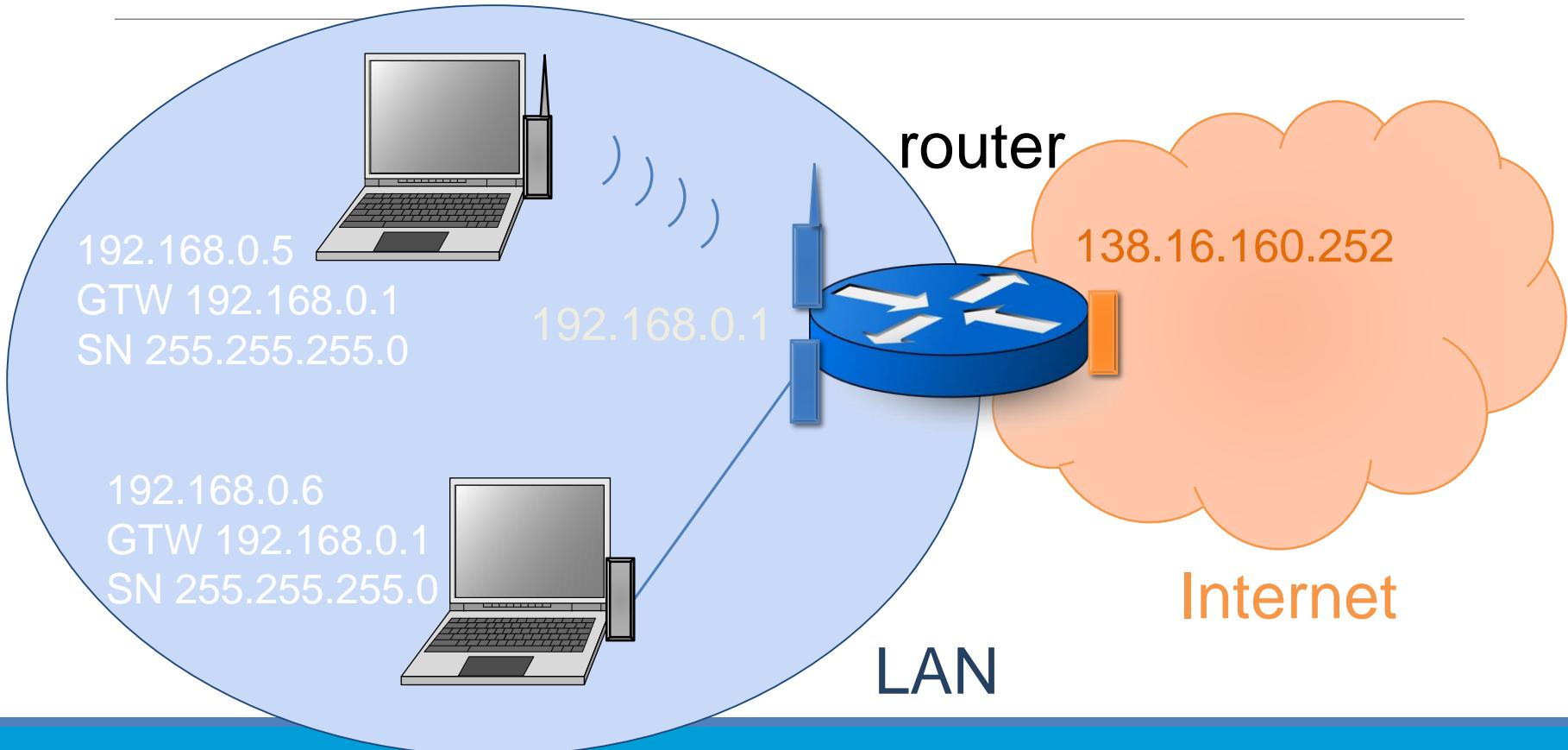
- Man-in-the-middle attack
 - ARP cache poisoning leads to eavesdropping



ARP Poisoning & ARP Spoofing

- Almost all ARP implementations are stateless
- An ARP cache updates every time that it receives an ARP reply
 - ... even if it did not send any ARP request!
- Can “poison” ARP cache with **gratuitous ARP replies**
- Using static entries solves the problem but it is almost impossible to manage!

From the LAN to the Internet



Edinburgh's IP Space

- Edinburgh is part of the autonomous system (AS786) of Jisc Services Limited, for Joint Information Systems Committee
 - Class B network **129.215.0.0**/16 (64K addresses)
- School of Informatics
 - 40 or so sub-networks, class C (/24) with 254 addresses or slightly larger
 - Server machines: 129.215.**33**.0/24
 - DICE desktop machines: 129.215.**24**.0/22
 - Laptops without a fixed IP address: 129.215.**90**.0/23

User Datagram Protocol

- UDP is a stateless, unreliable datagram protocol built on top of IP, that is it lies at the transport layer
- UDP does not provide delivery guarantees or acknowledgments, which makes it efficient
- Can however distinguish data for multiple concurrent applications on a single host
- A lack of reliability implies applications using UDP must be ready to accept a fair amount of corrupted and lost data
 - Most applications built on UDP will suffer if they require reliability
 - VoIP, streaming video, and streaming audio all use UDP

Transmission Control Protocol

- Transport layer protocol for reliable data transfer, in-order delivery of messages and ability to distinguish multiple applications on same host
 - HTTP and SSH are built on top of TCP
- TCP packages a data stream into segments transported by IP
 - Order maintained by marking each packet with **sequence number**
 - Every time TCP receives a packet, it sends out an ACK to indicate successful receipt of the packet
- TCP generally checks data transmitted by comparing a checksum of the data with a checksum encoded in the packet

Ports

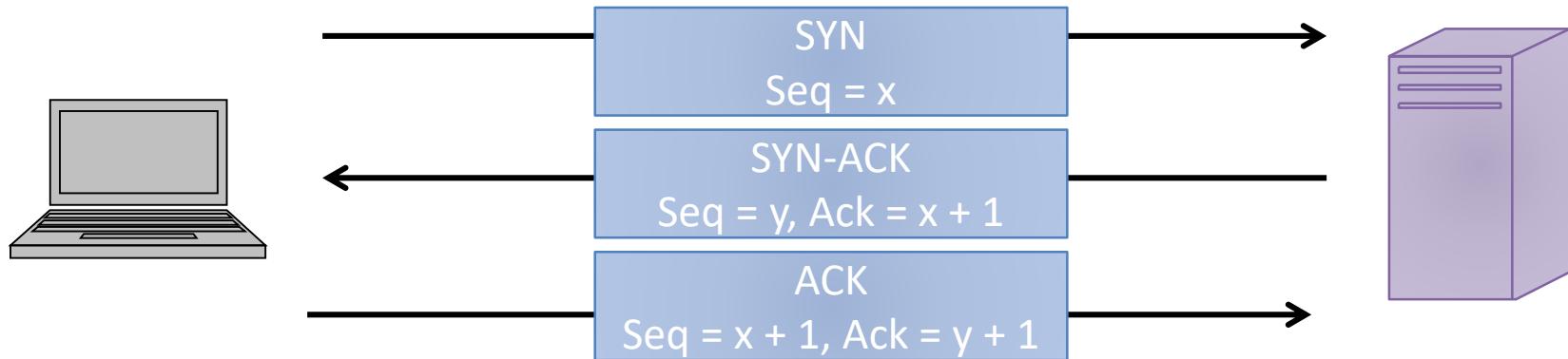
- TCP (& UDP) supports concurrent applications on the same server
- Ports are 16 bit numbers identifying where data is directed
- The TCP header includes both a source and a destination port
- Ports 0 through 1023 are reserved for use by known protocols
 - E.g., HTTPS uses 443 and SSH uses 22
- Ports 1024 through 49151 are known as user ports, and are used for listening to connections

TCP Packet Format

Bit Offset	0-3	4-7	8-15	16-18	19-31		
0	Source Port			Destination Port			
32	Sequence Number						
64	Acknowledgment Number						
96	Offset	Reserved	Flags	Window Size			
128	Checksum			Urgent Pointer			
160	Options						
>= 160	Payload						

Establishing TCP Connections

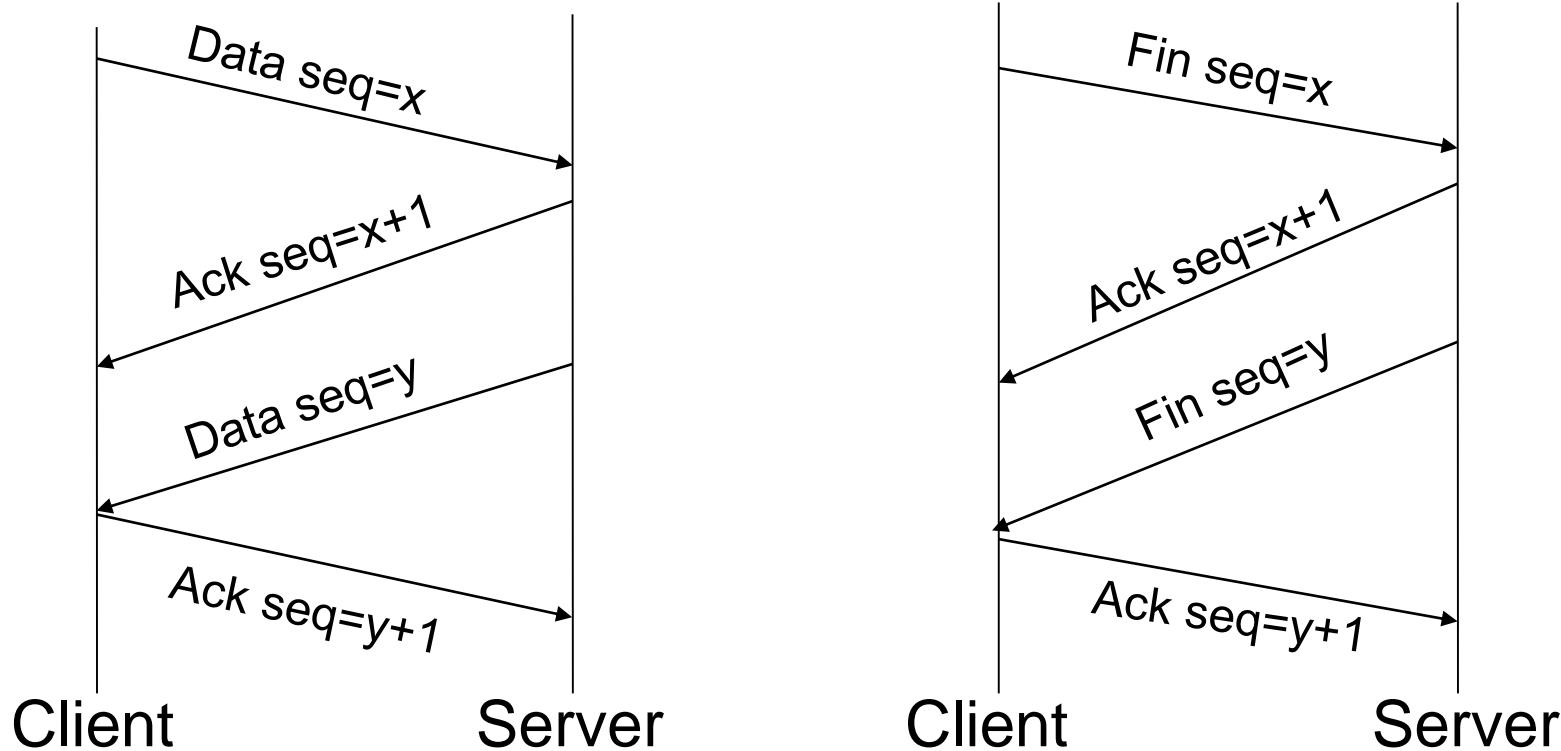
- TCP connections are established through a three-way handshake.
- The server generally is a passive listener, waiting for a connection request
- The client requests a connection by sending out a SYN packet
- The server responds by sending a SYN/ACK packet, acknowledging the connection
- The client responds by sending an ACK to the server, thus establishing connection



TCP Data Transfer

- During connection initialization using the three way handshake, initial sequence numbers are exchanged
- The TCP header includes a 16 bit checksum of the data and parts of the header, including the source and destination
- Acknowledgment or lack thereof is used by TCP to keep track of network congestion and control flow and such
- TCP connections are cleanly terminated with a 4-way handshake
 - The client which wishes to terminate the connection sends a FIN message to the other client
 - The other client responds by sending an ACK
 - The other client sends a FIN
 - The original client now sends an ACK, and the connection is terminated

TCP Data Transfer and Teardown



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.

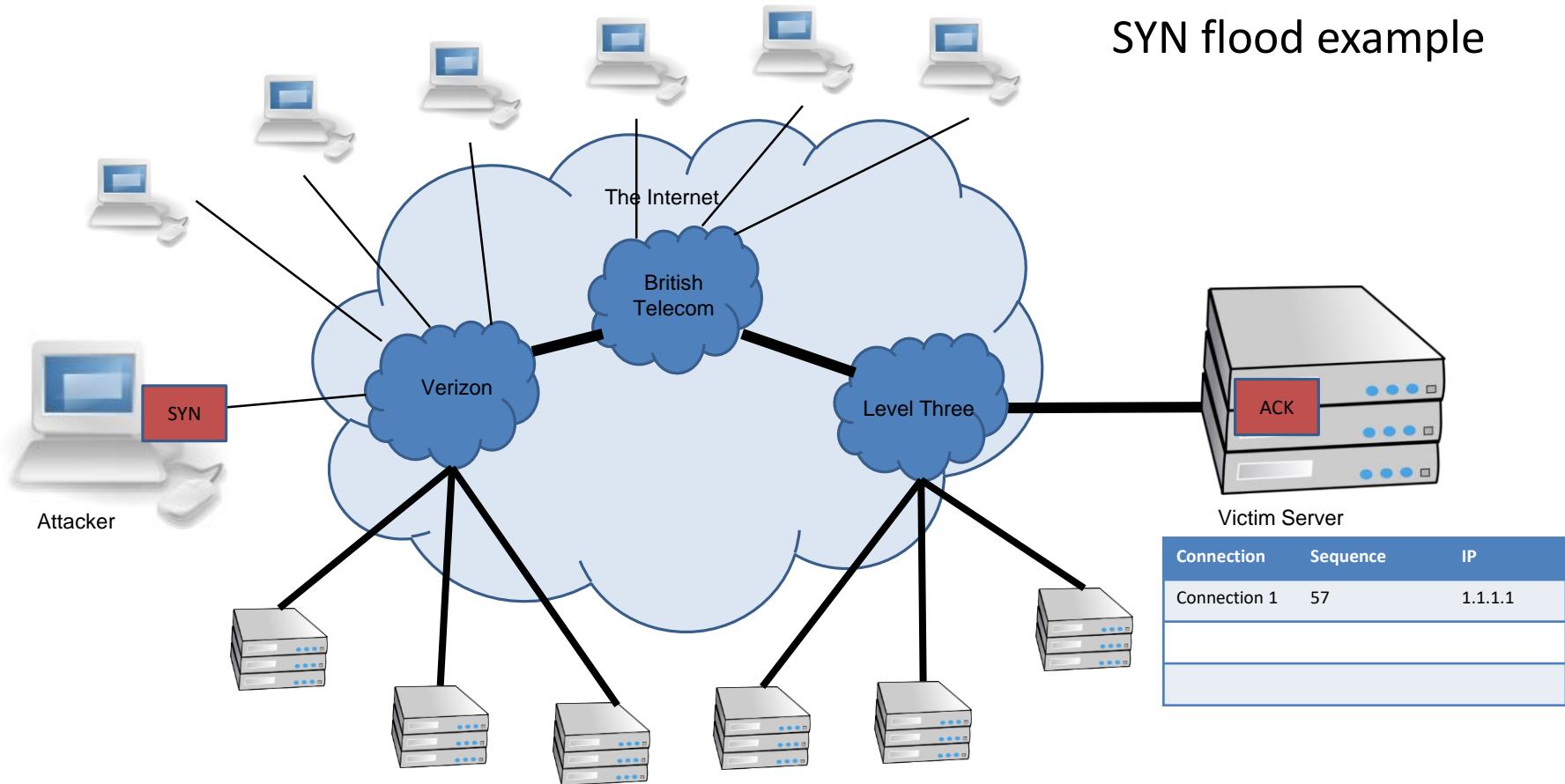
$A \rightarrow B : \text{SYN}, X$

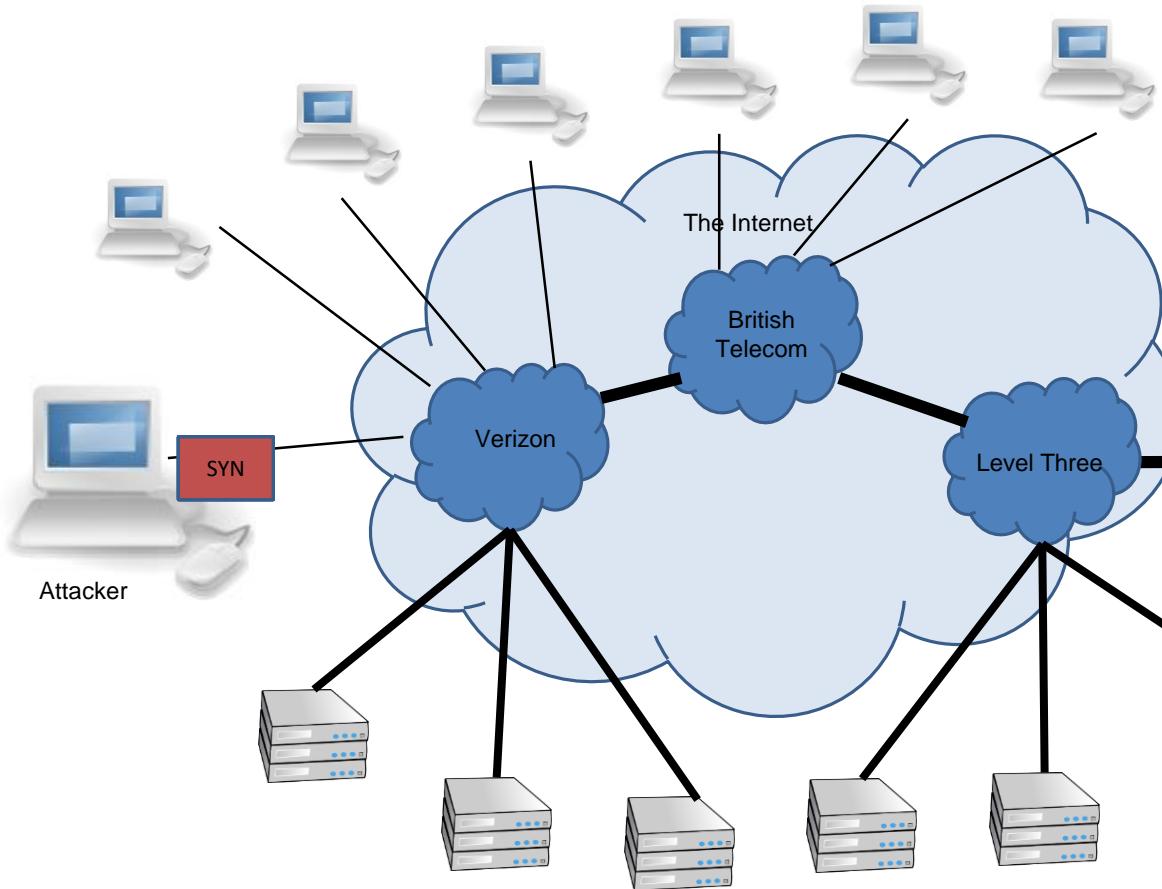
$B \rightarrow A : \text{ACK}, X + 1; \text{SYN}, Y$

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

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

SYN flood example





SYN flood example

- Attacker sends SYN and ignores ACK
- Victim must maintain state



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

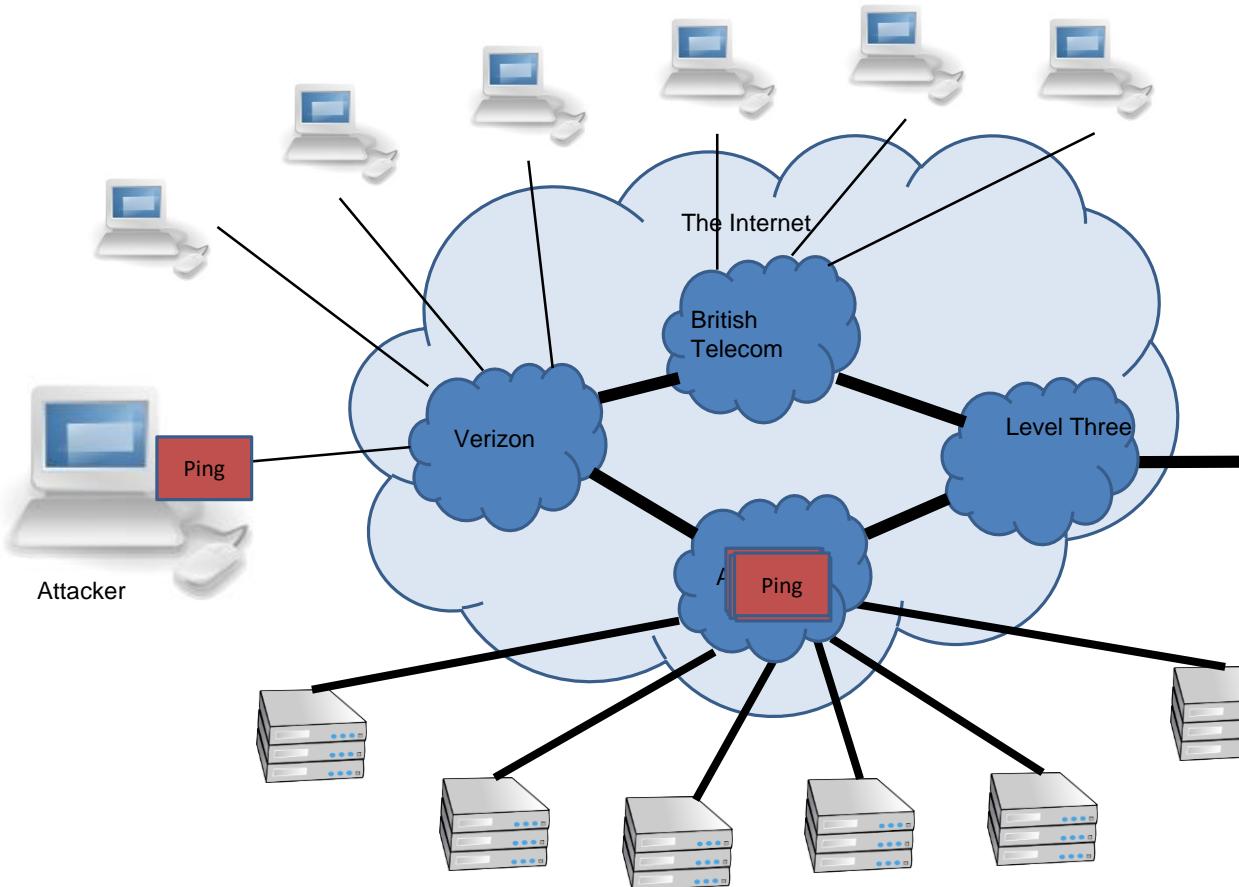
- Problems
 - Attribution – attacker uses their own IP which could be traced
 - Bandwidth – attacker uses their own bandwidth which is likely smaller than a server's
- Effective against a small target
 - Someone running a game server in their home
- Not effective against a large target
 - Company website

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.

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 swamps the target with replies.



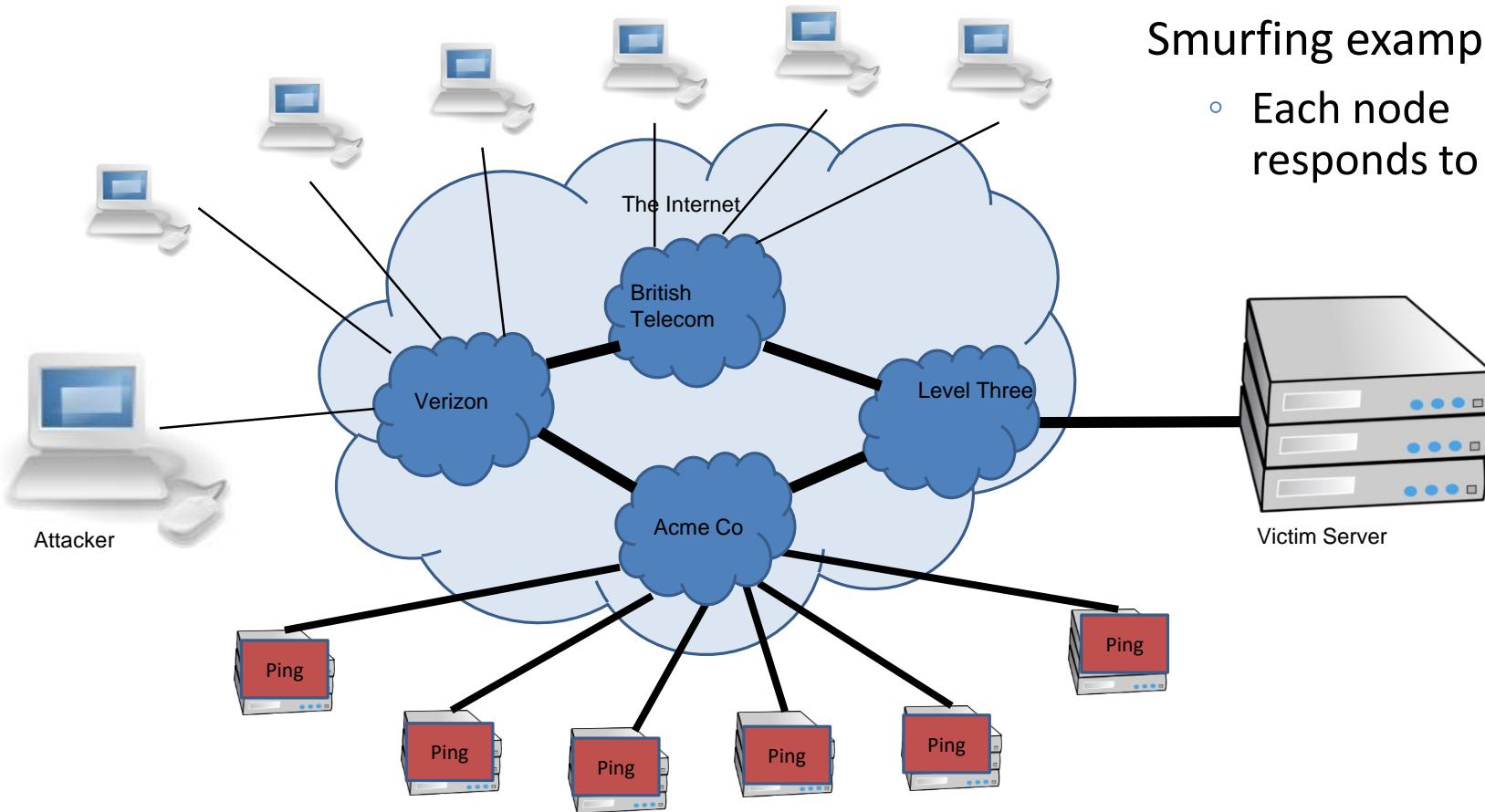
Smurfing example

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



Smurfing example

- Each node responds to victim



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

What We Have Learned

- ARP protocol
- ARP poisoning attack
 - MitM attack on a LAN
- Transport layer protocols
 - TCP for reliable transmission
 - UDP when packet loss/corruption is tolerated
- Lack of built-in security in network protocols
 - Security can be incorporated into application layer (SSL)

Network Security: Application-Layer and Domain Name System

COMPUTER SECURITY
MARKULF KOHLWEISS

Some slides adapted from those by Myrto Arapinis, Kami Vaniea, and Roberto Tamassia

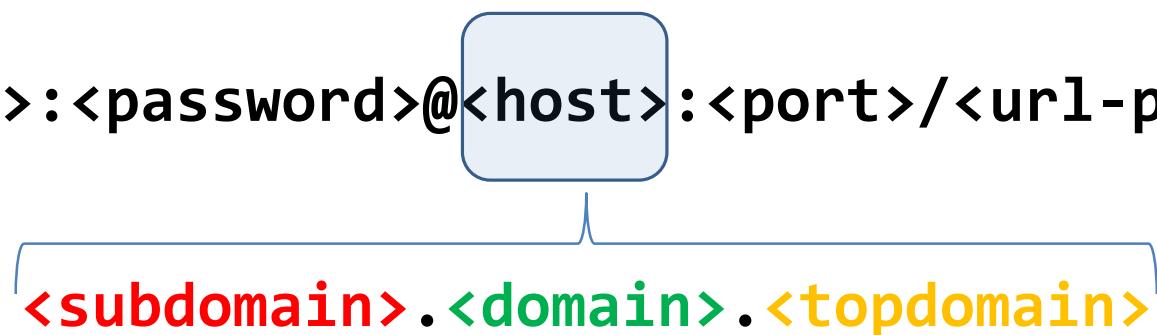
Sample Application-Layer Protocols

- Domain name system (**DNS**)
- Hypertext transfer protocol (**HTTP**)
- **SSL/TLS**. Protocol used for secure, encrypted browsing (**HTTPS**)
- **IMAP/POP/SMTP**. Internet email protocols
- File transfer protocol (**FTP**). An old but still used protocol for uploading and downloading files
- **Telnet**. Early remote access protocol
- **SSH**. More recent secure remote access protocol.

What is a URL?

- Uniform Resource Locators (URLs) are a standardized format for describing the location and access method of resources via the internet.

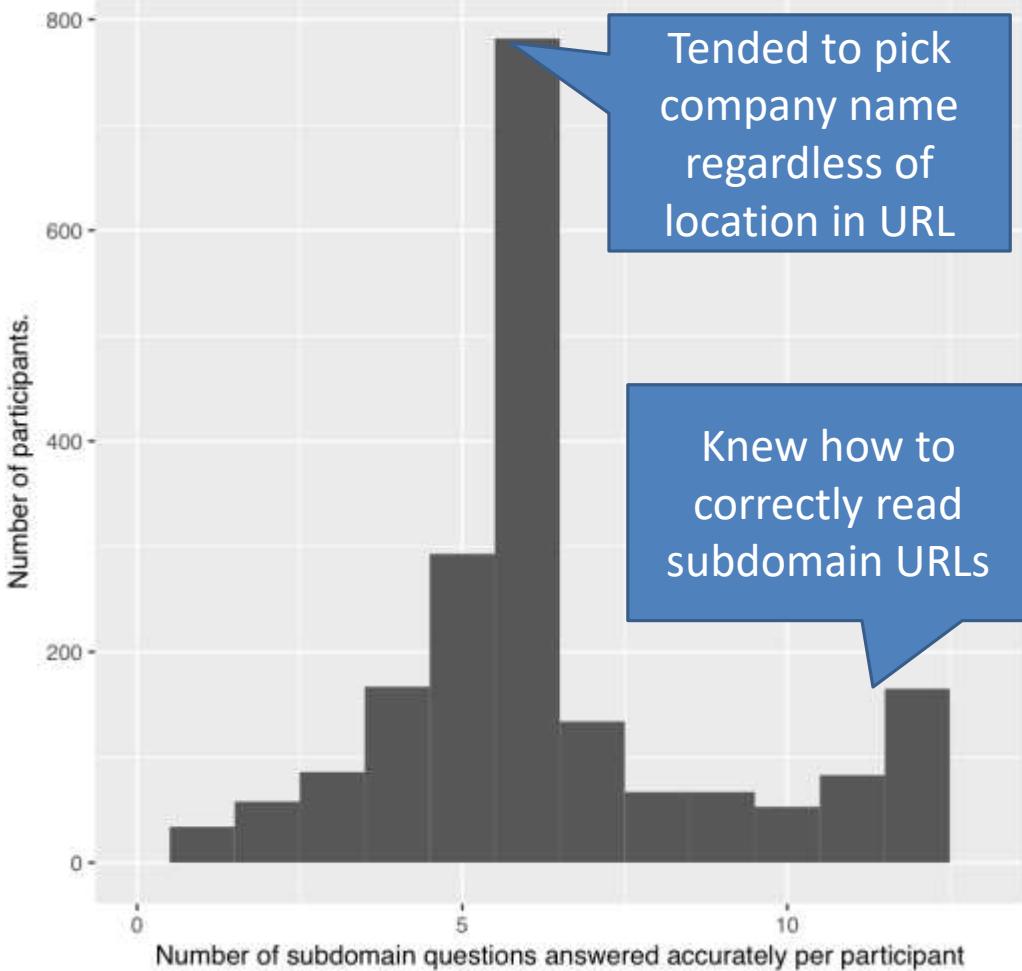
<scheme>://<user>:<password>@<host>:<port>/<url-path>?<query-string>



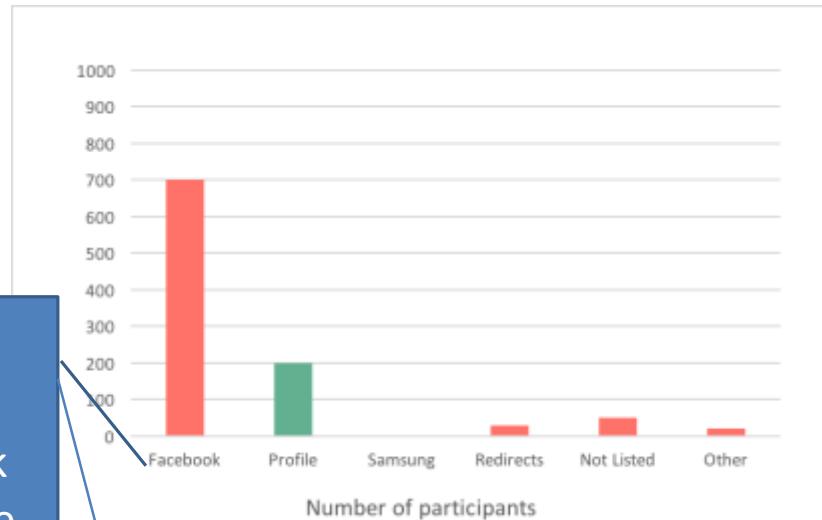
eg. https://profile.facebook.com

<https://facebook.profile.com>

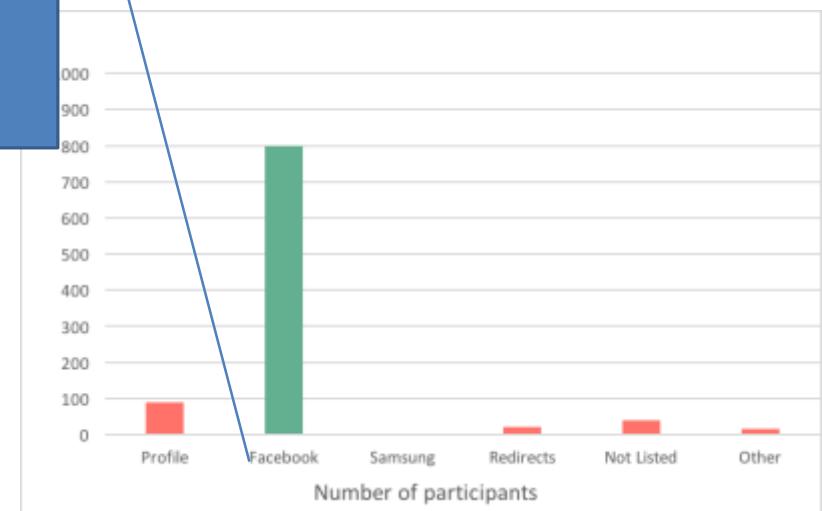
Total accuracy on subdomain questions



Tended to pick company name regardless of location in URL



<https://profile.facebook.com>



Domains

Domain name

- Two or more labels, separated by dots (e.g., [inf.ed.ac.uk](#))

Top-level domain (TLD)

- Generic (gTLD), e.g., [.com](#), [.org](#), [.net](#)
- Country-code (ccTLD), e.g., [.ca](#), [.it](#)
- New top level domains, e.g., [.scot](#), [.tirol](#)

ICANN

- (non-profit) Internet Corporation for Assigned Names and Numbers
- Keeps database of registered gTLDs ([InterNIC](#))
- Accredits registrars for gTLDs

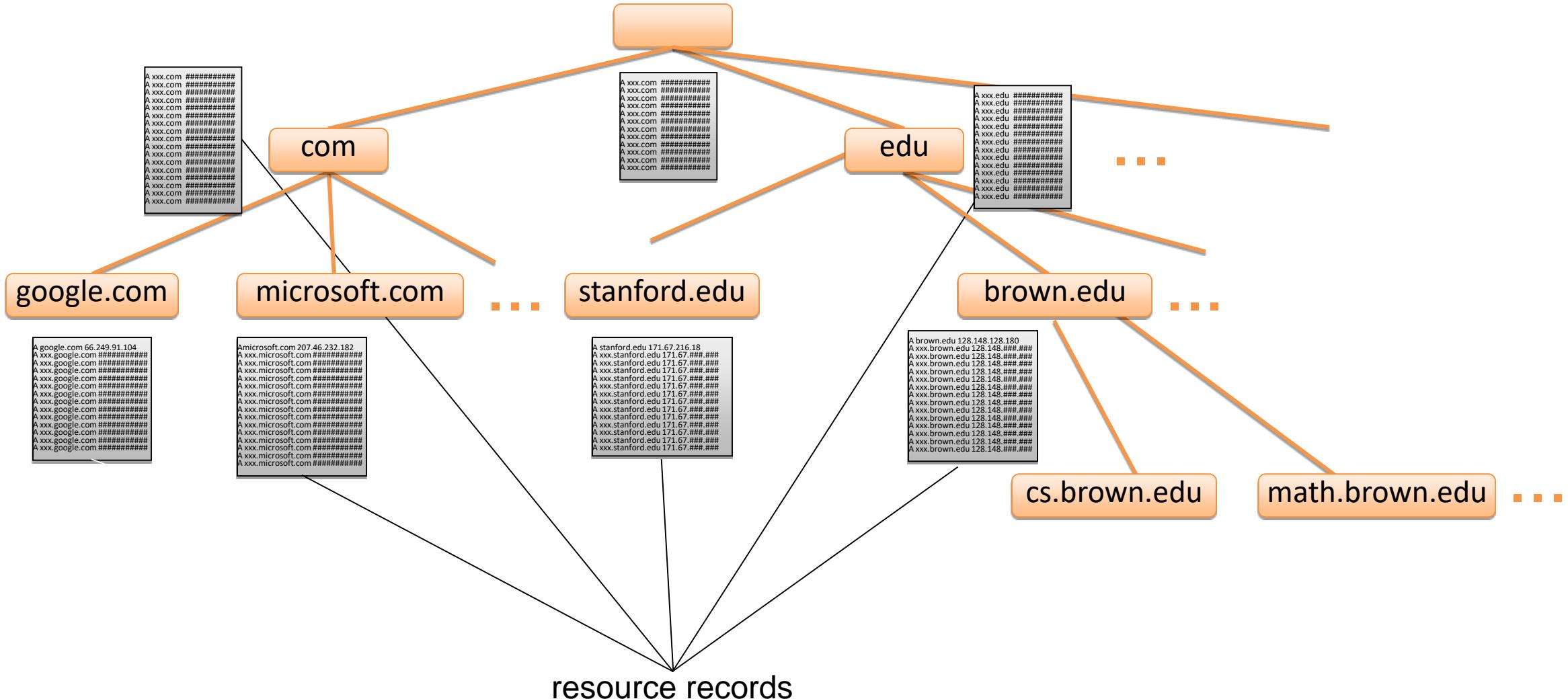
gTLDs

- Managed by ICANN

ccTLDs

- Managed by government organizations

DNS Tree



Name Servers

- Name server
 - Keeps local database of DNS records
 - Answers DNS queries
 - Can ask other name servers if record not in local database
- Authoritative name server
 - Stores reference version of DNS records for a zone (partial tree)
- Examples
 - `dns0.ed.ac.uk` is authoritative for `ed.ac.uk` and `dns0.inf.ed.ac.uk` for `inf.ed.ac.uk`
- Root servers
 - Authoritative for the root zone (TLDs)
 - `[a-m].root-servers.net`
 - Supervised by ICANN

Name Resolution

- Resolver
 - Program that retrieves DNS records
 - E.g., `dig` in Linux and `nslookup` in Windows
 - Caches records received
 - Connects to a name server (default, root, or given)

Iterative resolution

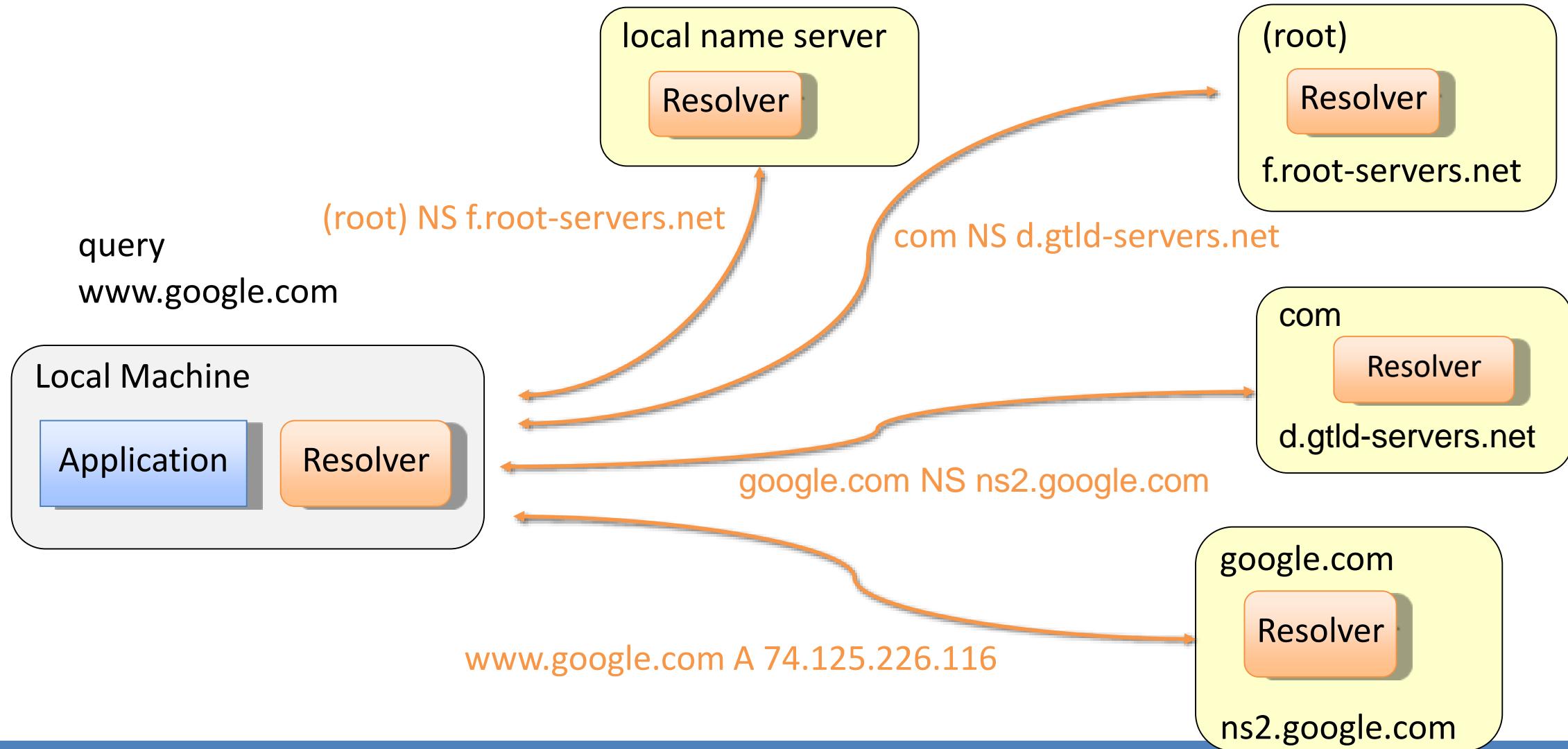
Name server refers client to authoritative server (e.g., a TLD server) via an NS record

Repeat

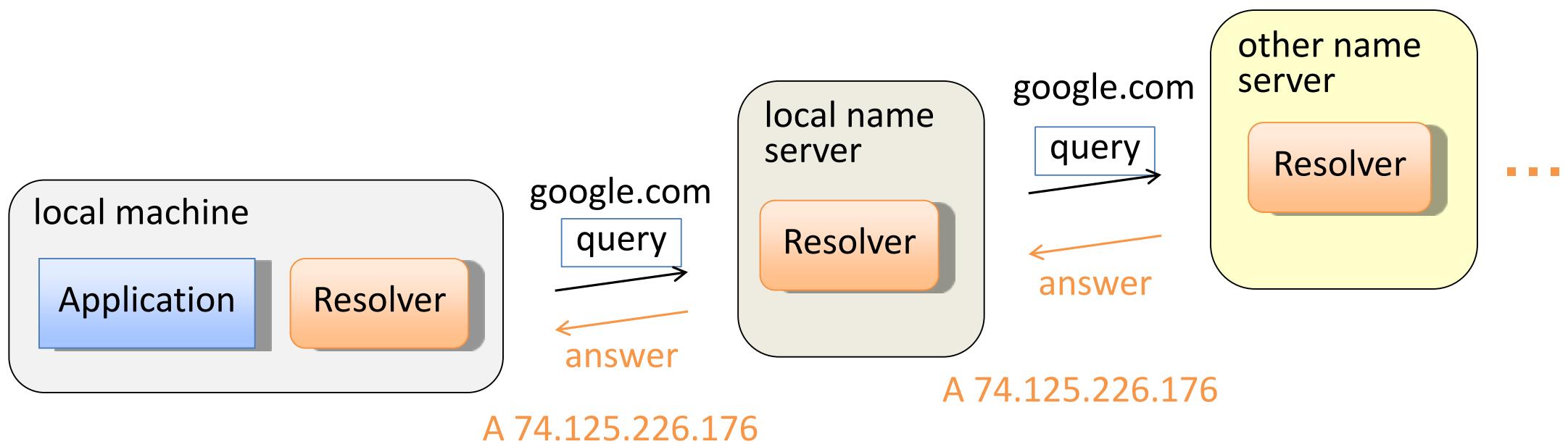
Recursive resolution

Name server queries another server and forwards the final answer (e.g., A record) to client

Iterative Name Resolution



Recursive Name Resolution



Glue Records

Circular references

The authoritative name server for a domain may be within the same domain

E.g., **dns0.inf.ed.ac.uk** is authoritative for **inf.ed.ac.uk**

Glue record

Record of type A (IP address) for a name server referred to NS record
Essential to break circular references

Example

inf.ed.ac.uk.	NS	dns0.inf.ed.ac.uk.
dns0.inf.ed.ac.uk.	A	129.215.160.240 [glue record]

DNS Caching

There would be too much network traffic if a path in the DNS tree would be traversed for each query

Root servers and TLD servers would be rapidly overloaded

DNS servers **cache** records that are results of queries for a specified amount of time

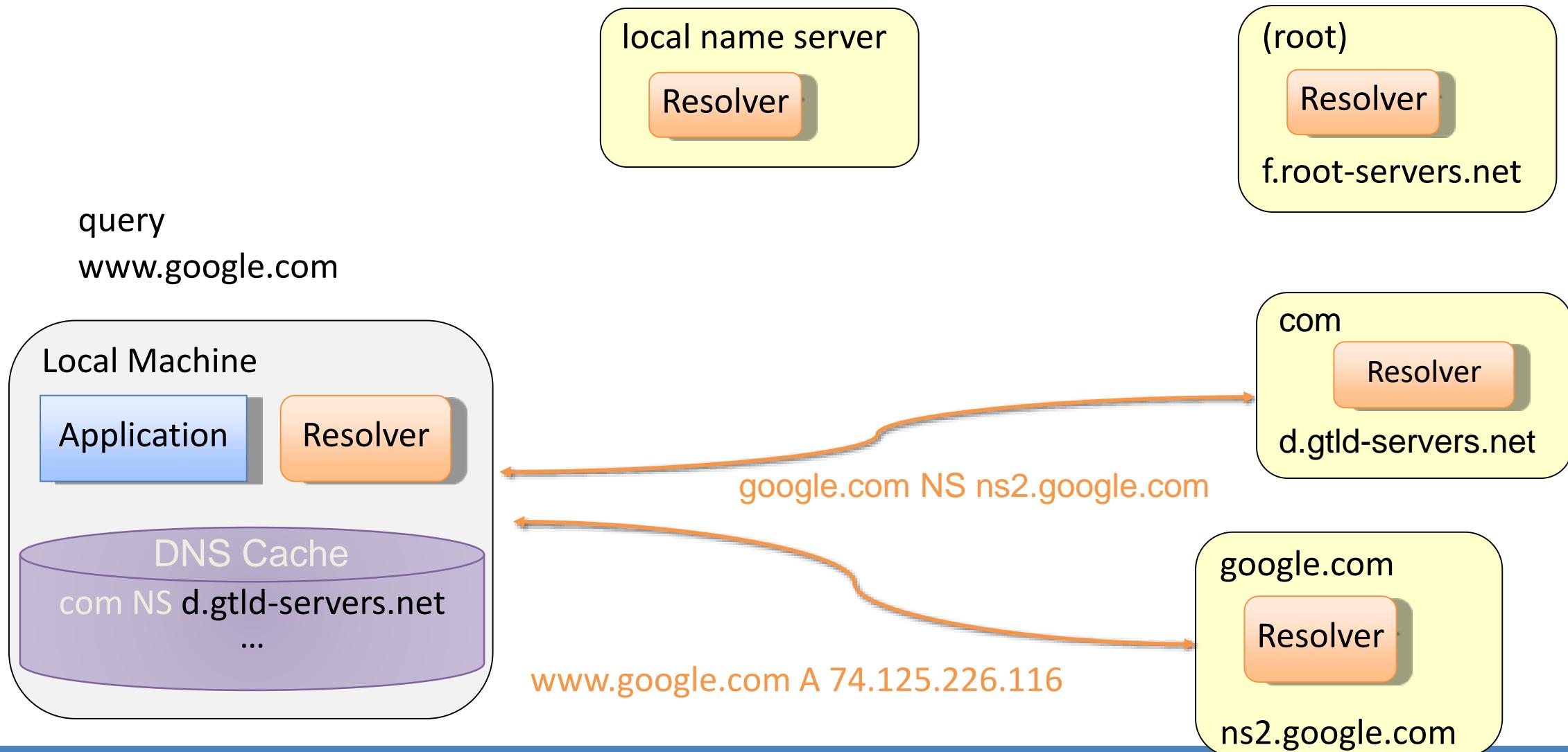
Time-to-live field

DNS queries with caching

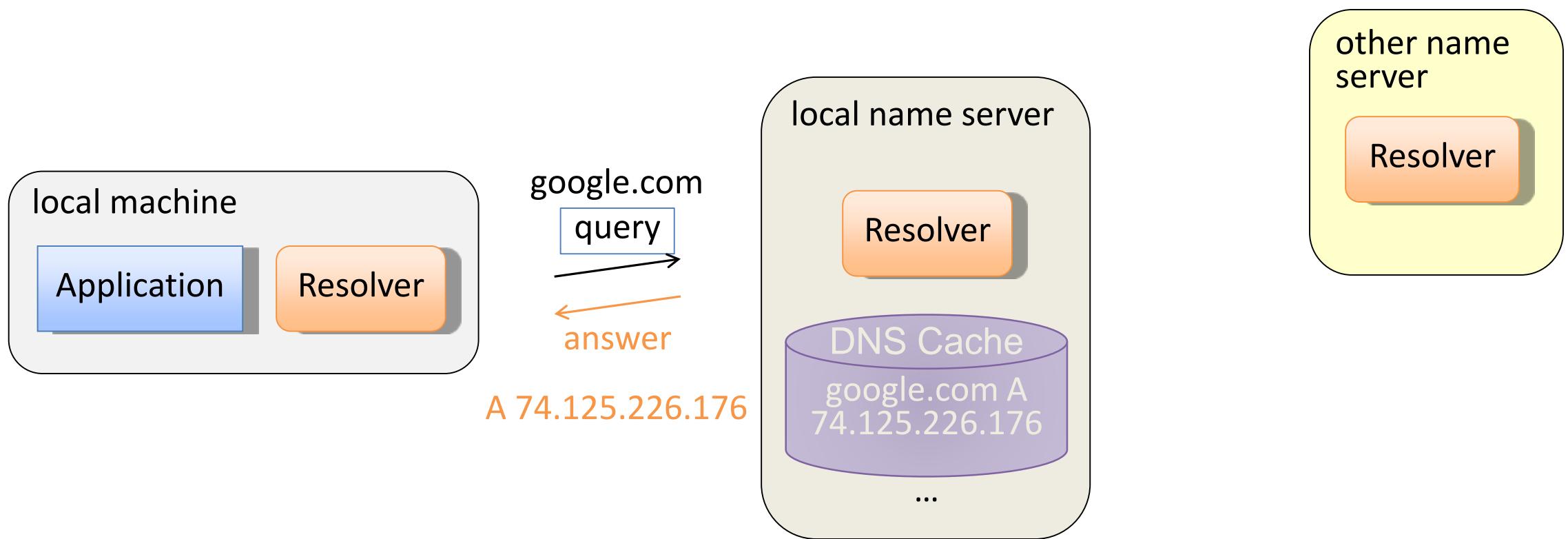
First, resolver looks in cache for A record of query domain

Next , resolver looks in cache for NS record of longest suffix of query domain

Iterative Name Resolution with Caching



Recursive Name Resolution with Caching



Local DNS Cache

Operating system maintains DNS cache

Shared among all running applications

Can be displayed to all users

View DNS cache in Windows with command `ipconfig /displaydns`

Clear DNS cache in Windows with command `ipconfig /flushdns`

Privacy issues

Browsing by other users can be monitored

Note that private/incognito browsing does not clear DNS cache

```
C:\Users\marku>ipconfig /displaydns
```

```
Windows IP Configuration
```

```
arstechnica.com
```

```
-----  
Record Name . . . . . : arstechnica.com  
Record Type . . . . . : 1  
Time To Live . . . . . : 128  
Data Length . . . . . : 4  
Section . . . . . . . : Answer  
A (Host) Record . . . : 50.31.169.131
```

DNS Cache Poisoning

Basic idea

Give a DNS server a false address record and get it cached

DNS query mechanism

Queries issued over UDP on port 53

16-bit **request identifier** in payload to match answers with queries

No authentication

Cache may be poisoned when a resolver

Disregards identifiers

Has predictable identifiers and return ports

Accepts unsolicited DNS records

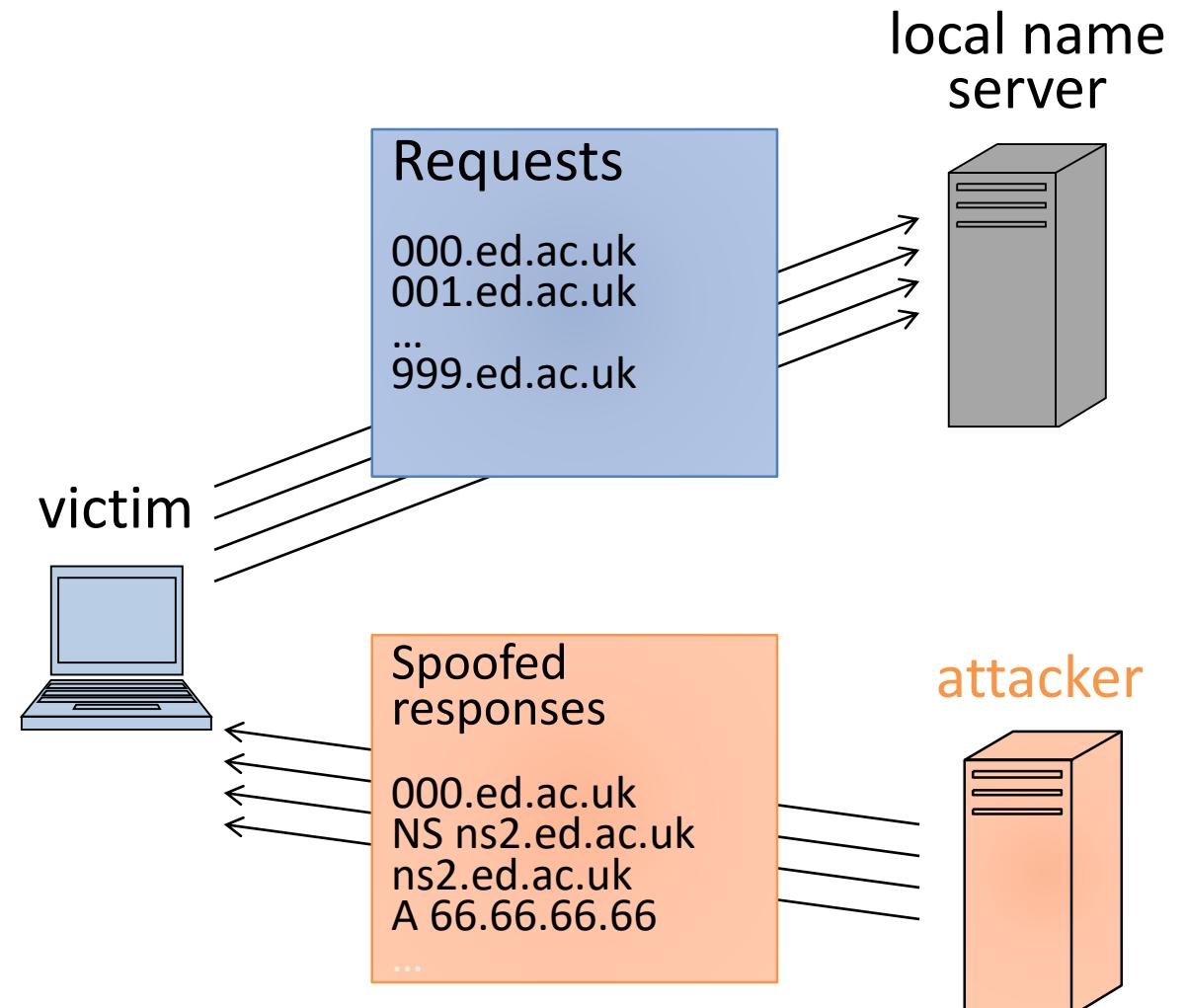
Early versions of BIND (popular DNS software) vulnerable to cache poisoning

DNS Cache Poisoning Defenses

- Query randomization
 - Random request identifier (16 bits)
 - Random return port (16 bits)
- Probability of guessing request ID
or return port
 - $1 / 2^{16} = 0.0015\%$
- Probability of guessing request ID
and return port is
 - $1 / 2^{32}$ (less than one in four billion)
- Check request identifier
- Use signed records
- DNSSEC

Subdomain DNS Cache Poisoning (Kaminsky)

- Attacker causes victim to send
 - Many DNS requests for nonexistent subdomains of target domain
- Attacker sends victim
 - Forged NS responses for the requests
- Format of forged response
 - Random ID
 - Correct NS record
 - Spoofed glue record pointing to the attacker's name server IP



Steve Friedl's Unixwiz.net Tech Tips

An Illustrated Guide to the Kaminsky DNS Vulnerability

The big security news of Summer 2008 has been [Dan Kaminsky's](#) discovery of a [serious vulnerability in DNS](#). This vulnerability could allow an attacker to redirect network clients to alternate servers of his own choosing, presumably for ill ends.

Table of Contents

- [Terminology](#)
- [Following a simple DNS query](#)
- [What's in a DNS packet?](#)
- [Resource Record Types](#)
- [Drilling down to a real query](#)
- [What's in the cache?](#)
- [Poisoning the cache](#)
- [Shenanigans, Version 1](#)
- [Dan's Shenanigans](#)
- [What's the fix?](#)
- [Summary](#)
- [Other References](#)

This all led to a mad dash to patch DNS servers worldwide, and though there have been many writeups of just how the vulnerability manifests itself, we felt the need for one in far more detail. Hence, one of our Illustrated Guides.

This paper covers how DNS works: first at a high level, then by picking apart an individual packet exchange field by field. Next, we'll use this knowledge to see how weaknesses in common implementations can lead to cache poisoning.

By fully understanding the issues at play, the reader may be better equipped to mitigate the risks in his or her own environment.

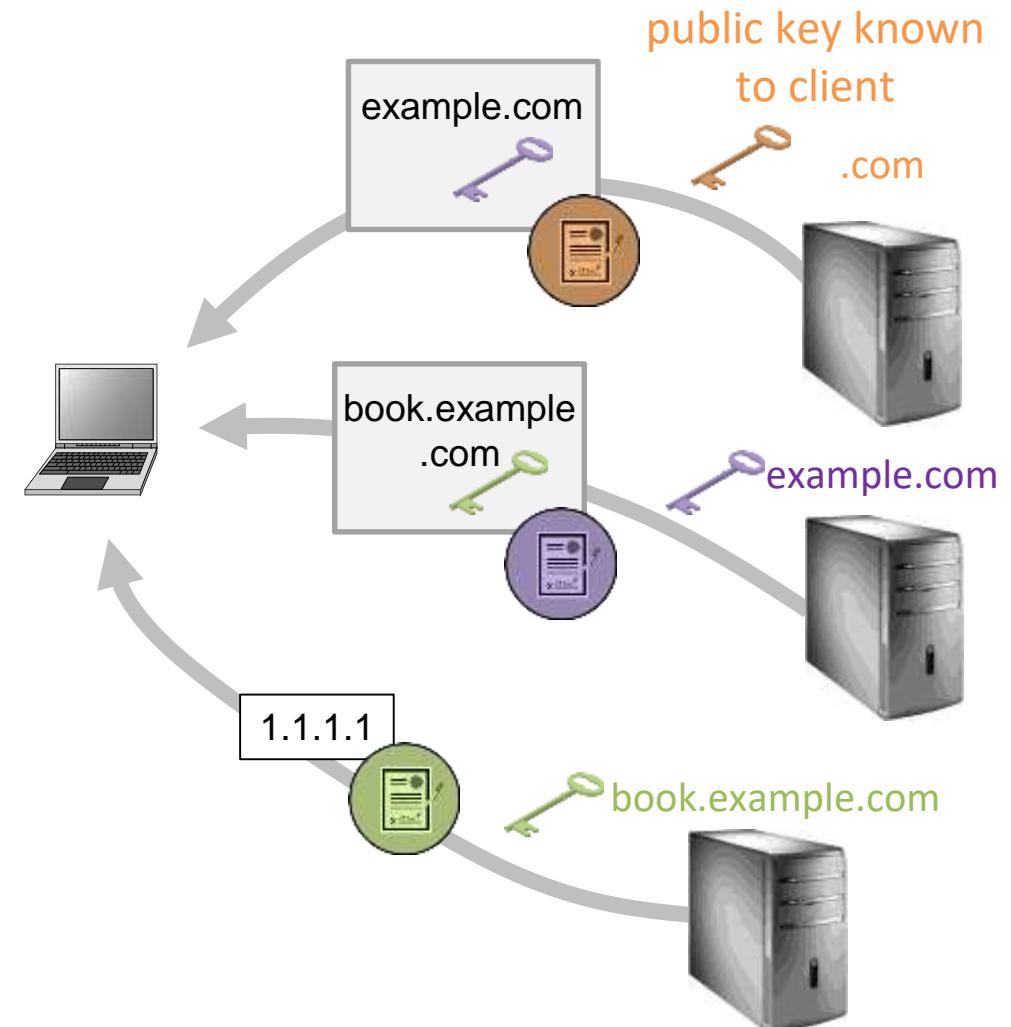
We hope everybody who runs a DNS server patches soon.



Nice work, Dan

DNSSEC

- Goals
 - Authenticity of DNS answer origin
 - Integrity of reply
 - Authenticity of denial of existence
- Implementation
 - Signed DNS replies at each step
 - Public-key cryptography
 - Certificates in the OS
- Slow deployment
 - Root servers support since 2010



What We Have Learned

- How DNS operates
 - Distributed database
 - Resolvers and name servers
 - Iterative vs. recursive resolution
 - Caching
- DNS cache poisoning attacks
- DNSSEC

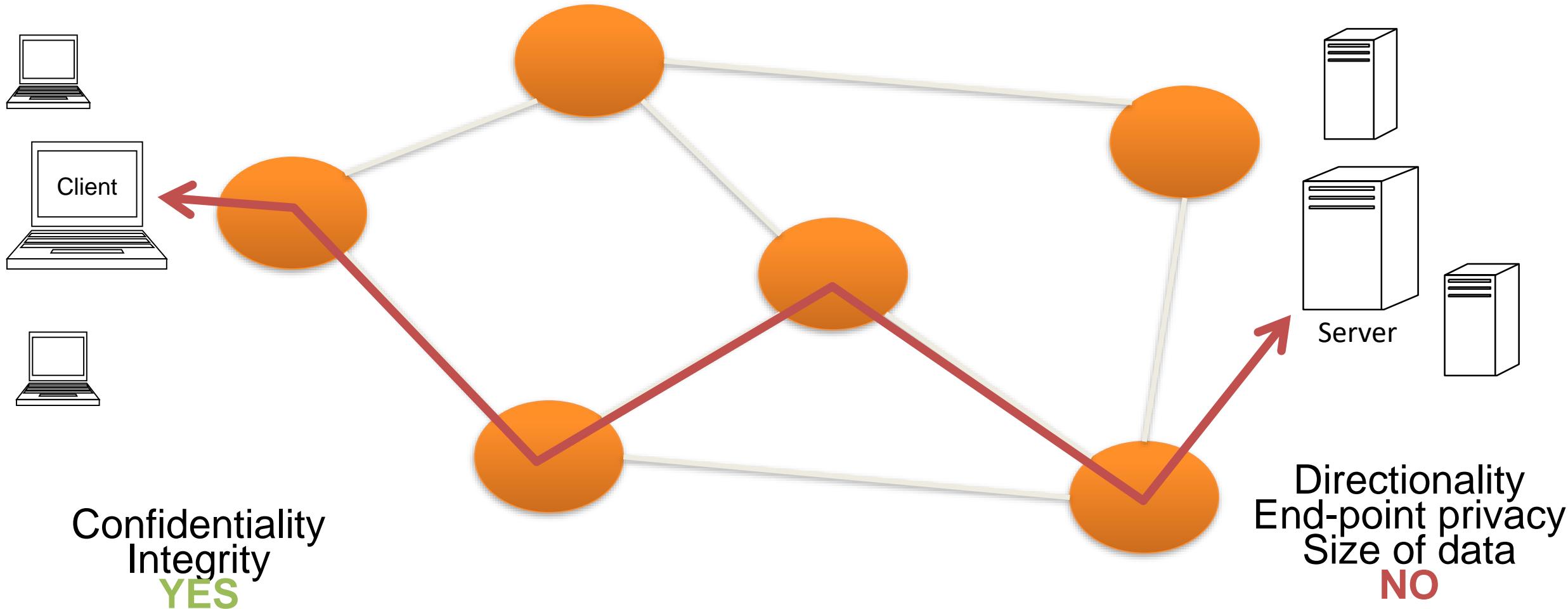
SSL/TLS

COMPUTER SECURITY
MARKULF KOHLWEISS

Some slides adapted from those by Myrto Arapinis, Kami Vaniea, Aggelos Kiayias, and Roberto Tamassia

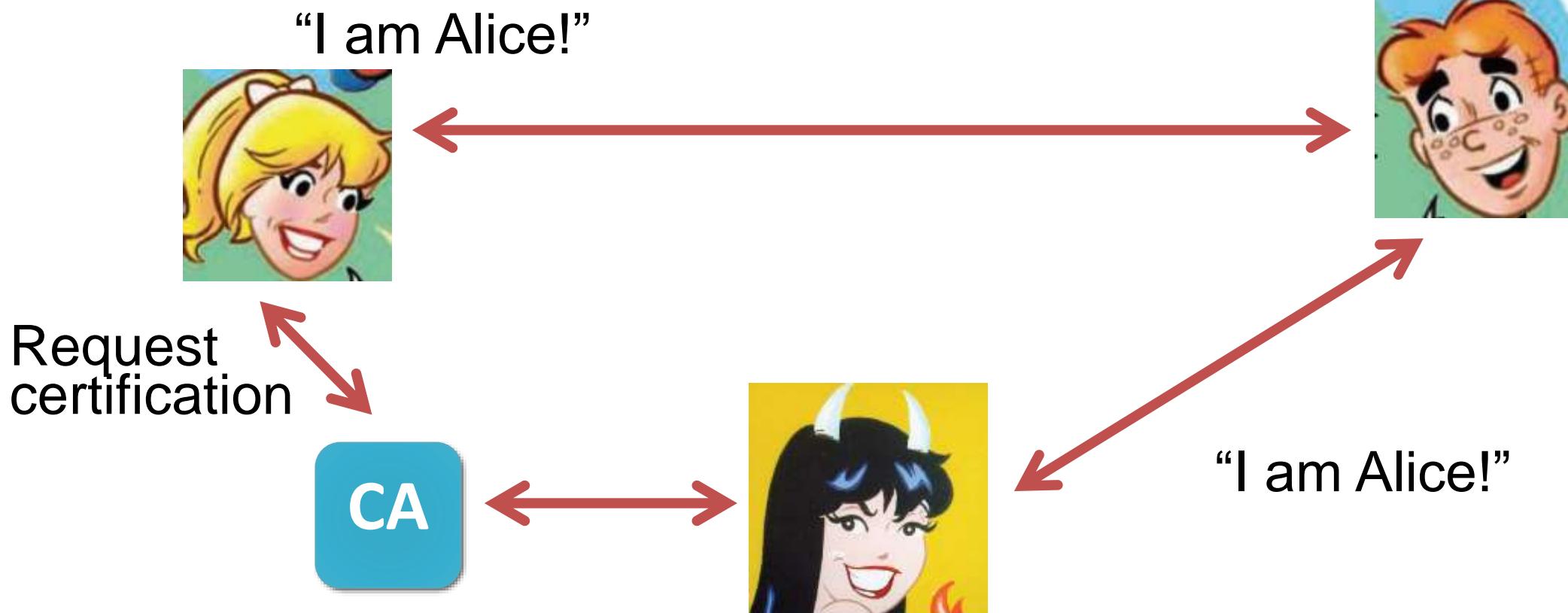


Objective: point-to-point secure channel



Identification Problem

?



SSL/TLS

- Secure Sockets Layer: Developed by Netscape.
- SSL Version 3 released in 1996.
- Substituted by TLS 1.0 in 1999: (Transport Layer Security). Standardized by IETF. Published as RFC 2246
- TLS 1.3. Published as RFC 8445 in 2018.
- On top of TCP/IP, below application protocols.



Taher Elgamal

Source [Alexander Klink](#) via [Wikipedia](#)



An old story ...

... with new twists

1976-1978

- New Directions in Cryptography
- Data Encryption Standard (DES)
- RSA

1994

- Netscape SSL
- SSL2

1995

- SSL3
- Predictable IV (Rogaway)

1996

- MD5 (Dobbertin)

1998

- PKCS1 (Bleichenbacher)

1999

- TLS 1.0

2000

- C

Comments
[draft-ronnaway-insec-comments-a00.txt](#)

P. Rogaway
UC Davis

Security Flaws Induced by CBC Padding

Lucky Thirteen: Breaking the TLS and DTLS Record Protocols

On the Security of RC4 in TLS¹

Nadhem J. AlFardan
*Information Security Group,
University of Illinois at Chicago and
Technische Universität Eindhoven*

Daniel J. Bernstein
*University of Illinois at Chicago and
Technische Universität Eindhoven*

Kenneth G. Paterson
*Information Security Group,
University of London*



Verifying TLS implementations

1999 2002-2003 2006 2008 2009 2011 2012-2013

- TLS 1.0
- **CBC Padding (Vaudenay)**
- TLS 1.1
- TLS 1.2
- **Symbolic Model**
- **Renegotiation**
- **BEAST**
- **CRIME**
- **Lucky 13**
- **RC4**
- **Most Dangerous Code**
- **Implementation Attacks**
- **Cross Protocol Attacks**

Renegotiating TLS

A Cross-Protocol Attack on the TLS Protocol

Marsh Ray
Steve Dispensa
PhoneFactor, Inc.

v1.1 November 4, 2009

Summary

Transport Layer Security (TLS) is subject to a number of serious vulnerabilities. One of these is renegotiation. In general, the chosen plaintext into the beginning of the renegotiation message can be controlled by the attacker.

Nikos Mavrogiannopoulos
KU Leuven
ESAT/SCD/COSIC – IBBT
Leuven, Belgium
nikos@esat.kuleuven.be

Frederik Vercauteren
KU Leuven
ESAT/SCD/COSIC – IBBT
Leuven, Belgium
fvercaut@esat.kuleuven.be

Vesselin Velichkov
University of Luxembourg
Luxembourg
vesselin.velichkov@uni.lu

Bart Preneel
KU Leuven
ESAT/SCD/COSIC – IBBT
Leuven, Belgium
preneel@esat.kuleuven.be



Verifying TLS implementations

1999	2002-2003	2006	2008	2009	2011	2012-2013	2013
• TLS 1.0	• CBC Padding (Vaudenay)	• TLS 1.1	• TLS 1.2 • Symbolic Model	• Renegotiation	• BEAST	• CRIME • Lucky 13 • RC4 • Most Dangerous Code • Implementation Attacks • Cross Protocol Attacks	• Implementing Verified Cryptographic Security Bhargavan, Fournet, Kohlweiss, Pöhlmann • Snowden



Analyses the *entirety* of TLS using machine-assisted proof techniques.

- **ambitious but only real way.**
- **must not simplify** the underlying cryptography.



Toward TLS 1.3

2008	2009	2011	2012-2013	2013	2014
<ul style="list-style-type: none">• TLS 1.2• Symbolic Model	<ul style="list-style-type: none">• Renegotiation	<ul style="list-style-type: none">• BEAST	<ul style="list-style-type: none">• CRIME• Lucky 13• RC4• Most Dangerous Code• Implementation Attacks• Cross Protocol Attacks	<ul style="list-style-type: none">• Implementing TLS with Verified Cryptographic Security Bhargavan, Fournet, Kohlweiss, Pironti, Strub• Snowden	<ul style="list-style-type: none">• Proving the TLS Handshake Secure (as it is) Bhargavan, Fournet, Kohlweiss, Pironti, Strub, Zanella-Beguelin• Triple Handshake attack• New Bleichenbacher attacks

Internet Engineering Task Force (IETF)
Request for Comments: 8446
Obsoletes: 5077, 5246, 6961
Updates: 5705, 6066
Category: Standards Track
ISSN: 2070-1721

E. Rescorla
Mozilla
August 2018

The Transport Layer Security (TLS) Protocol Version 1.3

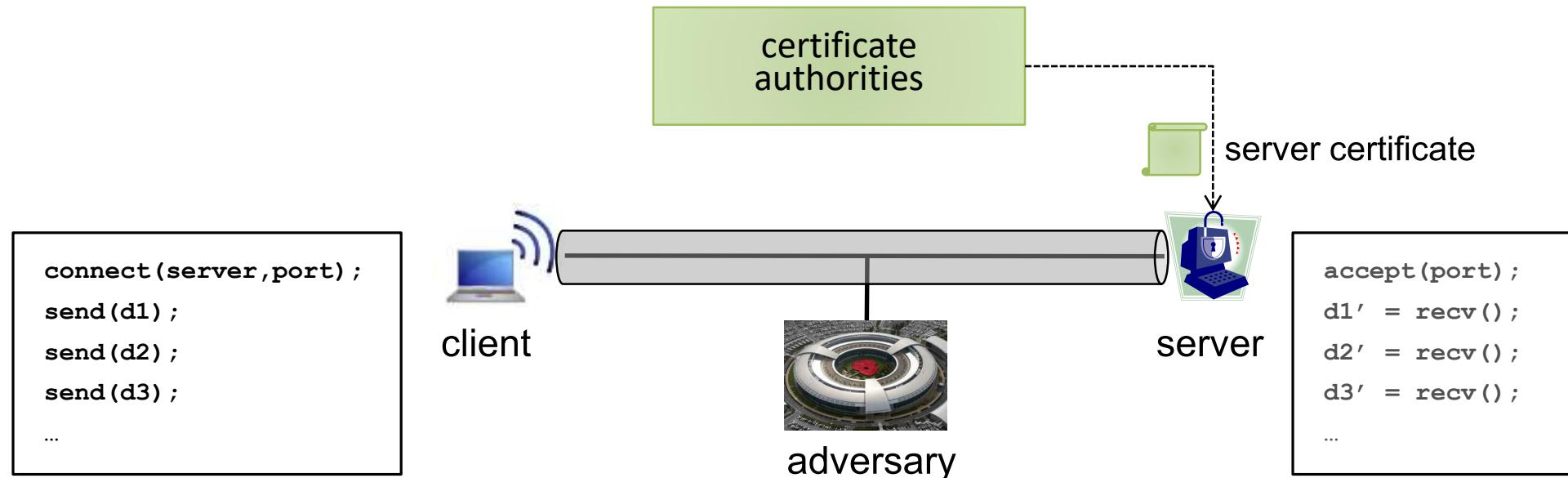
Abstract

This document specifies version 1.3 of the Transport Layer Security (TLS) protocol. TLS allows client/server applications to communicate over the Internet in a way that is designed to prevent eavesdropping, tampering, and message forgery.

This document updates RFCs 5705 and 6066, and obsoletes RFCs 5077, 5246, and 6961. This document also specifies new requirements for TLS 1.2 implementations.



Secure channels for the Web



Security Goal: As long as the client is honest and the adversary does not know the server's private key, it cannot

- Inject forged data into the data stream (integrity)
- Distinguish the data stream from random bytes (confidentiality)

Goals of SSL/TLS

- End-to End Confidentiality
 - Encrypt communication between client and server applications
- End-to-End Integrity
 - Detect corruption of communication between client and server applications
- Required server authentication
 - Identity of server always proved to client
- Optional client authentication
 - Identity of client optionally proved to server
- Modular deployment
 - Intermediate layer between application and transport layers
- Handles encryption, integrity, and authentication on behalf of client and server applications



Certificates

- Public key certificate
 - Assurance by a third party that a public key is associated with an identity
 - E.g., QuoVadis certifies that the public key below is associated with University of Edinburgh web server
 - `d6 23 7e f5 e7 56 2c e3 50 d7 e1 4e 98 f4 cc 97 61 b2 ae 07 b8 b8 3d 6e 02 f4 9b c1 32 e5 56 bb 78 ea c0 2e 62 84 33 27 e4 2a 83 64 9f 53 cf e7 04 92 3e 4b 6d f8 55 68 a3 40 21 ff 70 66 a6 a4 50 a8 d9 87 54 97 fe ee 5a a4 b7 99 22 57 d2 df 84 35 5b 26 8c 09 2d 98 a9 74 0f e0 d9 d3 97 1d fd 80 8f 9c 5a e8 cc 78 0f 7b f7 95 2f 4f b4 07 cc 05 6d d3 0d 9a a4 37 fb ef 0d a8 b9 00 6f 4b d6 3f 80 04 38 09 9a e8 6e 27 aa a4 f1 20 7e 13 57 f4 9b ca cb 7f 3c 93 4d 1f e6 55 a1 4e 7c e2 06 a5 4b 8e 20 25 5d 07 e8 98 68 1a ea 0b cc fd 53 0a 66 93 be 31 b9 75 bb aa 04 b4 8f ac 56 8b 05 4d e1 68 2e 65 04 b6 da 93 49 60 2c c7 d2 74 fa 34 da 70 8c 7d bb f2 e6 b6 df 2a 8e 48 8f 15 ae 2f a0 ad 86 07 9c 9d c9 c0 1d 8b 06 b8 b9 ba`
- Certificate fields
 - Issuer aka certificate authority
 - ([QuoVadis](#))
 - Subject ([University of Edinburgh](#); www.ed.ac.uk)
 - Subject's public key parameters ([RSA2048](#))
 - Subject's public key
 - Validity period ([29/11/16-29/11/19](#))
 - Signature parameters ([SHA256](#); [PKCS #1](#), [RSA2048](#))
 - Signature



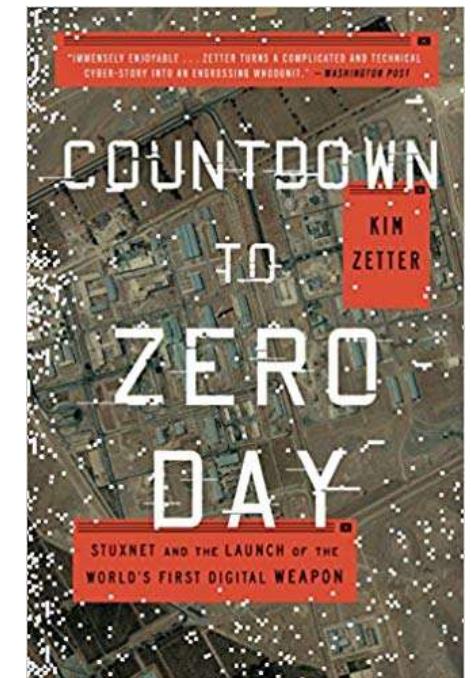
Chain of Trust and Revocation

- Transitive trust
 - Trust of (public key of) issuer implies trust of (public key of) subject
 - Issuer can be subject in another certificate
 - Chain of certificates
 - Root of trust?
 - Root certificates preconfigured in operating system and browser
- Certificate revocation
 - Mechanism to invalidate a previously issued certificate
 - E.g., when private key of the subject is compromised
- Revocation methods
 - List of revoked certificates posted on CA's website
 - Online verification service provided by CA (OCSP stapling)



Rogue Certificates: DigiNotar hacked in 2011

- Google noted a DigiNotar issued certificate for [google.com](#) that wasn't on Google's own list of Google certificates
- Used to impersonate Google mail web site and collect usernames and passwords
- Serious impact on Dutch government IT services
 - http://www.onderzoeksraad.nl/uploads/items-docs/1833/Rapport_Diginotar_EN_summary.pdf
- DigiNotar went bankrupt
- Do you know the CAs you are trusting?



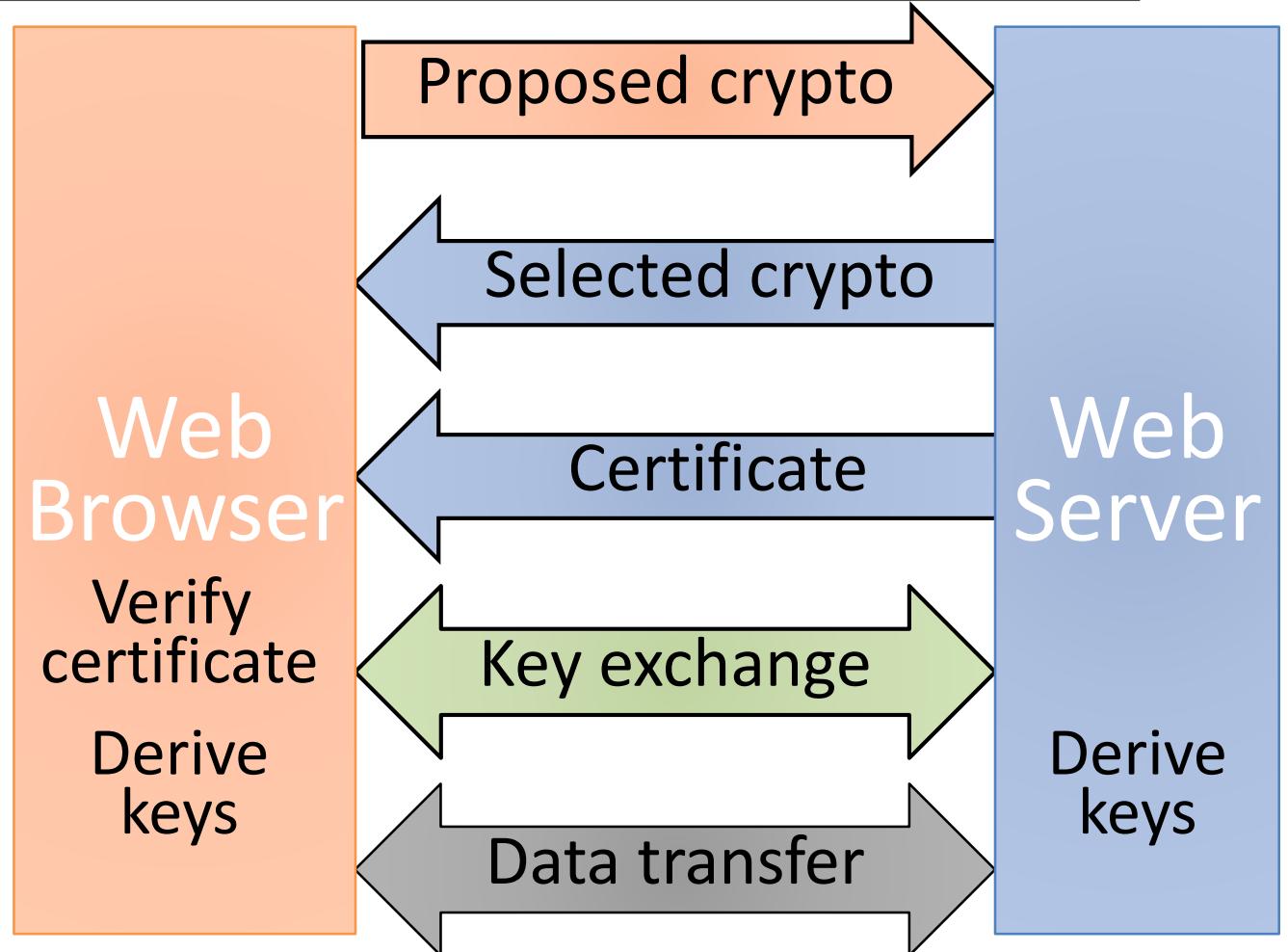
TLS Building Blocks

	Confidentiality	Integrity	Authentication
Setup	Public-key based key-exchange (RSA and DH)	Public-key digital signature (e.g., RSA)	Public-key digital signature (e.g., RSA)
Data transmission	Symmetric encryption (e.g., AES in CBC mode)	Hash-based MACs (e.g., HMAC using SHA256)	



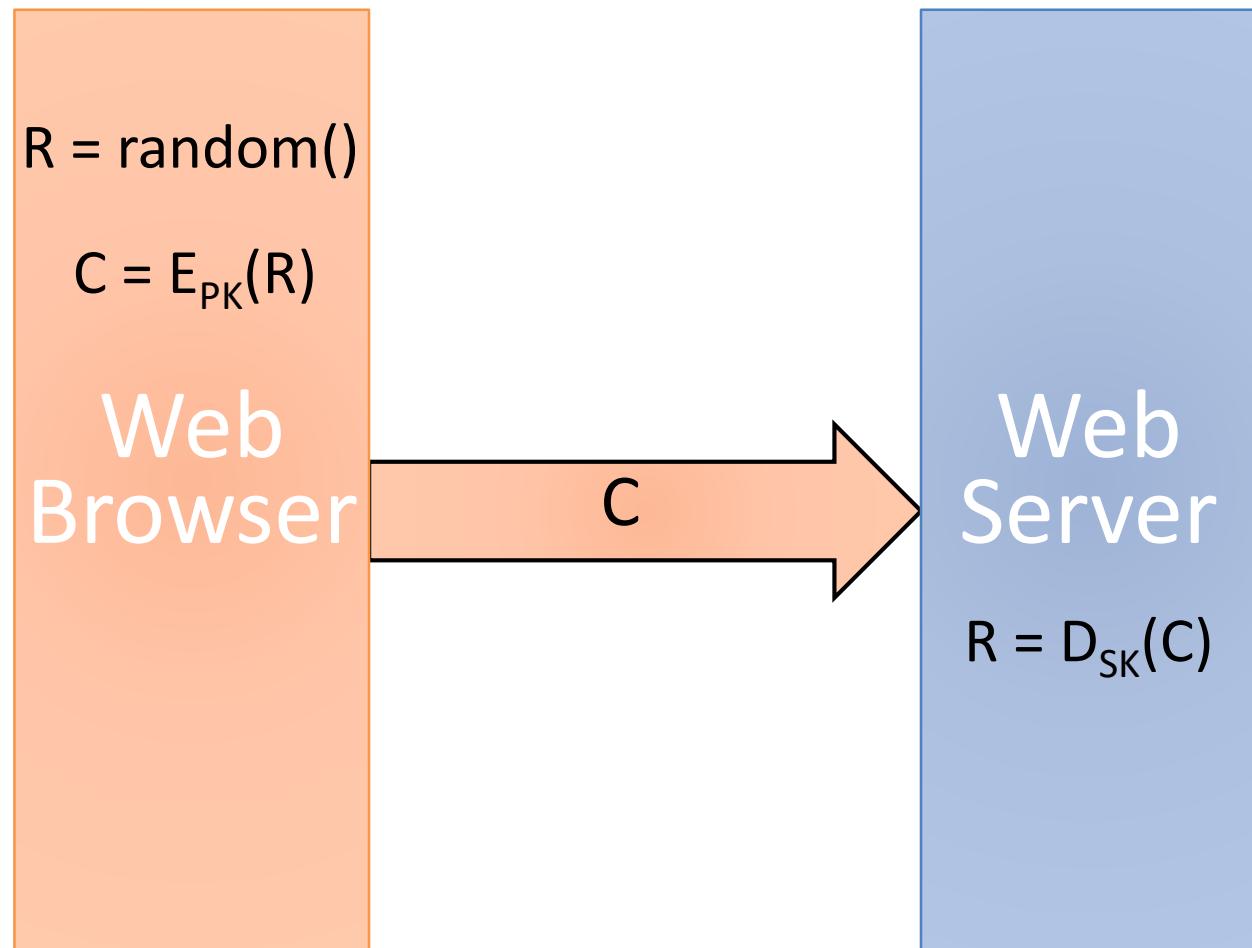
TLS Overview (Simplified)

- Browser sends supported crypto algorithms
- Server picks strongest algorithms it supports
- Server sends certificate (chain)
- Client verifies certificate (chain)
- Client and server agree on secret value R by exchanging messages
- Secret value R is used to derive keys for symmetric encryption and hash-based authentication of subsequent data transfer



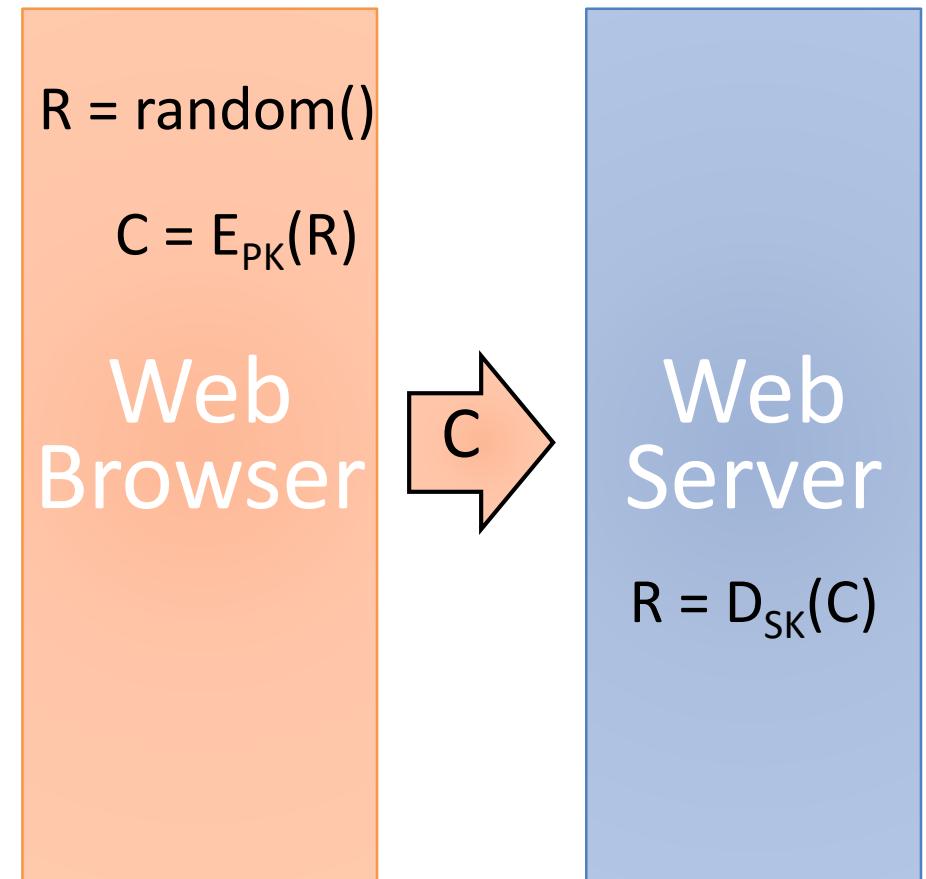
Basic Key Exchange

- Called **RSA key exchange** for historical reasons
- Client generates random secret value R
- Client encrypts R with public key, PK , of server $C = E_{PK}(R)$
- Client sends C to server
- Server decrypts C with private key, SK , of server
 $R = D_{SK}(C)$



Forward Secrecy

- Compromise of public-key pairs private keys does not break confidentiality of past messages
- TLS with basic key exchange does not provide forward secrecy
 - Attacker eavesdrop and stores communication
 - If server's private key is compromised, attacker finds secret value R in key exchange and derives encryption keys





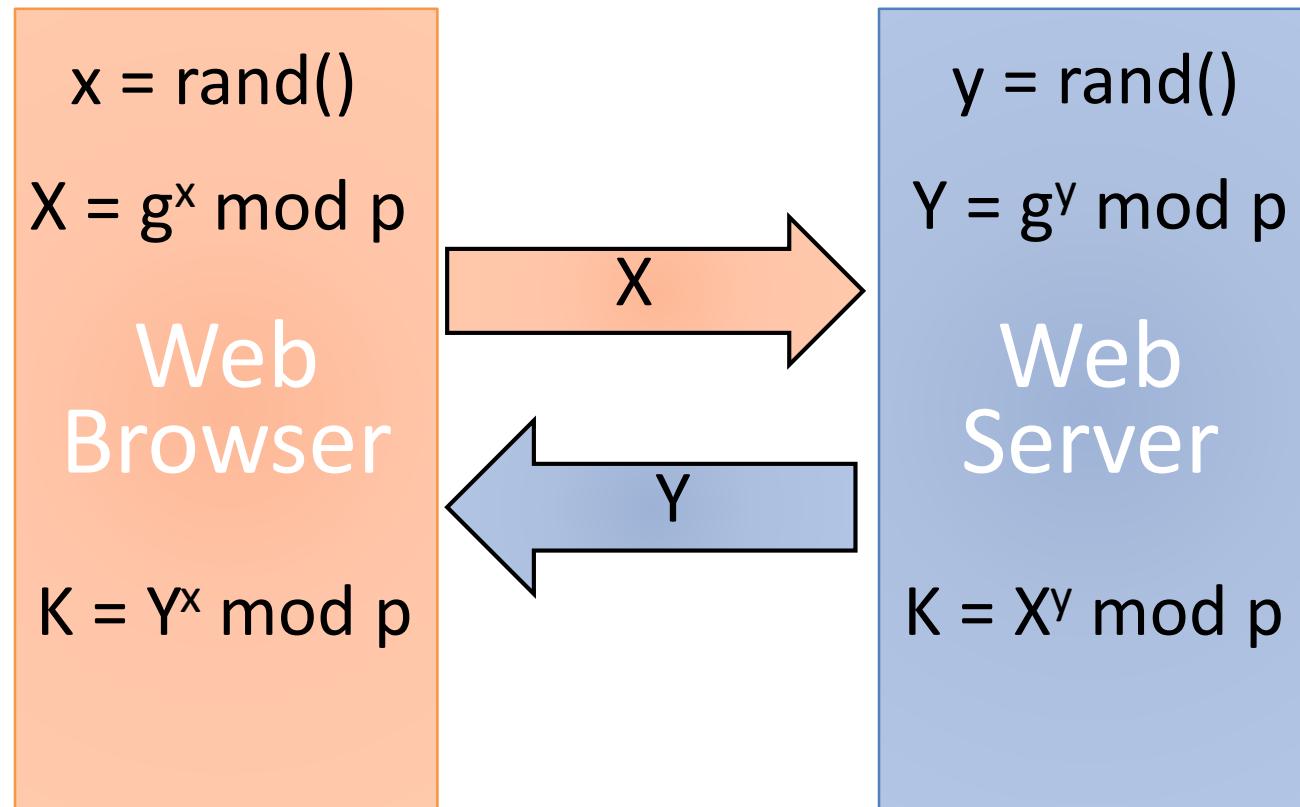
Source [ACM](#)

Achieves forward secrecy Diffie Hellman Key Exchange

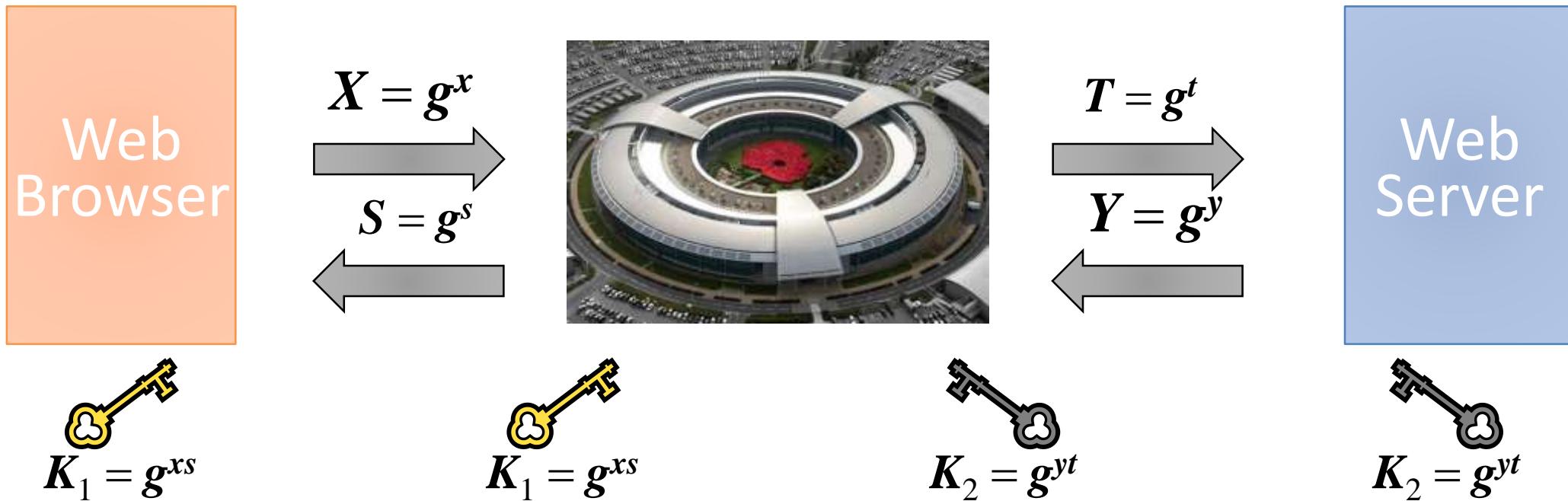


Source [ACM](#)

- Public parameters: prime p and generator g of Z_p
- Client generates random x and computes $X = g^x \text{ mod } p$
- Server generates random y and computes $Y = g^y \text{ mod } p$
- Client sends X to server
- Server sends Y to client
- Client and server compute $K = g^{xy} \text{ mod } p$



Attacker in the Middle

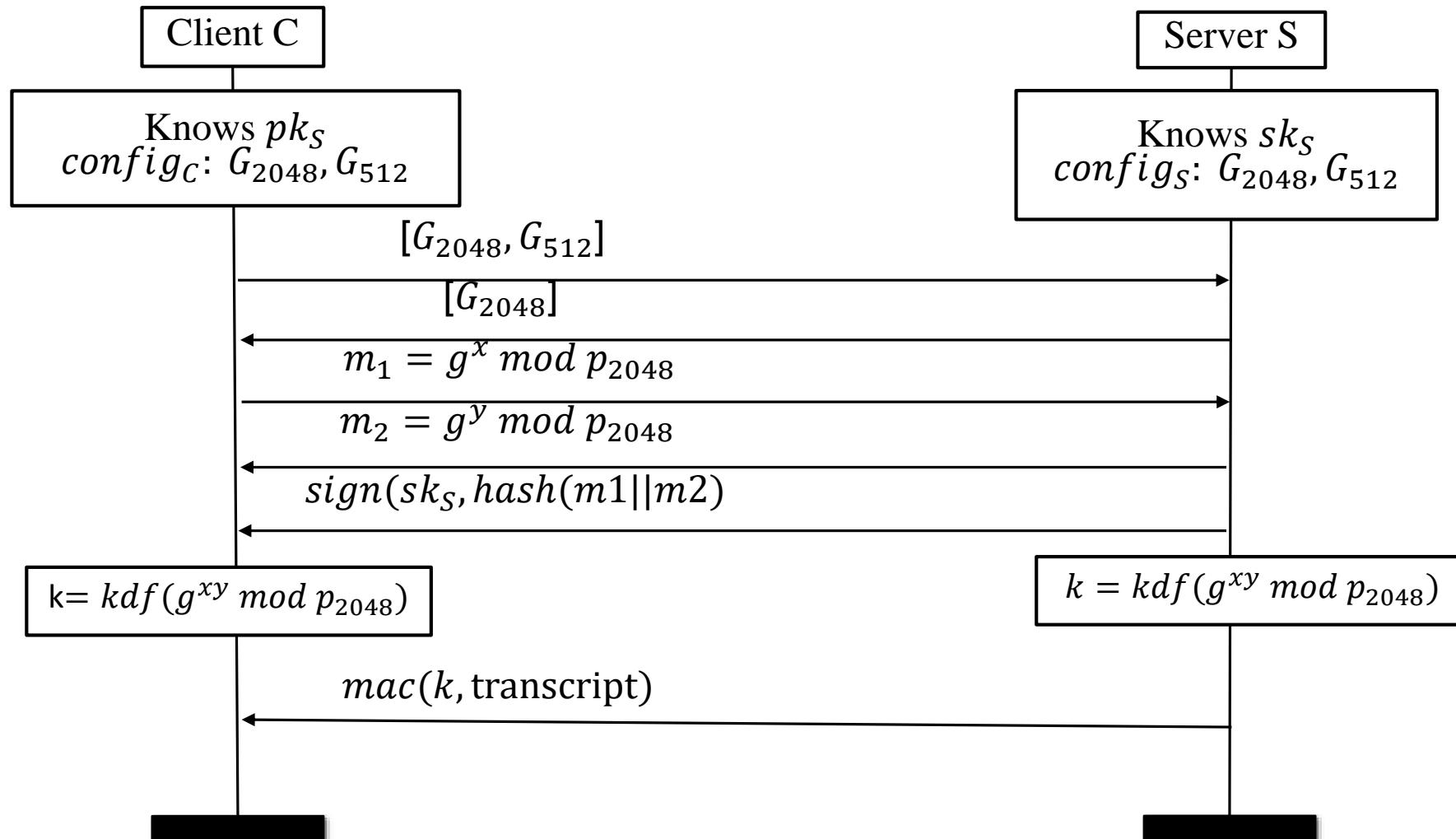


Solution

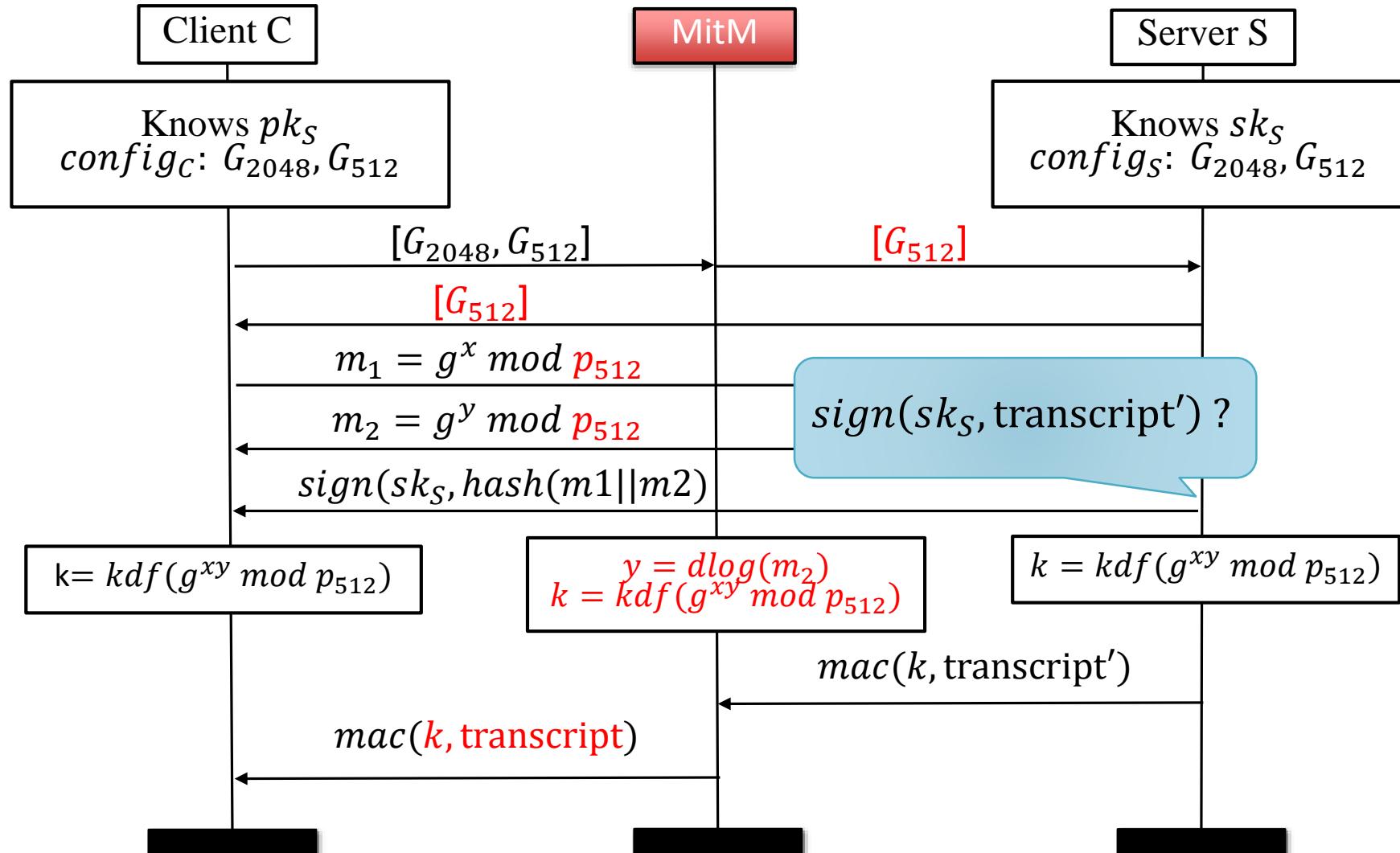
- Browser and server send signed X and Y
- Requires each to know the public key of the other
- One sides authentication if only server signs



Signed DH key exchange



LOGJAM



Back to the roots $\sqrt[e]{c}$ [RSA78]

There was RSA

$$E(m) = m^e \bmod n$$

$$n = pq - \text{product of primes}$$

RSA is homomorphic

$$(m^e \cdot s^e)^d = (m \cdot s)^{ed} = m \cdot s$$



Padding Attacks [B98]

Bleichenbacher Attack on PKCS#1:

Pick s_i adaptively. For accepted $c \times s_i^e$ attacker knows that

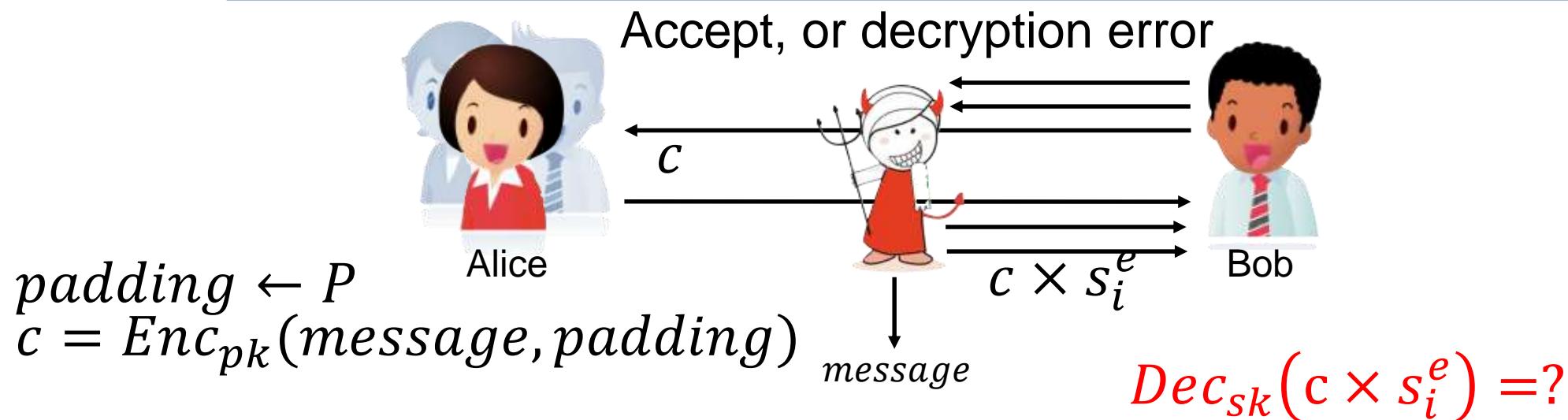
$$M \times s_i = (01\ 02\ \text{*****}\ 00\ \text{*****}).$$

Build system of inequalities:

$$(01\ 02\ 00\ \dots\ 00) \leq M \times s_i \leq (01\ 02\ ff\ \dots\ ff)$$

PKCS#1 standard for RSA:

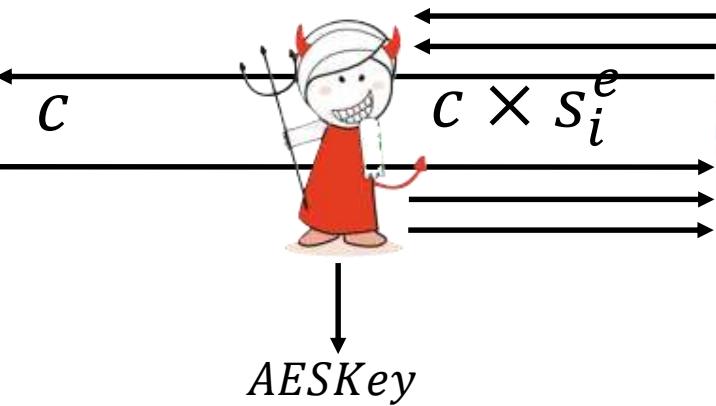
$Enc_{(n,e)}(\text{message}, \text{padding}) = M^e$ where $M = (01\ 02\ \text{padding}\ 00\ \text{message})$



Padding Attacks [B98]



Accept, or decryption error



What We Have Learned

- Goals and history of the SSL/TLS protocol
- TLS Certificates, chain of trust, and revocation
- Overview of the TLS protocol
- Diffie-Hellman key exchange and forward secrecy
- Logjam and Bleichenbacher attack



References

- [RFC 8445](#) - The Transport Layer Security (TLS) Protocol Version 1.3 (2018)
- [Logjam](#) attack (2015)

Firewalls and Intrusion Detection

MARKULF KOHLWEISS
COMPUTER SECURITY

Some slides adapted from those by Kami Vaneia, and Michael Goodrich





INDEPENDENT

RUSSIA PLANS TO BRIEFLY DISCONNECT FROM THE INTERNET TO

When it is passed, the Digital Economy National Program legislation requires the local internet, known as the Runet, to pass through exchange points managed by Russia's telecommunications regulator Roskomnazor.

The test will see the Runet separated from the wider internet for a short period of time at some point before 1 April, according to local news agency RosBiznesKonsalting (RBK).

Once in force, the Digital Economy National Program will simultaneously protect Russia in the event of cyber war, while also filtering internet traffic to the country in a similar way to the 'Great Firewall of China'.





Genes, chips, qubits, rockets, reactors, surveillance, and sand—the tools of a rising superpower

YOUNG PEOPLE IN CHINA DON'T KNOW THE INTERNET WE DO – AND THEY LIKE IT THAT WAY

2

'The Chinese apps have got everything'

Even if the western apps and sites make it into China, they may face apathy from young people.

Two economists from Peking University and Stanford University concluded this year, after an 18-month survey, that Chinese college students were indifferent about having access to uncensored, politically sensitive information. They had given nearly 1,000 students at two Beijing universities free tools to bypass censorship, but found that nearly half the students did not use them. Among those who did, almost none spent time browsing foreign news websites that were blocked.

"Our findings suggest that censorship in China is effective, not only because the regime makes it difficult to access sensitive information, but also because it fosters an environment in which citizens do not demand such information in the first place," the scholars wrote.



MIT
Technology
Review

T The China issue

120TH ANNIVERSARY ISSUE

VOL. 120 JAN/FEB 2019 \$9.99 USD

ISSUE 1 \$10.99 CAD

中国制造



Genes, chips, qubits, rockets, reactors, surveillance,
and sand—the tools of a rising superpower

YOUNG PEOPLE IN CHINA DON'T KNOW THE INTERNET WE DO – AND THEY LIKE IT THAT WAY

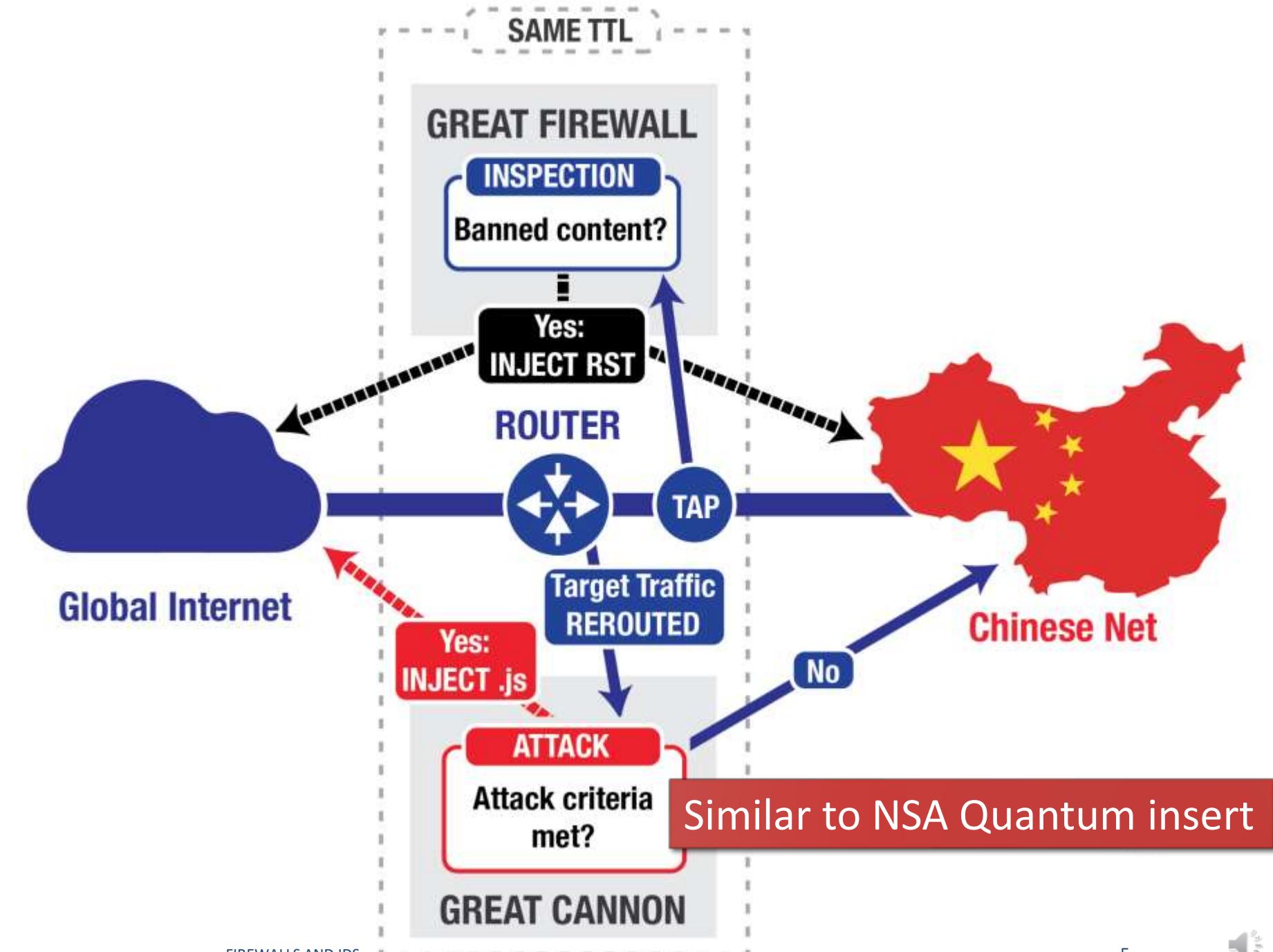
'The Chinese apps have got everything'

When Chinese hackers declared war on the rest of us

Many thought the internet would bring democracy to China. Instead it empowered rampant government oppression, and now the censors are turning their attention to the rest of the world.

Old news...

- <https://citizenlab.ca/2015/04/chinas-great-cannon/>



End-to-End Principle

Application-specific features

- should reside in the communicating end nodes of the network,
- rather than in intermediary nodes, such as routers,
that exist to establish the network

Examples:

- reliability from unreliable parts
- end-to-end encryption
and authentication

Counter examples:

- content distribution networks
- network address translation



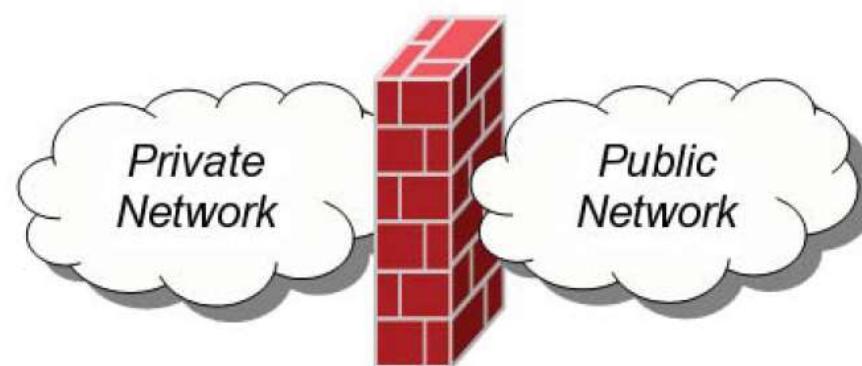
Today

- Firewalls
- Network Address Translation (NAT)
- Intrusion Detection Systems (IDS)

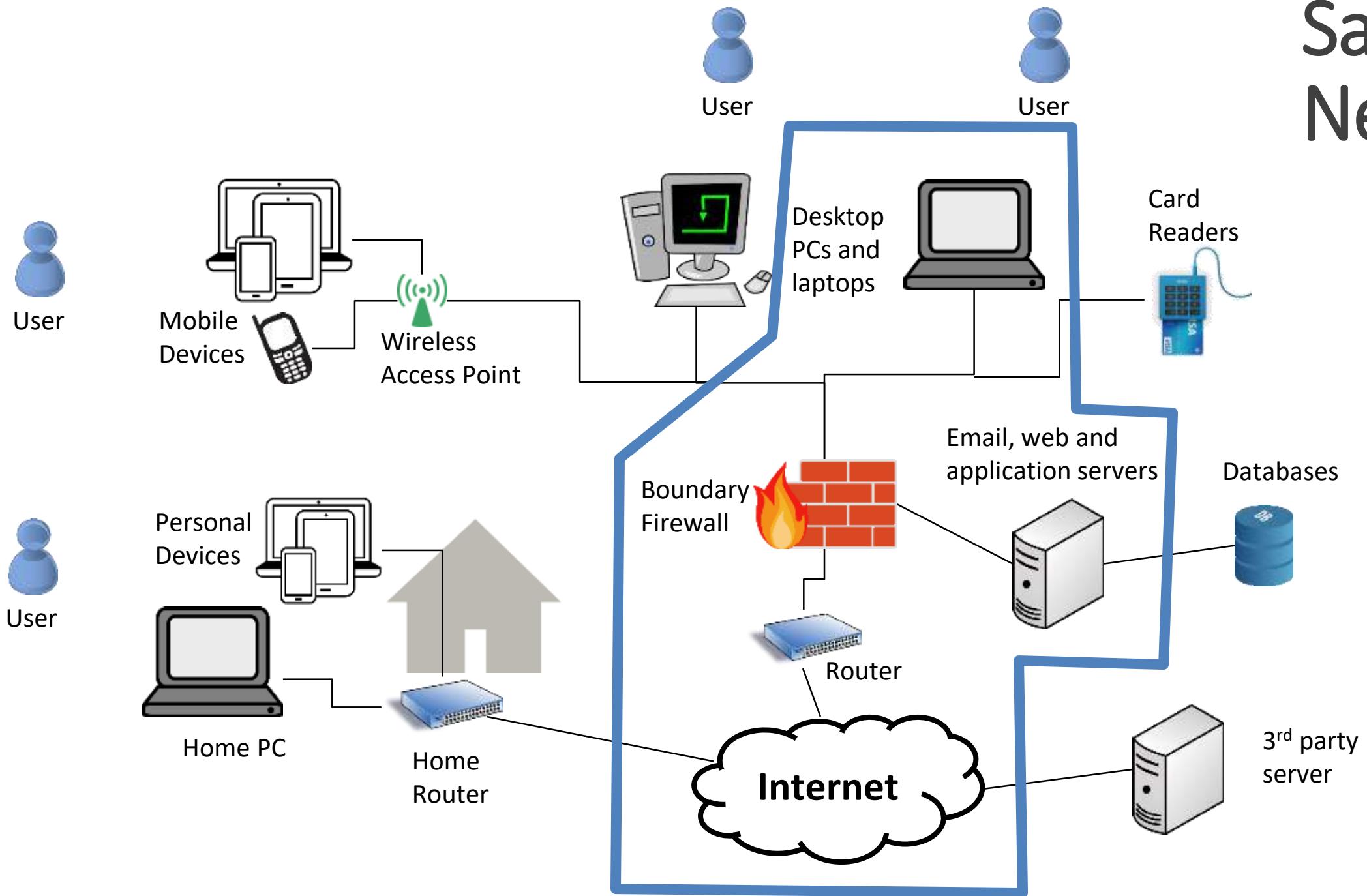


Firewalls

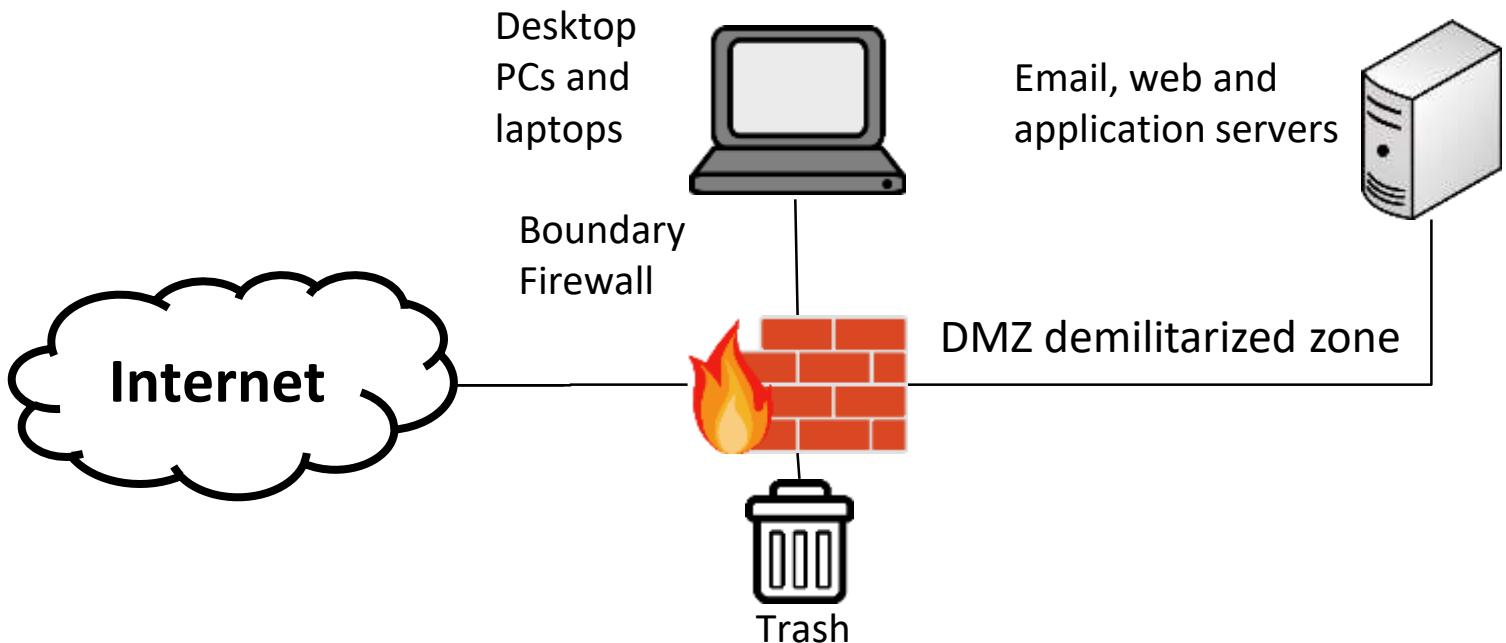
- A **firewall** is a security measures designed to prevent **unauthorized electronic access** to a networked computer system.
- Intuition: Similar to firewalls in building construction. Intent is to isolate one “network” or “compartment” from another.



Sample Network



- Malicious actions from the **Internet** AND **local network**
- Firewall applies a set of rules called **firewall policies**
- Based on rules, it allows or denies the 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



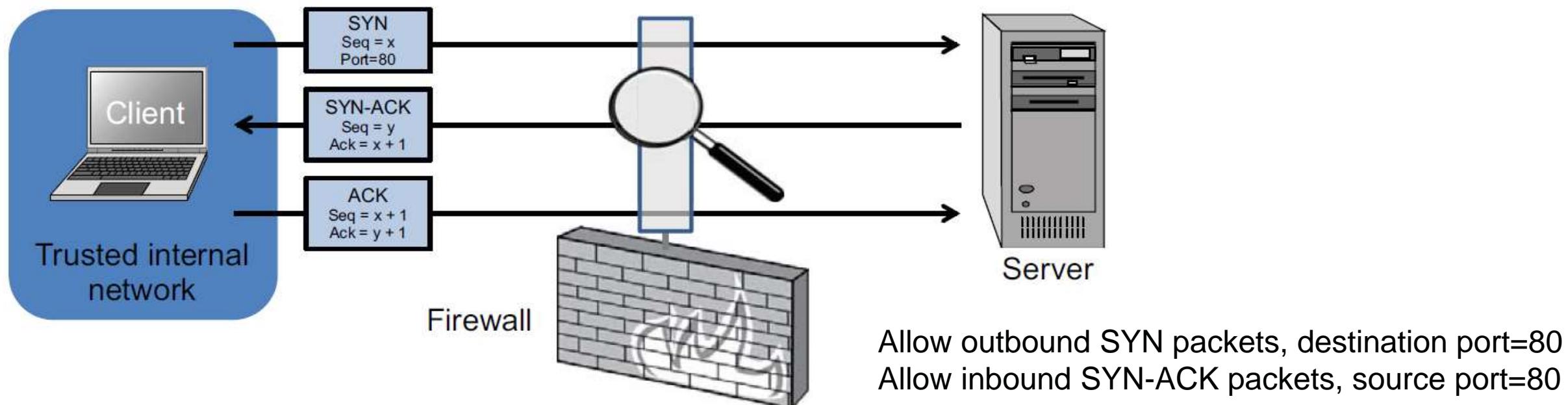
Firewall Types

- **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, ...)



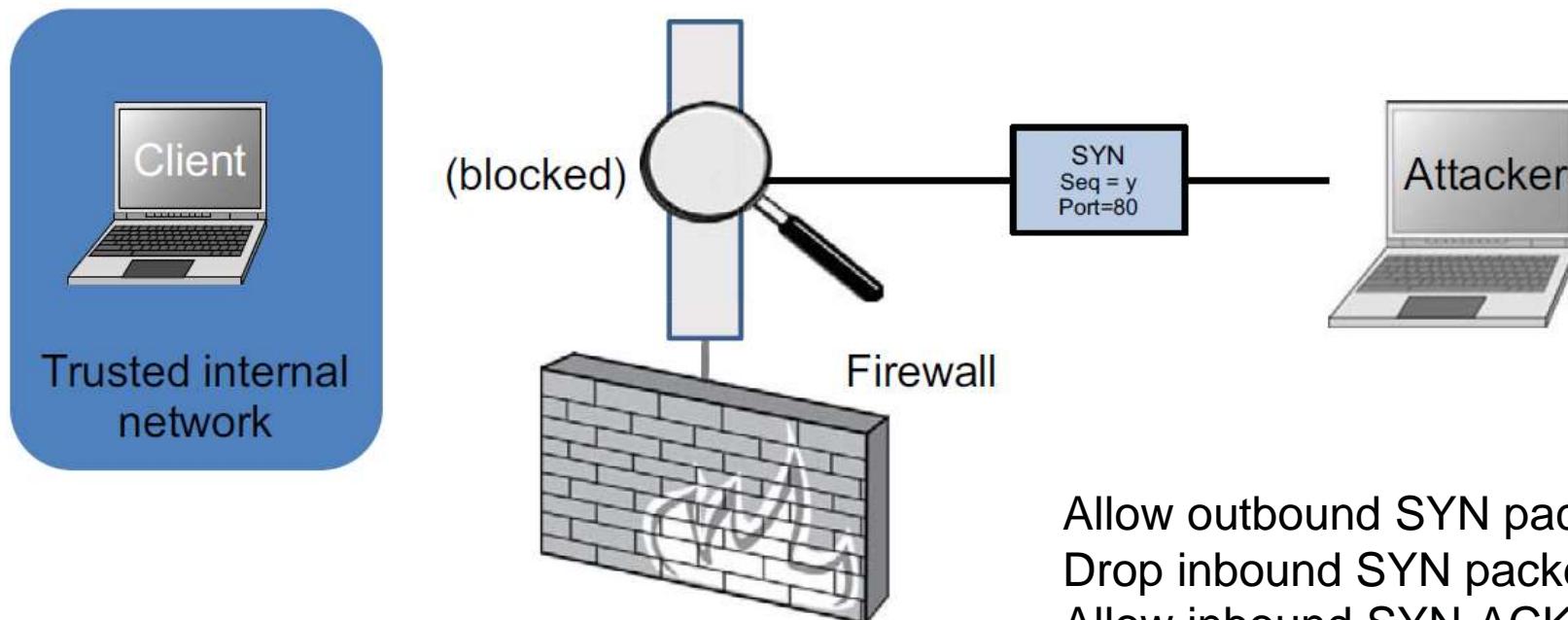
Stateless Firewalls

- 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.



Stateless Restrictions

- Stateless firewalls may have to be fairly restrictive in order to prevent most attacks.



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



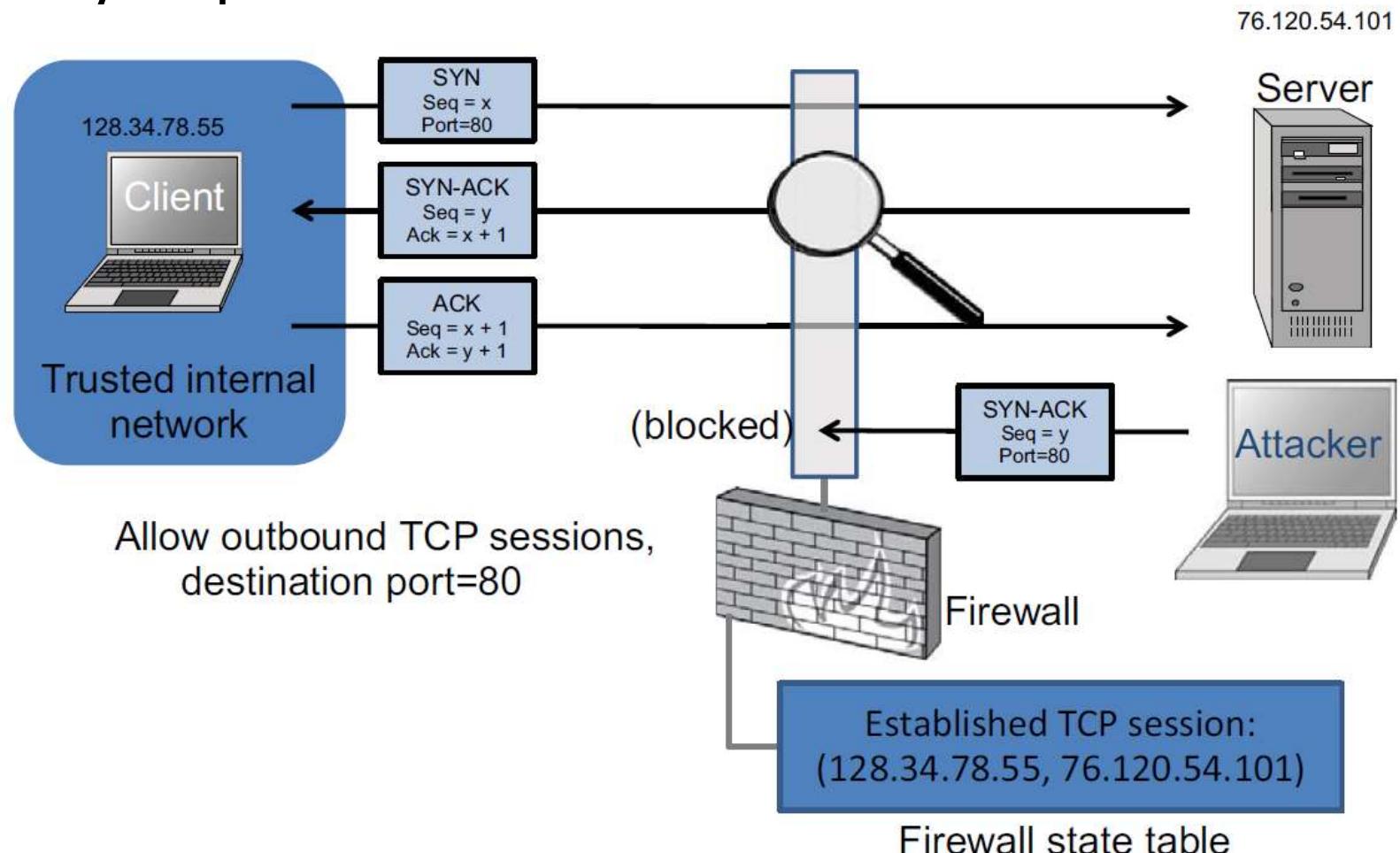
Stateful Firewalls

- **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:



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

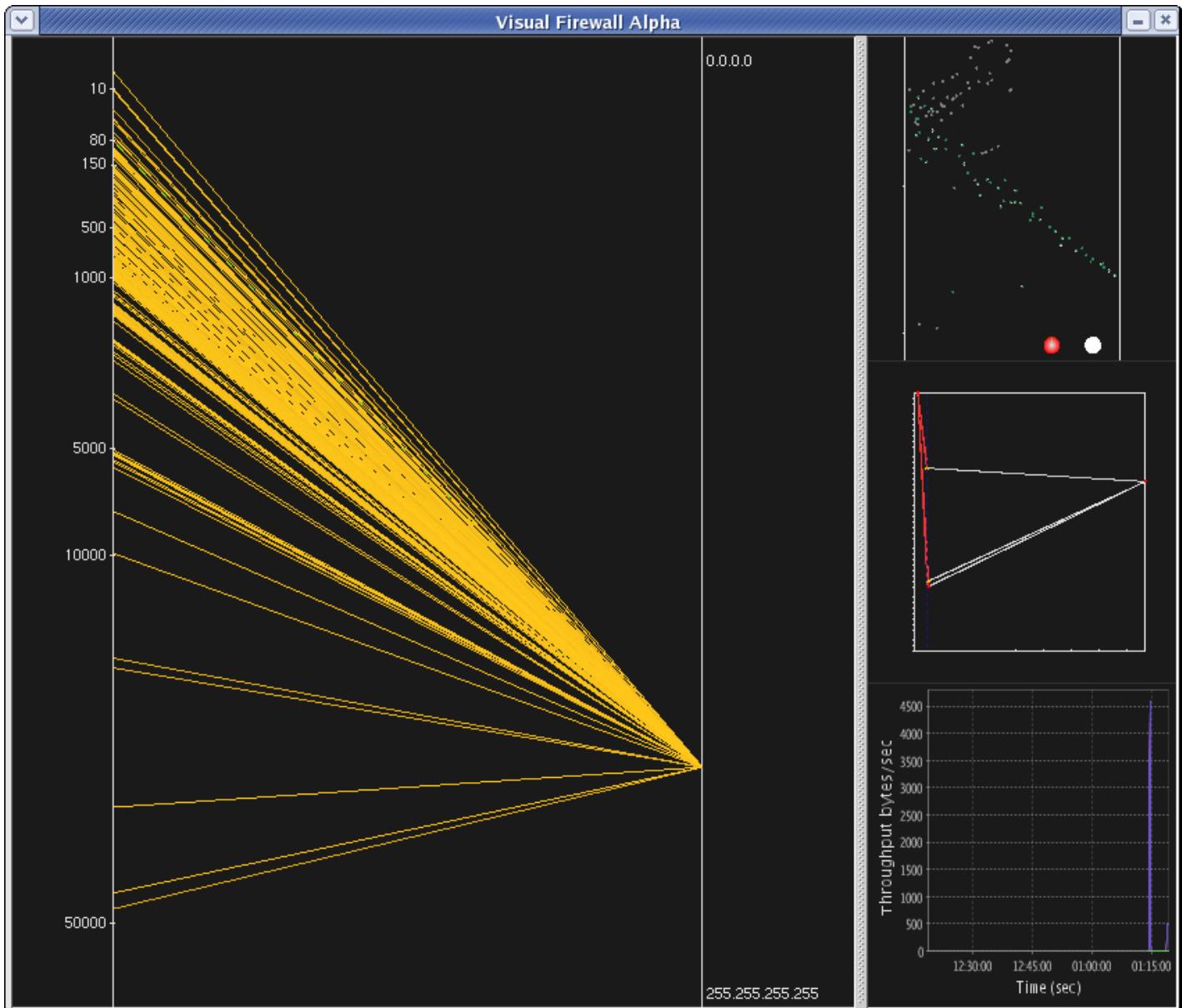


Image: <http://chrislee.dhs.org/projects/visualfirewall.html>



Application layer firewall/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
- Any rootkit type software can disable the firewall



Think-pair-share

- **Think** quietly to yourself for 1 minute
- **Pair** and discuss with your neighbour for 3 minutes
- **Share** with the class – group discussion

Think about the different types of firewalls.

- Do they violate the end-to-end principle?

Application-specific features should reside in the communicating end nodes of the network



Network Address Translation (NAT)



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

Command Prompt

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

C:\Users\kamiv_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 . :
```

My computer as seen from a remote server

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

My IP
previously
showed as:
172.20.106.96

What
happened?

```
HTTP_ACCEPT: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
HTTP_ACCEPT_ENCODING: gzip, deflate
HTTP_ACCEPT_LANGUAGE: en-US,en;q=0.5
HTTP_CONNECTION: keep-alive
HTTP_COOKIE: __utma=145846189.271110778.1474893692.1474893692.1474893692.1; __utmc=1474893692.1.1.utmcsr=google|utmccn=(organic)|utmcmd=organic|utmctr=(not%20provided); PRUM_EPISODES=s=1474893750106&r=http%3A//www.hashemian.com/whoami/
HTTP_HOST: www.hashemian.com
HTTP_REFERER: https://www.google.co.uk/
HTTP_UPGRADE_INSECURE_REQUESTS: 1
HTTP_USER_AGENT: Mozilla/5.0 (Windows NT 6.3; WOW64; rv:49.0) Gecko/20100101 Firefox/49.0
REMOTE_ADDR: 192.41.131.255
REMOTE_PORT: 7535
REQUEST_METHOD: GET
REQUEST_TIME: 1474906336
REQUEST_URI: /whoami/
SERVER_ADDR: 173.162.146.61
SERVER_NAME: www.hashemian.com
SERVER_PORT: 80
SERVER_PROTOCOL: HTTP/1.1
SERVER_SIGNATURE:
SERVER_SOFTWARE: Apache
```

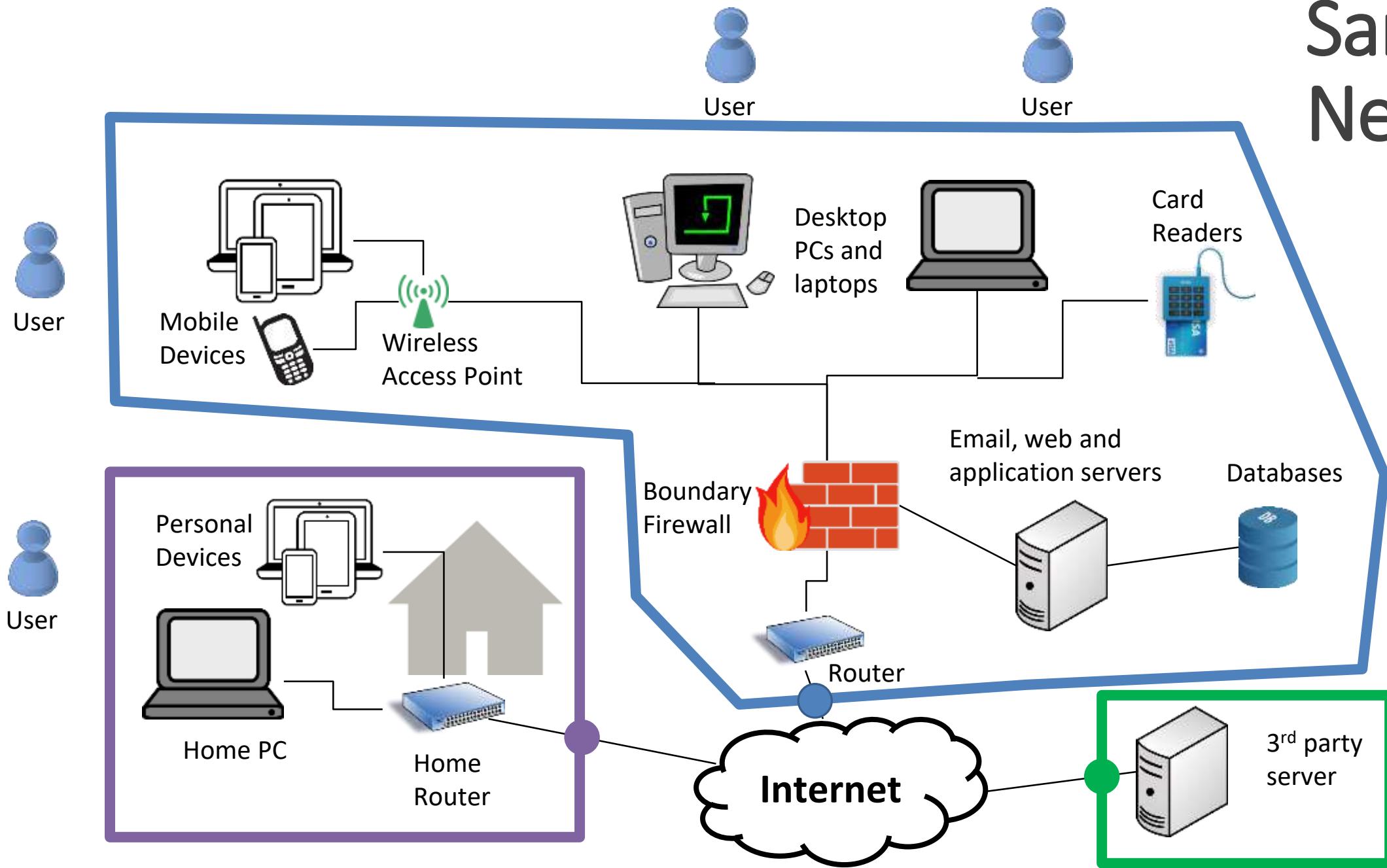


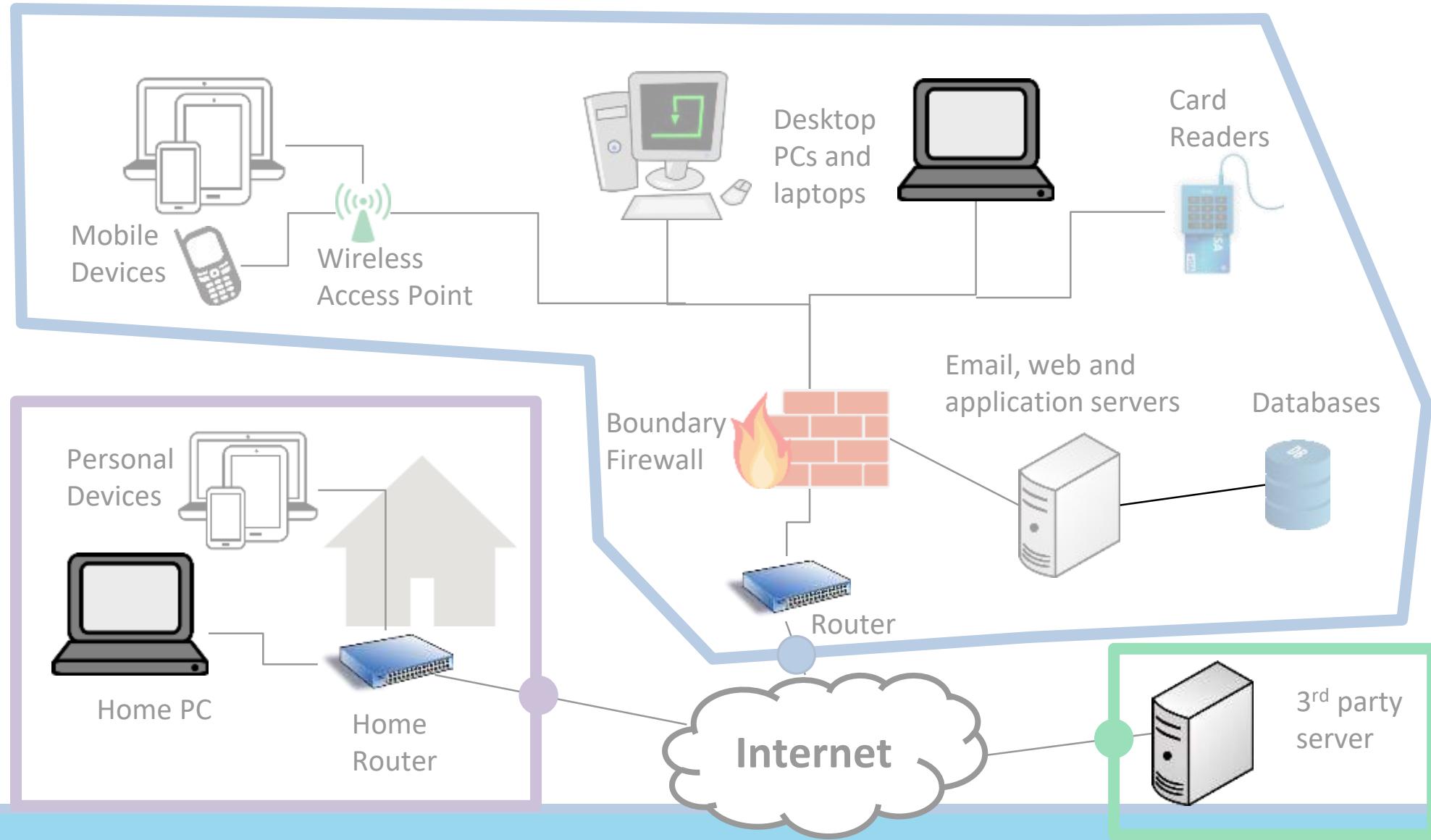
IPv4 and address space exhaustion

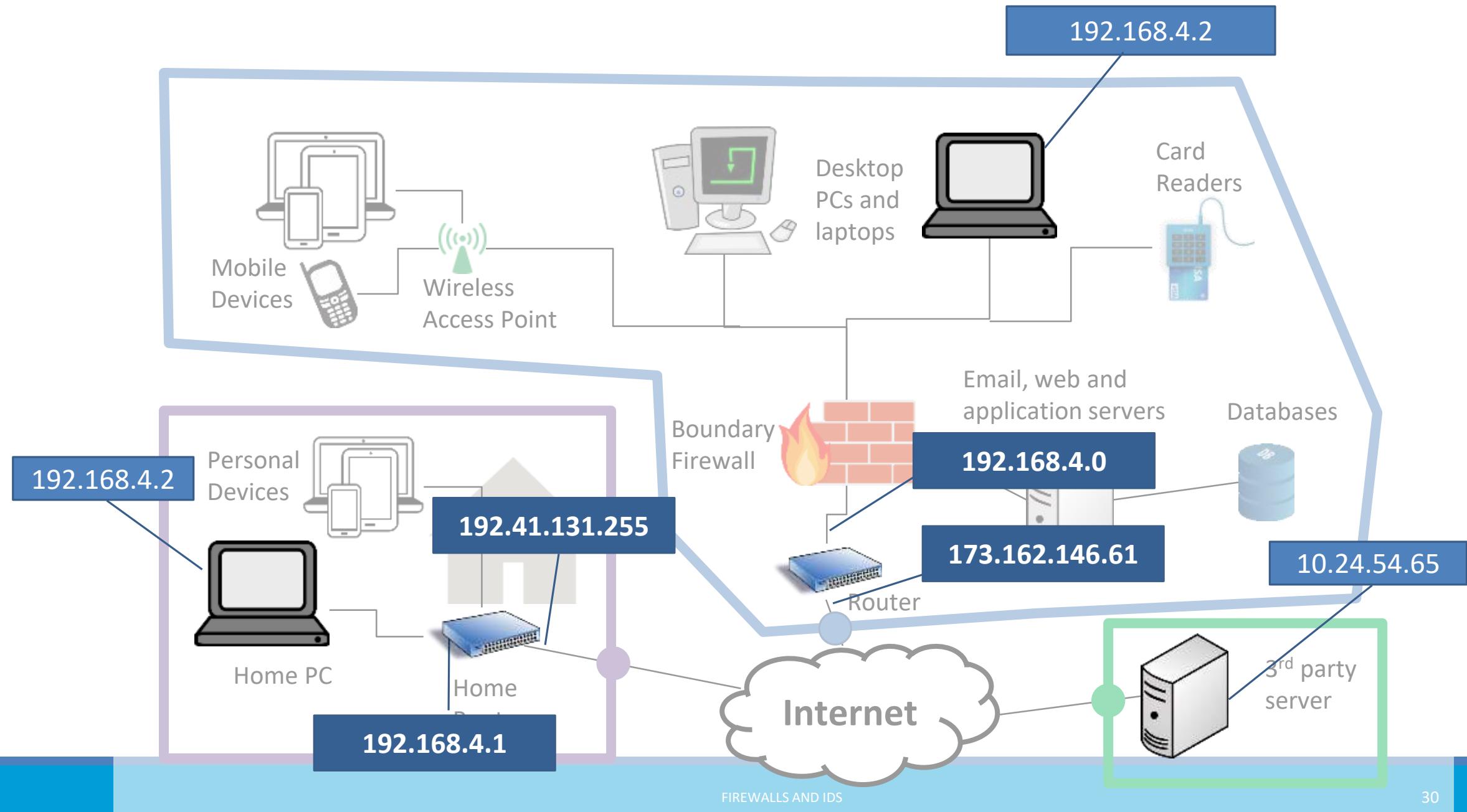
- Version 4 of the Internet Protocol
 - 192.168.2.6
- There are less than 4.3 billion IPv4 addresses available
- We do not have enough addresses for every device on the planet
- Answer: Network Address Translation
 - Internal IP different than external IP
 - Border router maps between its own IP and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

Command Prompt

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

C:\Users\kamiv_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:

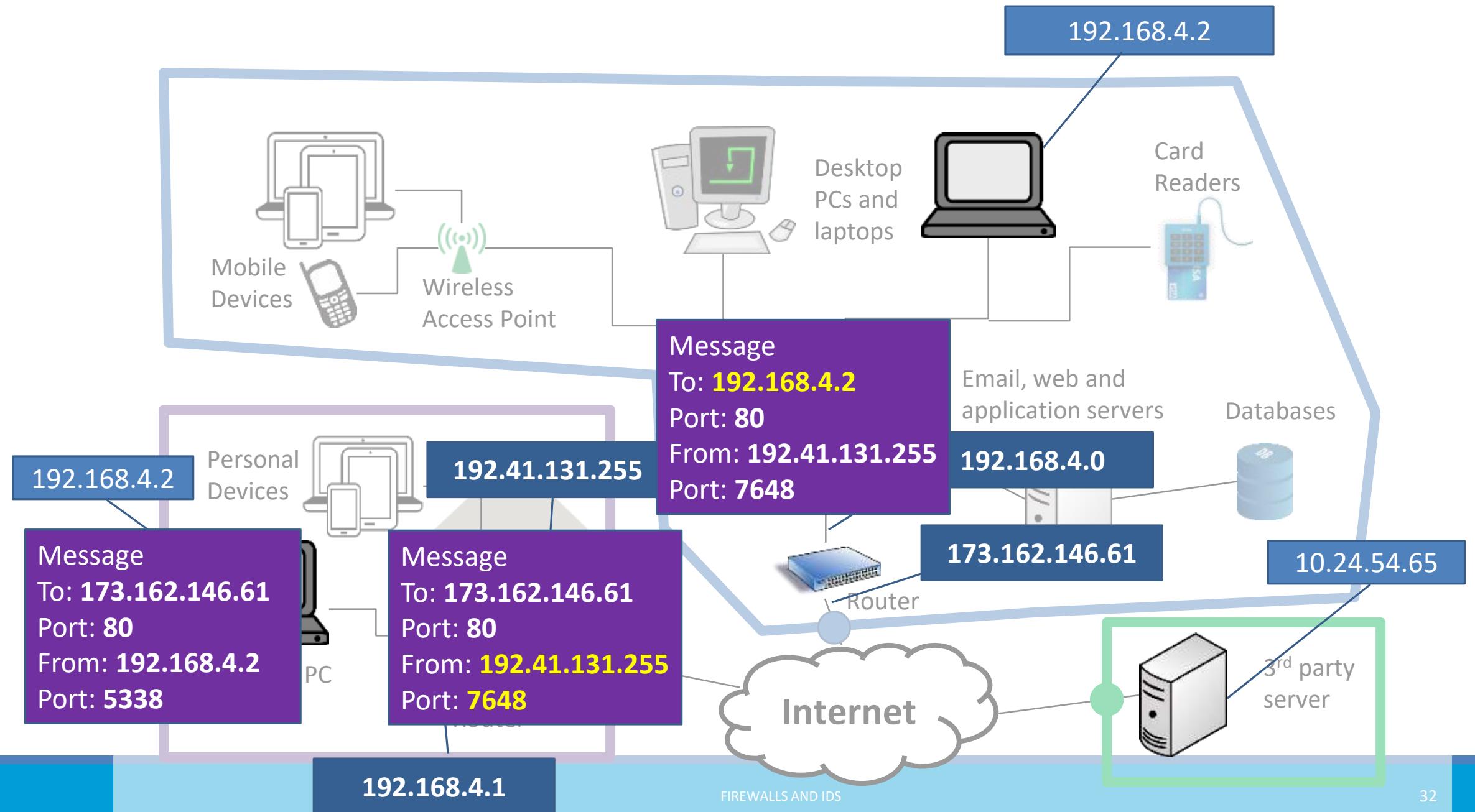
    Connection-specific DNS Suffix . . . . . : ed.ac.uk
    Link-local IPv6 Address . . . . . : fe80::483:b9e3:91bd:d0d1%4
    IPv4 Address. . . . . : 172.20.106.96
    Subnet Mask . . . . . : 255.255.240.0
    Default Gateway . . . . . : 172.20.111.254

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?

<https://www.youtube.com/watch?v=ISwB49vO0ys>



Intrusion Detection Systems (IDS)



Firewalls are preventative, IDS detects a potential incident in progress

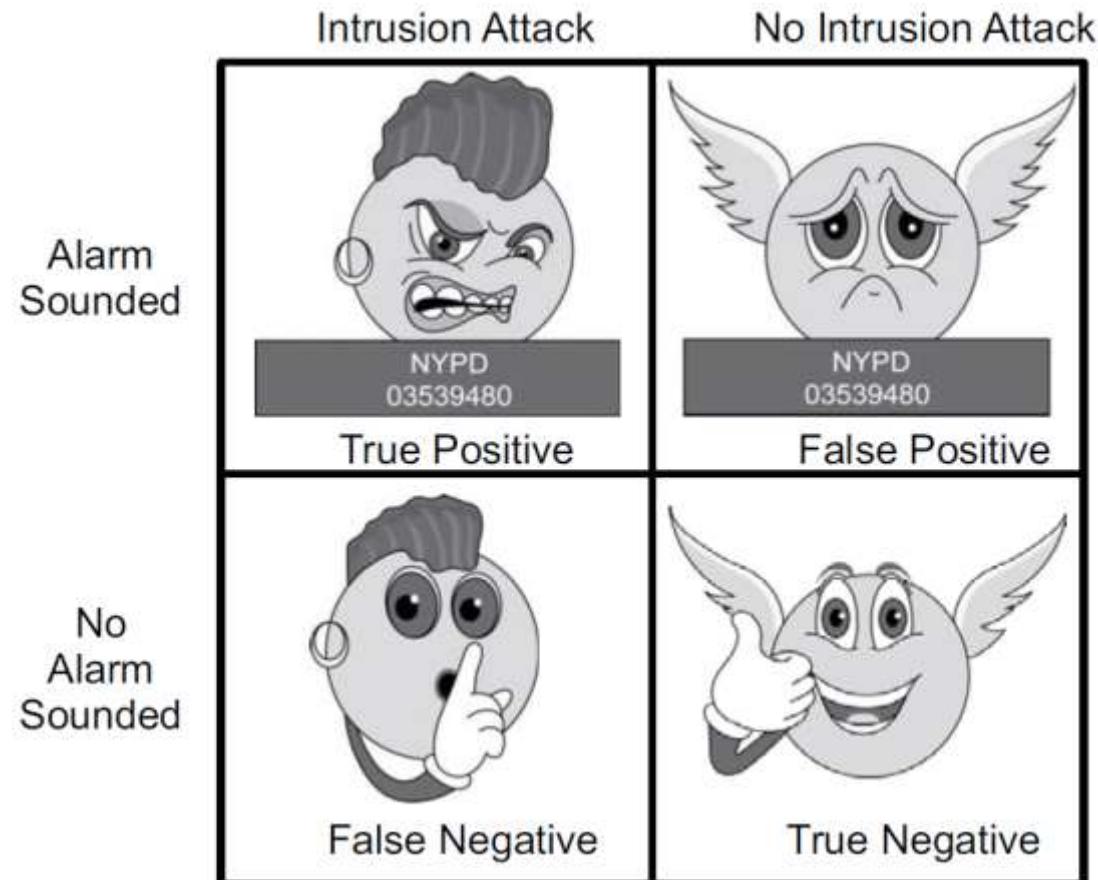
- At some point you have to let some traffic into and out of your network (otherwise users get upset)
- Most security incidents are caused by a user letting something into the network that is malicious, or by being an insider threat themselves
- These cannot be prevented or anticipated in advance
- The next step is to identify that something bad is happening quickly so you can address it





Possible Alarm Outcomes

- Alarms can be sounded (positive) or not (negative)



Rule-Based Intrusion Detection

- Rules identify the types of actions that match certain known intrusion attack. Rule encode a **signature** for such an attack.
- Requires that admin anticipate attack patterns in advance
- Attacker may test attack on common signatures
- Impossible to detect a new type of attack
- High accuracy, low false positives



Statistical Intrusion Detection

- Dynamically build a statistical model of acceptable or “normal” behavior and flag anything that does not match
- Admin does not need to anticipate potential attacks
- System needs time to warm up to new behavior
- Can detect new types of attacks
- Higher false positives, lower accuracy



Base-Rate Fallacy

Suppose an IDS is 99% accurate, having a 1% chance of false positives or false negatives.

Suppose further...

- An intrusion detection system generates 1,000,100 log entries.
- Only 100 of the 1,000,100 entries correspond to actual malicious events.
- Because of the success rate of the IDS, of the 100 malicious events, 99 will be detected as malicious, which means we have **1 false negative**.
- Nevertheless, of the 1,000,000 benign events, 10,000 will be mistakenly identified as malicious. That is, we have **10,000 false positives!**
- Thus, there will be 10,099 alarms sounded, 10,000 of which are false alarms. That is, roughly 99% of our alarms are false alarms.



Number of alarms is a big problem

- In the **2013 Target breach** the IDS did correctly identify that there was an attack on the Target network
- There were too many alarms going off to investigate all of them in great depth
- Some cyberattack insurance policies state that if you know about an attack and do nothing they will not cover the attack.
- Having a noisy IDS can potentially be a liability



Questions

Piazza: <https://piazza.com/class/jqw6jfkkzns3l0>

“Hard statistical data prove that emotional support directly impacts every metric of academic performance — and, as it turns out, every other aspect of our lives as well.” Mark Greene



In-depth analysis of the Great Firewall of China

Chao Tang

December 14, 2016

TCP Reset

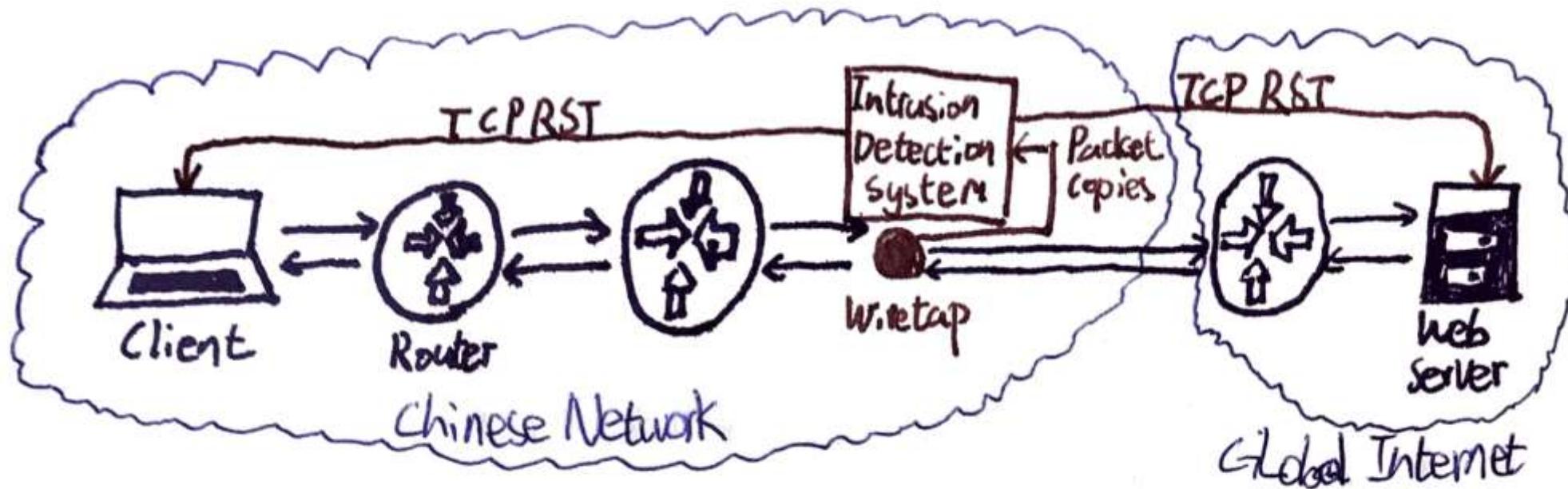


Figure-1: An illustration of TCP Reset

DNS Tampering

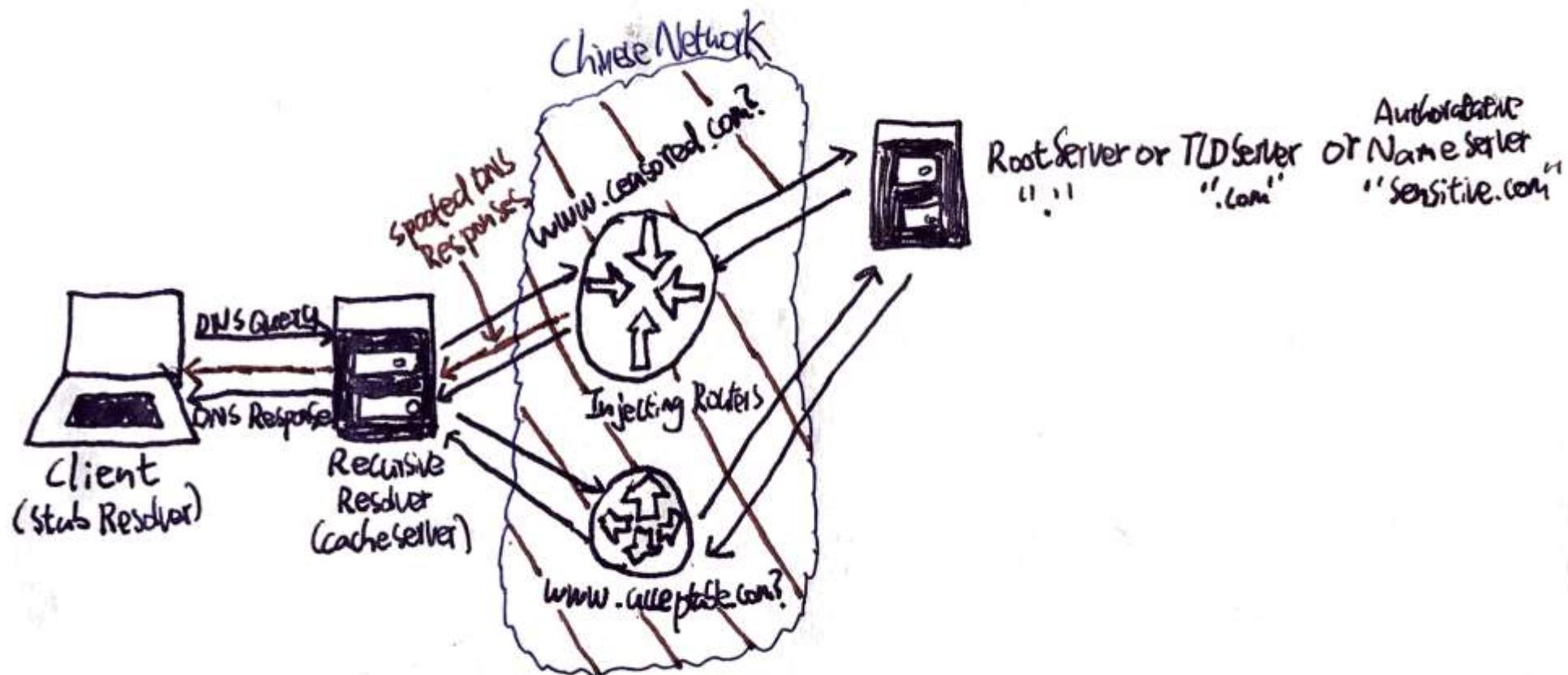


Figure-5: An illustration of DNS Tampering.

Cryptographic protocols

Myrto Arapinis
School of Informatics
University of Edinburgh

February 15, 2019

Context

Applications exchanging **sensitive data** over a **public network**:

- ▶ eBanking,
- ▶ eCommerce,
- ▶ eVoting,
- ▶ ePassports,
- ▶ Mobile phones,
- ▶ ...

Context

Applications exchanging **sensitive data** over a **public network**:

- ▶ eBanking,
- ▶ eCommerce,
- ▶ eVoting,
- ▶ ePassports,
- ▶ Mobile phones,
- ▶ ...

A malicious agent can:

- ▶ record, alter, delete, insert, redirect, reorder, and reuse past or current messages, and inject new messages
→ **the network is the attacker**
- ▶ control dishonest participants

The attacker controls the network (1)

Network Utility

Info | Netstat | Ping | Lookup | **Traceroute** | Whois | Finger | Port Scan

Enter the network address to trace an internet route to.

(ex. 10.0.2.1 or www.example.com)

Trace

```
Traceroute has started.

traceroute to star-mini.c10r.facebook.com (157.240.0.35), 64 hops max, 72 byte packets
 1 knussen (129.215.91.246) 0.435 ms 0.211 ms 0.185 ms
 2 vlan160.kb9-msfc.net.ed.ac.uk (129.215.160.254) 0.502 ms 0.435 ms 0.467 ms
 3 vlan688.s-pop.eastman.ja.net (194.81.57.211) 0.886 ms 0.824 ms 0.876 ms
 4 ae2.leedaq-sbr1.ja.net (146.97.41.33) 4.578 ms 4.550 ms 4.574 ms
 5 ae30.manchk-sbr1.ja.net (146.97.33.45) 7.165 ms 7.121 ms 7.133 ms
 6 port-channel205.car1.manchester1.level3.net (195.50.119.97) 21.449 ms 10.438 ms 205.486 ms
 7 4.15.154.86 (4.15.154.86) 111.234 ms 111.232 ms 111.284 ms
 8 po103.psw01c.mia1.tfbnw.net (157.240.32.223) 110.868 ms 110.802 ms 110.810 ms
 9 157.240.36.71 (157.240.36.71) 110.913 ms 110.802 ms 110.745 ms
10 edge-star-mini-shv-02-mia1.facebook.com (157.240.0.35) 112.594 ms 112.943 ms 112.861 ms
```

The attacker controls the network (2)

The Register®
Biting the hand that feeds IT

A DATA CENTRE SOFTWARE NETWORKS SECURITY TRANSFORMATION DEVOPS BUSINESS HARDWARE

Networks

Verizon, BT, Vodafone, Level 3 'let NSA jack into Google, Yahoo! fiber'

Telcos cooperated with g-men in data slurp, claim sources



27 Nov 2013 at 02:19, Shaun Nichols



2

In October, NSA whistleblower Edward Snowden claimed Uncle Sam's spies tapped into the optic-fiber cables linking the data centers of Google and Yahoo!

The attacker controls the network (3)

Network Utility

Info | Netstat | Ping | Lookup | **Traceroute** | Whois | Finger | Port Scan

Enter the network address to trace an internet route to.

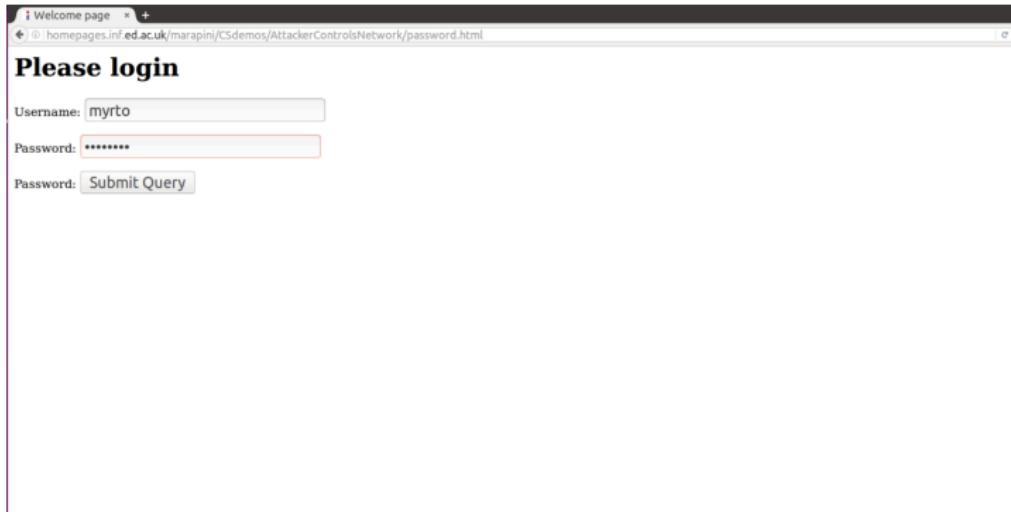
(ex. 10.0.2.1 or www.example.com)

Trace

```
Traceroute has started.

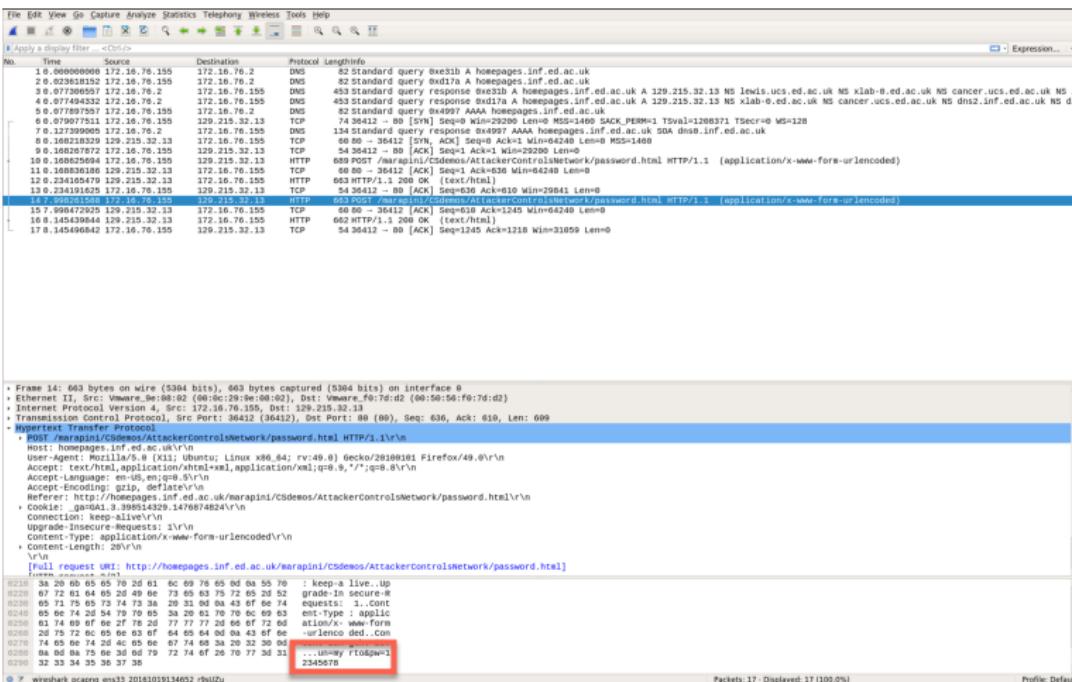
traceroute to star-mini.c10r.facebook.com (157.240.0.35), 64 hops max, 72 byte packets
 1 knussen (129.215.91.246) 0.435 ms 0.211 ns 0.185 ms
 2 vlan160.kb9-msfc.net.ed.ac.uk (129.215.160.254) 0.502 ms 0.435 ms 0.467 ms
 3 vlan688.s-pop.eastman.ja.net (194.81.57.211) 0.886 ms 0.824 ms 0.876 ms
 4 ae2.leedaaq-sbr1.ja.net (146.97.41.33) 4.578 ms 4.550 ms 4.574 ms
 5 ae30.nanchk-sbr1.ja.net (146.97.7.45) 7.365 ms 7.121 ms 7.133 ms
 6 port-channel205.car1.manchester.level3.net (199.50.119.97) 21.449 ms 10.438 ms 205.486 ms
 7 4.15.154.86 (4.15.154.86) 111.201 ms 111.202 ms 111.284 ms
 8 po103.psw01c.mia1.tfbnw.net (157.240.32.223) 110.868 ms 110.802 ms 110.810 ms
 9 157.240.36.71 (157.240.36.71) 110.913 ms 110.802 ms 110.745 ms
10 edge-star-mini-shv-02-mia1.facebook.com (157.240.0.35) 112.594 ms 112.943 ms 112.861 ms
```

All messages can be intercepted by an attacker (1)

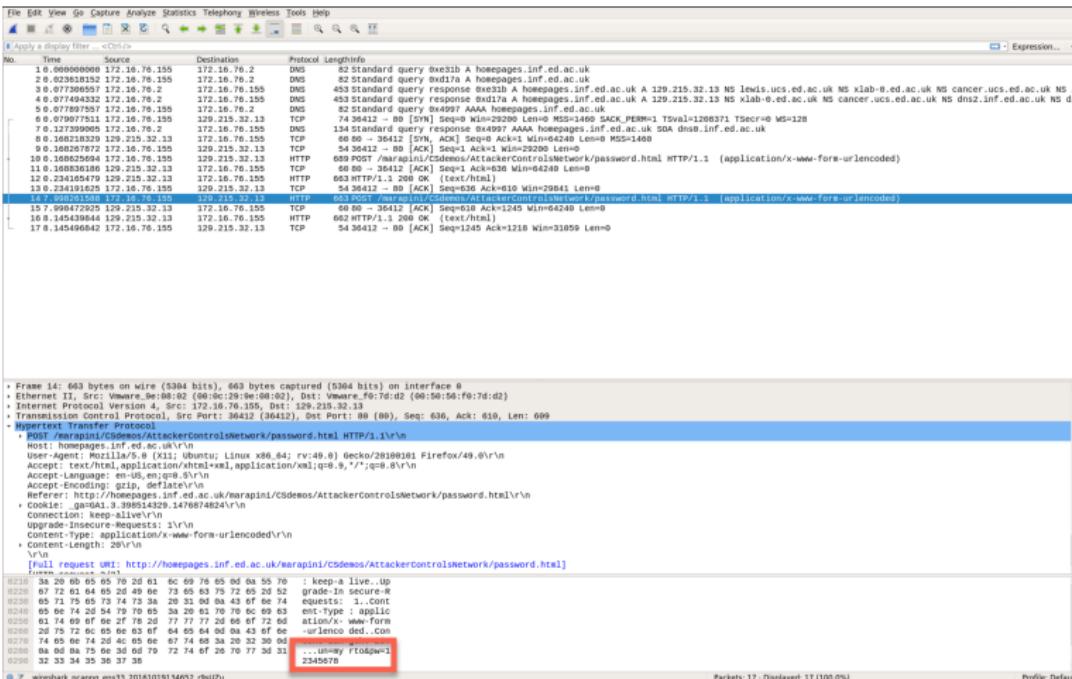


A screenshot of a web browser window titled "Welcome page". The address bar shows the URL "homepages.inf.ed.ac.uk/marapini/Csdemos/AttackerControlsNetwork/password.html". The main content area displays a "Please login" form. It has two text input fields: "Username: myrto" and "Password: *****". Below the password field is a button labeled "Submit Query".

All messages can be intercepted by an attacker (2)



All messages can be intercepted by an attacker (2)



An attacker can intercept packets, but also alter, forge new, and inject packets

More complex systems needed...

More complex systems needed...



$$\frac{e = E(K_E, \text{Transfer 100 € on Amazon's account})}{m = MAC(K_M, E(K_E, \text{Transfer 100 € on Amazon's account}))} \rightarrow$$



More complex systems needed...



$$\frac{e = E(K_E, \text{Transfer 100 € on Amazon's account})}{m = MAC(K_M, E(K_E, \text{Transfer 100 € on Amazon's account}))} \rightarrow$$



Replay attack



... to achieve more complex properties

- ▶ **Confidentiality:** Some information should never be revealed to unauthorised entities.
- ▶ **Integrity:** Data should not be altered in an unauthorised manner since the time it was created, transmitted or stored by an authorised source.
- ▶ **Authentication:** Ability to know with certainty the identity of an communicating entity.
- ▶ **Anonymity:** The identity of the author of an action (e.g. sending a message) should not be revealed.
- ▶ **Unlinkability:** An attacker should not be able to deduce whether different services are delivered to the same user
- ▶ **Non-repudiation:** The author of an action should not be able to deny having triggered this action.
- ▶ ...

Cryptographic protocols

Cryptographic protocols

Programs relying on [cryptographic primitives](#) and whose goal is the establishment of “secure” communications.

Cryptographic protocols

Cryptographic protocols

Programs relying on [cryptographic primitives](#) and whose goal is the establishment of “secure” communications.

But!

Many exploitable errors are due not to design errors in the primitives, but to the way they are used, *i.e.* bad protocol design and buggy or not careful enough implementation

Numerous deployed protocols are flawed...

... and end up in the news :(

ZDNet EDITION: UK Q MICROSOFT STORAGE INNOVATION HARDWARE APPLE MORE > NEWSLETTERS ALL WRITERS

FREAK: Another day, another serious SSL security hole

More than one third of encrypted Websites are open to attack via the FREAK security hole.

The Telegraph

Home Video News World Sport Business Money Comment Culture Travel Life W USA Asia China Europe Middle East Australasia Africa South America Central Asia

HOME > NEWS > WORLD NEWS > NORTH AMERICA > USA

Hacker remotely crashes Jeep from 10 miles away

Security experts warn that more than 470,000 cars made by Fiat Chrysler could be at risk of being attacked by similar means – including those driven in the UK

The Register®
Biting the hand that feeds IT

Defects in e-passports allow real-time tracking

This threat brought to you by RFID

threat **post**

CATEGORIES FEATURED PODCASTS VIDEOS

TRIPLE HANDSHAKE ATTACKS TARGET TLS RESUMPTION, RENEGOTIATION

Logical attacks

Many of these attacks do not even break the crypto primitives!!

Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: stream ciphers

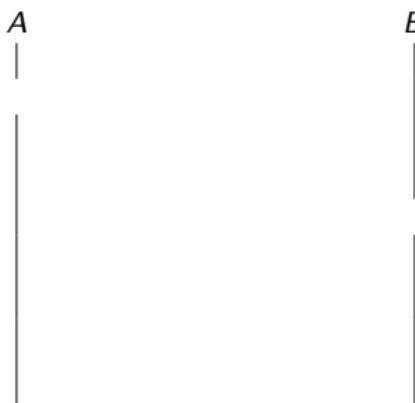
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: stream ciphers



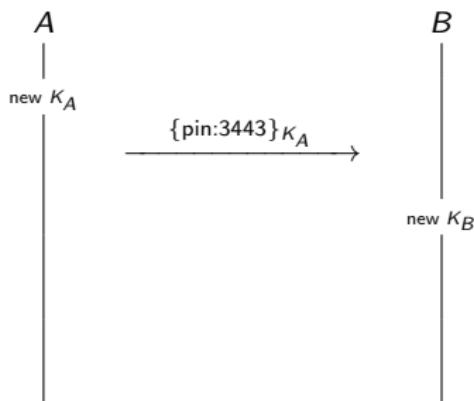
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: stream ciphers



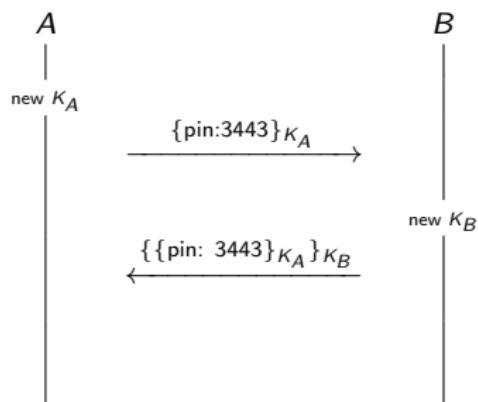
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: stream ciphers



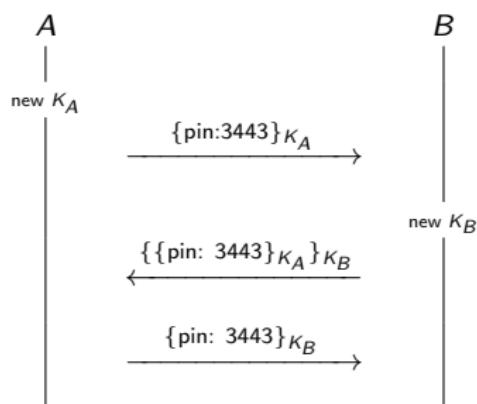
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: stream ciphers



since $\{\{\text{pin: 3443}\}_{K_A}\}_{K_B} = \{\{\text{pin: 3443}\}_{K_B}\}_{K_A}$ by commutativity

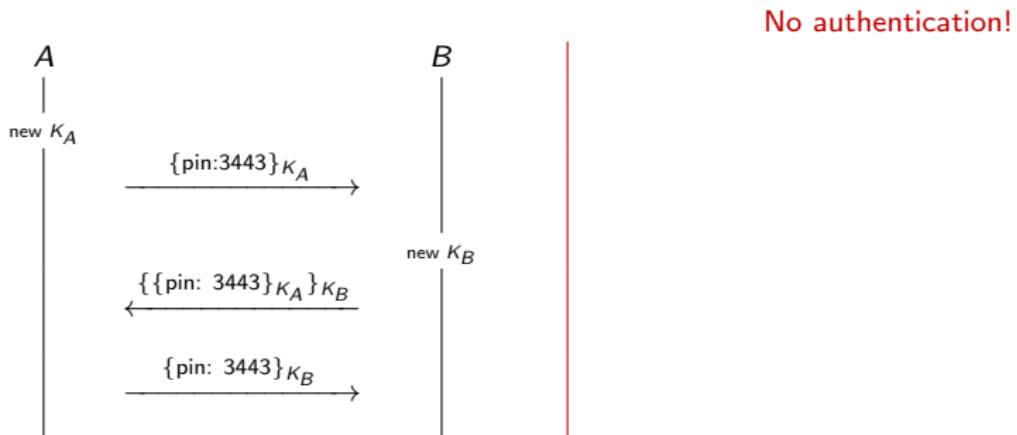
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: stream ciphers



since $\{\{\text{pin: 3443}\}_{K_A}\}_{K_B} = \{\{\text{pin: 3443}\}_{K_B}\}_{K_A}$ by commutativity

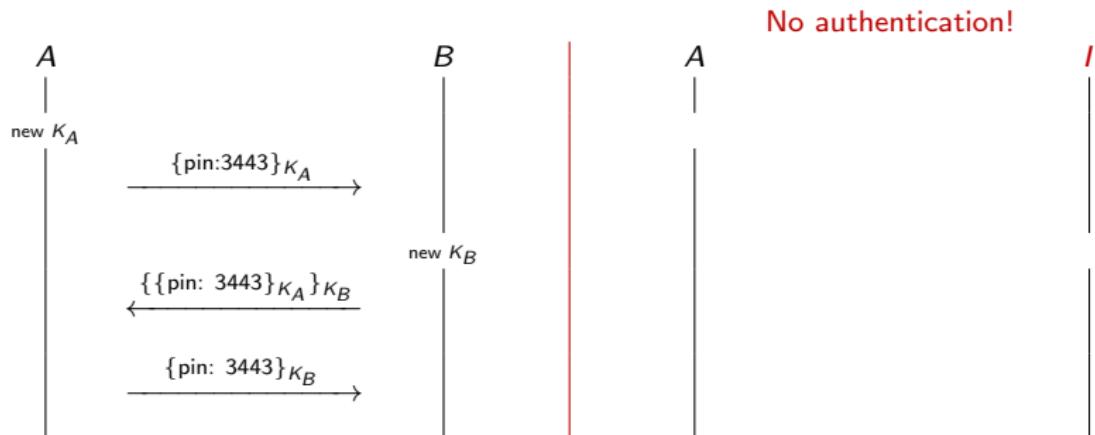
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: stream ciphers



since $\{\{\text{pin: 3443}\}_{K_A}\}_{K_B} = \{\{\text{pin: 3443}\}_{K_B}\}_{K_A}$ by commutativity

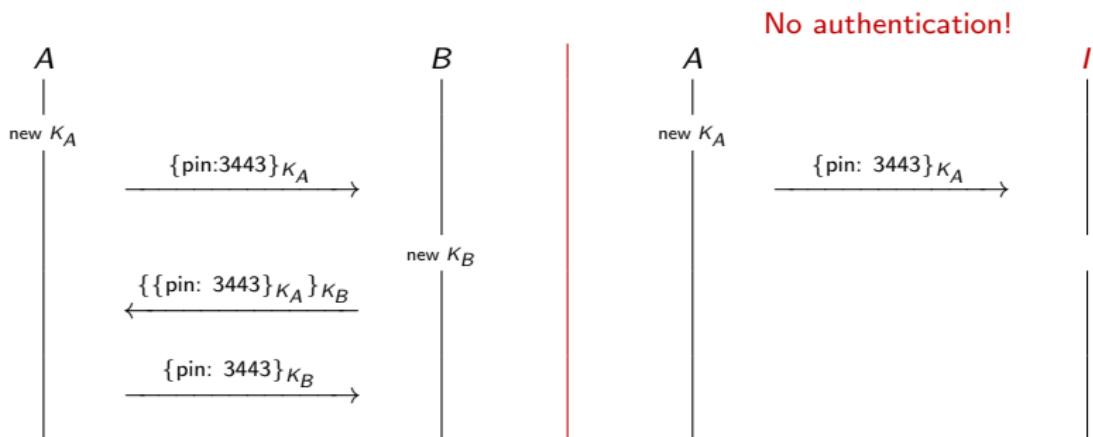
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: stream ciphers



since $\{\{\text{pin: } 3443\}_{K_A}\}_{K_B} = \{\{\text{pin: } 3443\}_{K_B}\}_{K_A}$ by commutativity

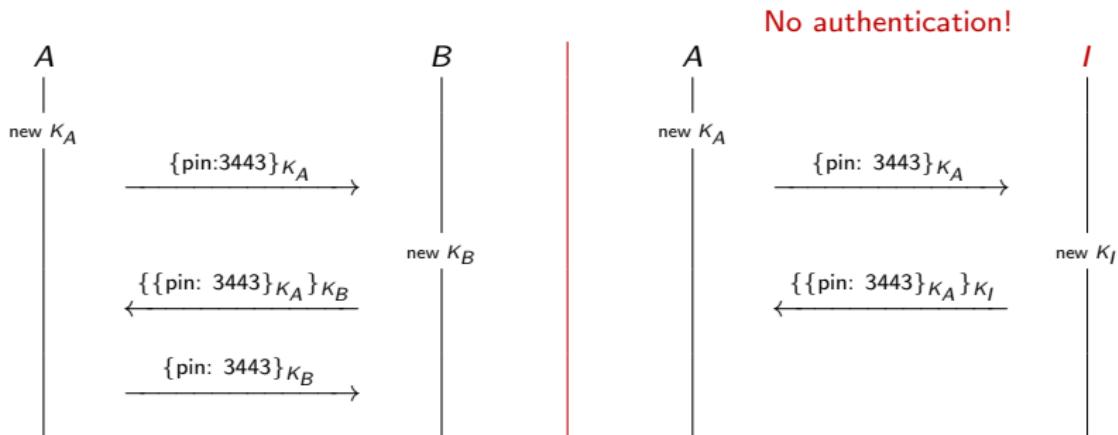
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: stream ciphers



since $\{\{\text{pin: 3443}\}_{K_A}\}_{K_B} = \{\{\text{pin: 3443}\}_{K_B}\}_{K_A}$ by commutativity

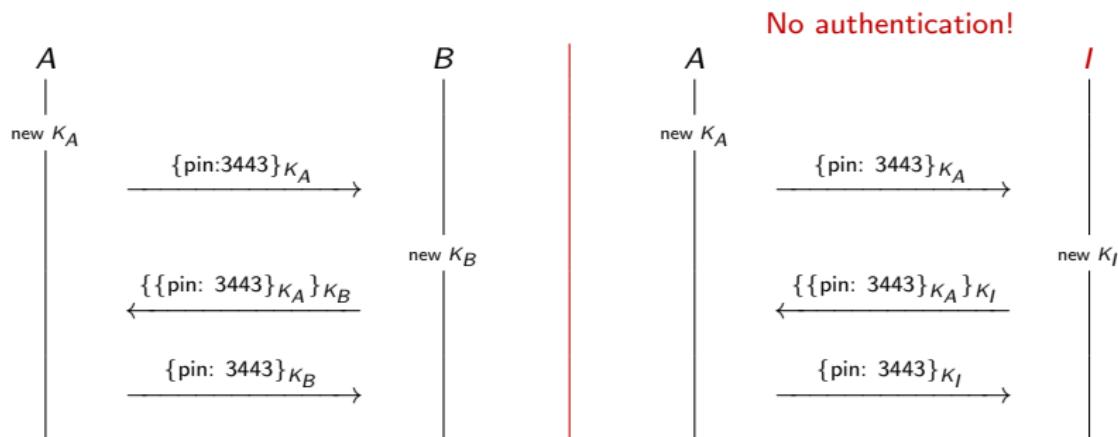
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: stream ciphers



since $\{\{\text{pin: 3443}\}_{K_A}\}_{K_B} = \{\{\text{pin: 3443}\}_{K_B}\}_{K_A}$ by commutativity

Authentication and key agreement protocols

Authentication and key agreement

- ▶ Long-term keys should be used as little as possible to reduce “attack-surface”
- ▶ The use of a key should be restricted to a specific purpose
 - e.g. you shouldn't use the same RSA key both for encryption and signing
- ▶ Public key algorithms tend to be computationally more expensive than symmetric key algorithms
- ~~> Long-term keys are used to establish short-term session keys
 - e.g. TLS over HTTP, AKA for 3G, BAC for epassports, etc.

Needham-Schroeder Public Key (NSPK)

NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

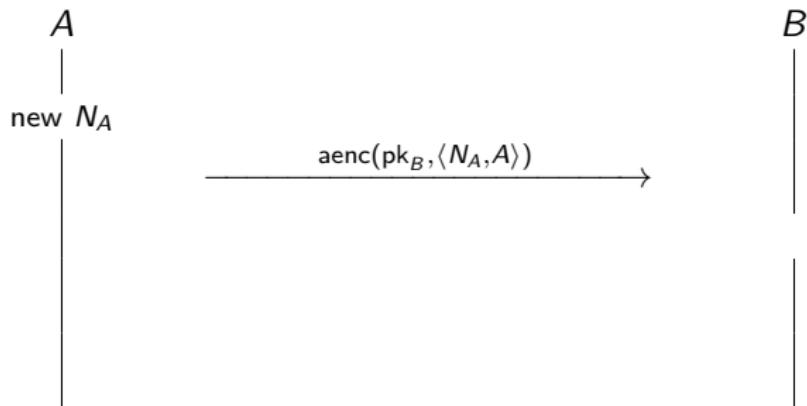
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

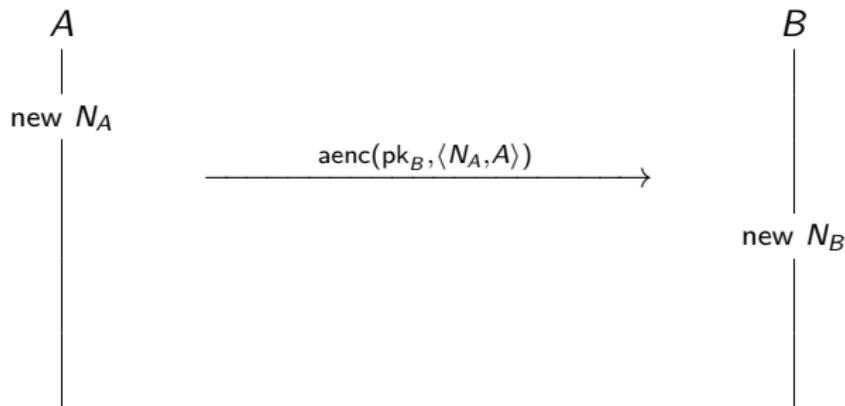
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

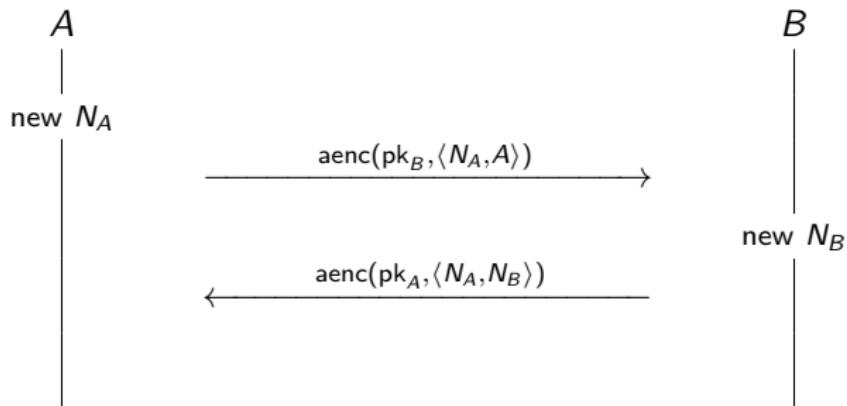
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

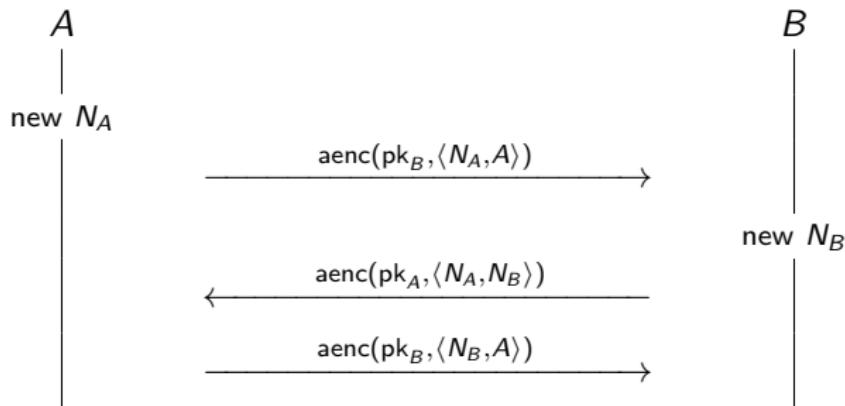
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

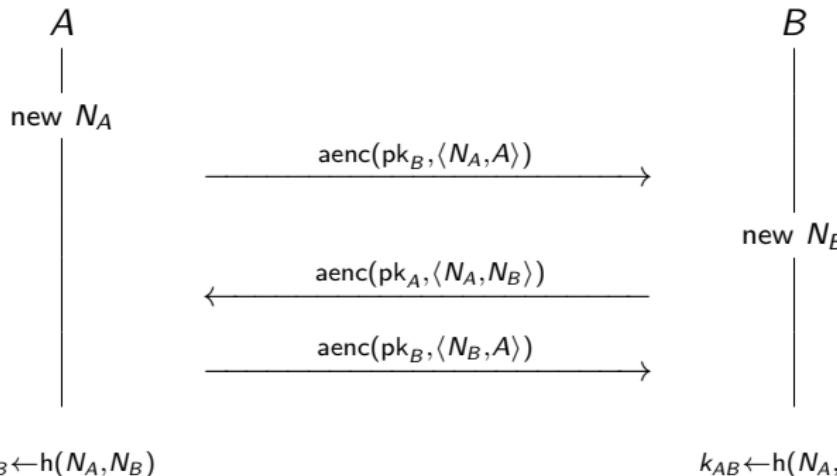
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

NSPK: security requirements

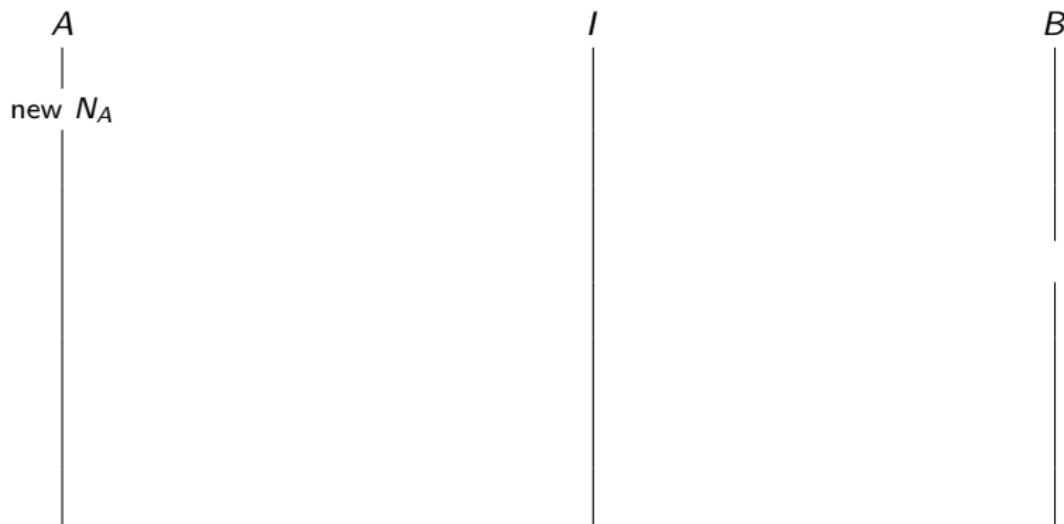
- ▶ **Authentication:** if Alice has completed the protocol, apparently with Bob, then Bob must also have completed the protocol with Alice.
- ▶ **Authentication:** If Bob has completed the protocol, apparently with Alice, then Alice must have completed the protocol with Bob.
- ▶ **Confidentiality:** Messages sent encrypted with the agreed key ($k \leftarrow h(N_A, N_B)$) remain secret.

NSPK: Lowe's attack on authentication

Attack found 17 years after the publication of the NS protocol!!

NSPK: Lowe's attack on authentication

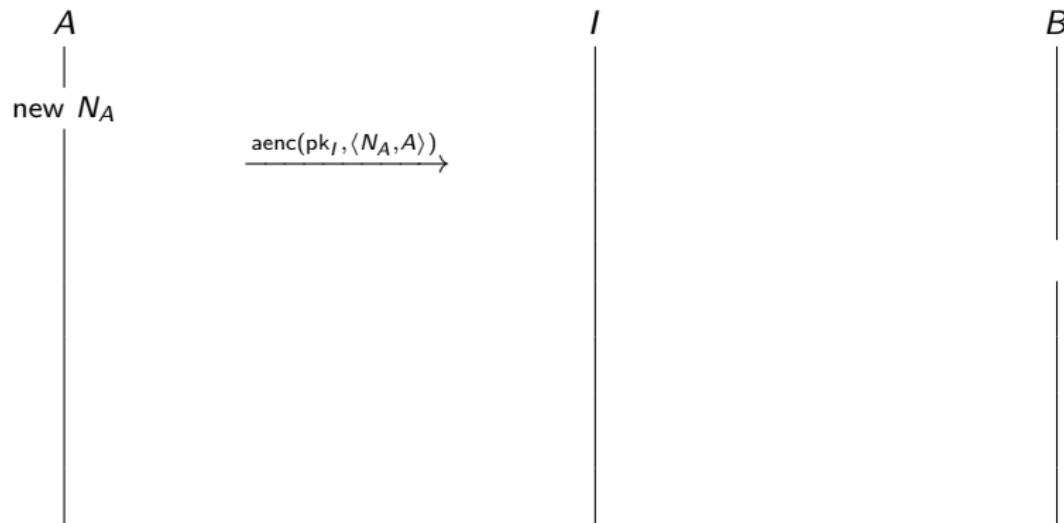
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

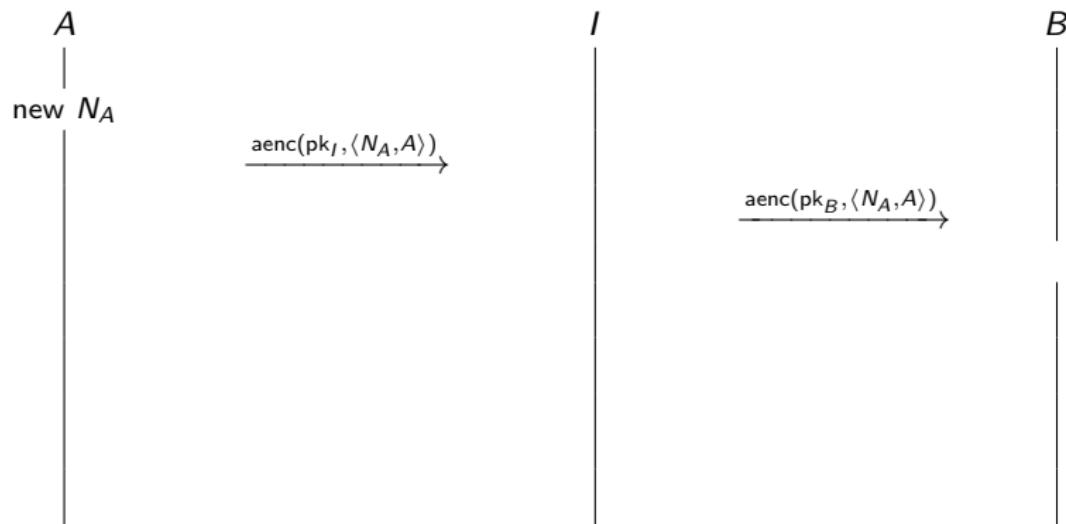
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

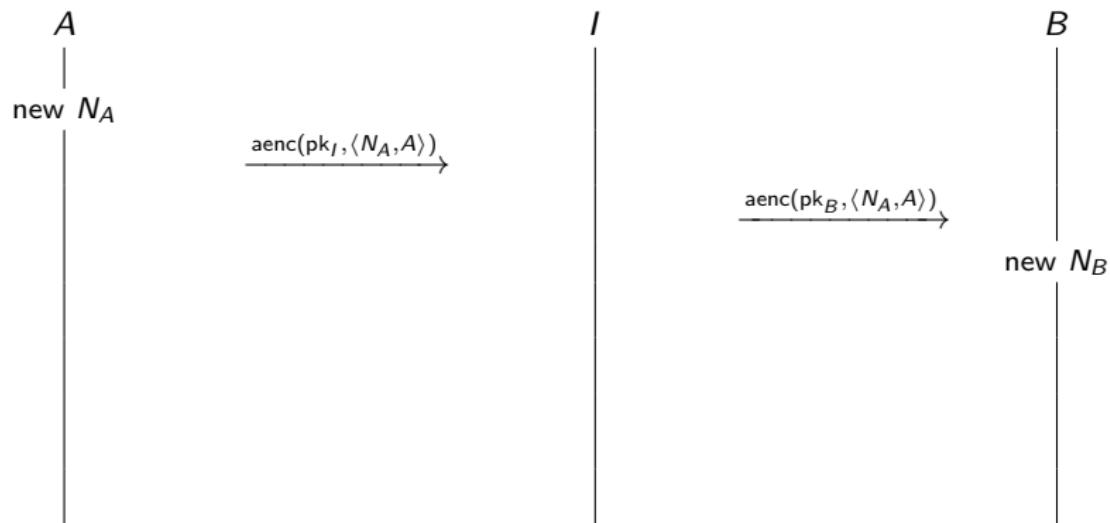
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

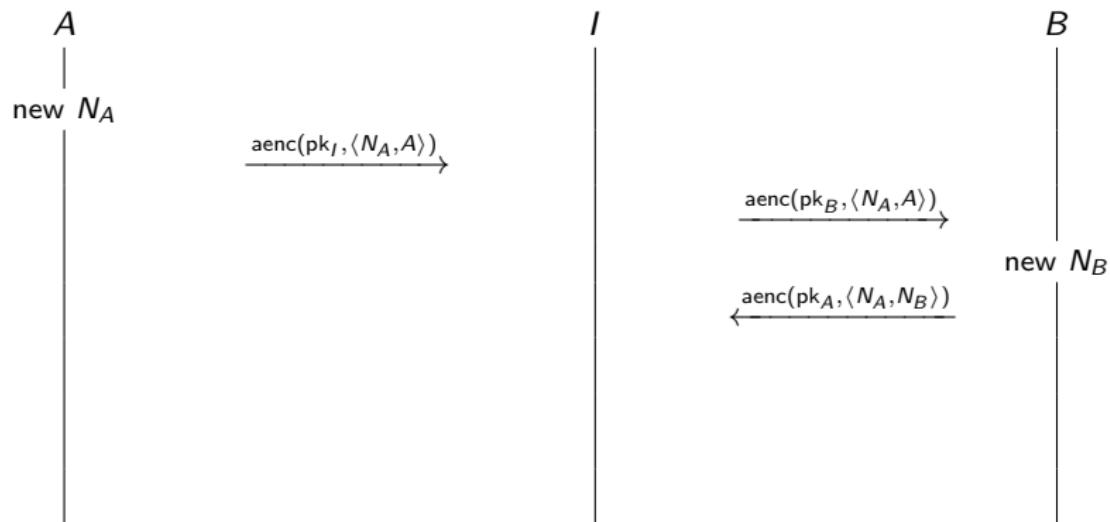
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

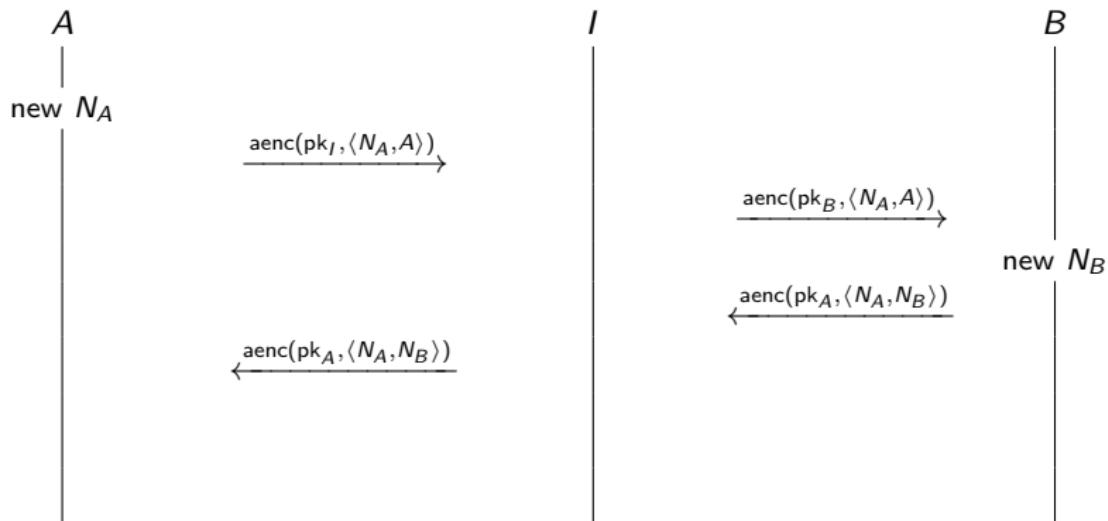
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

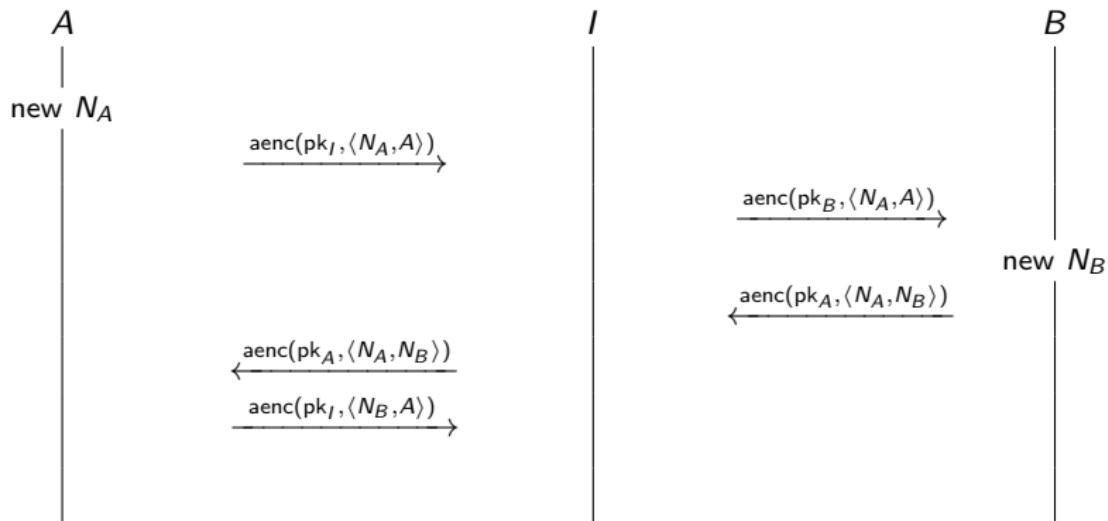
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

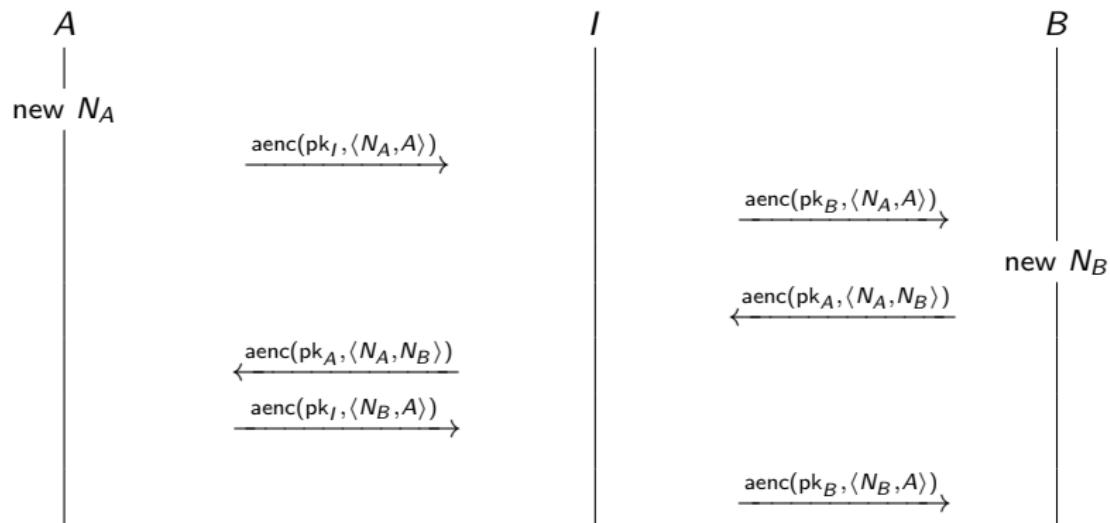
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

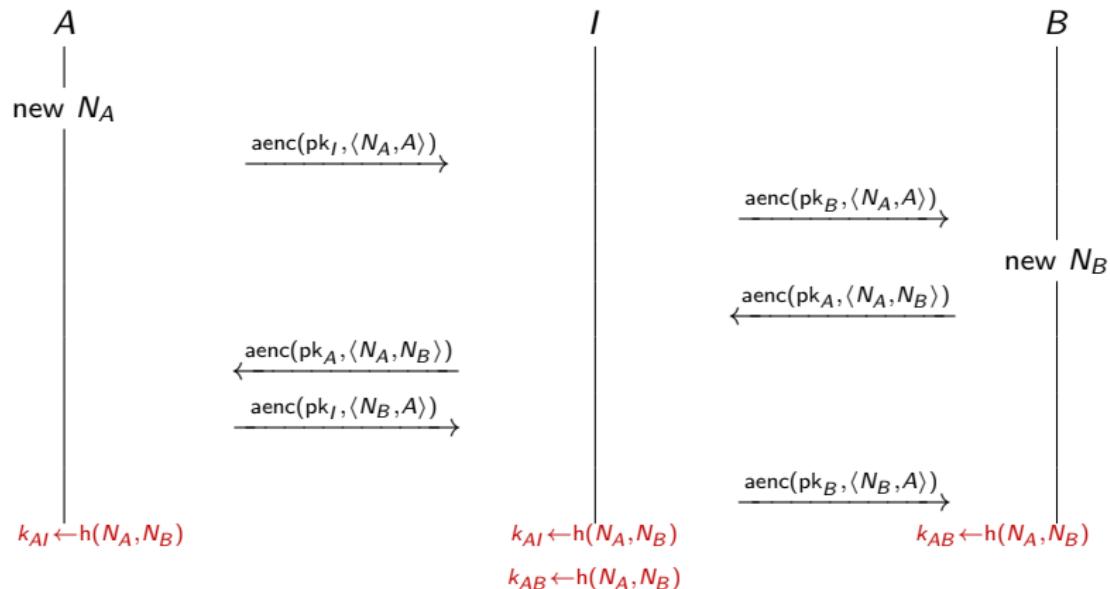
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

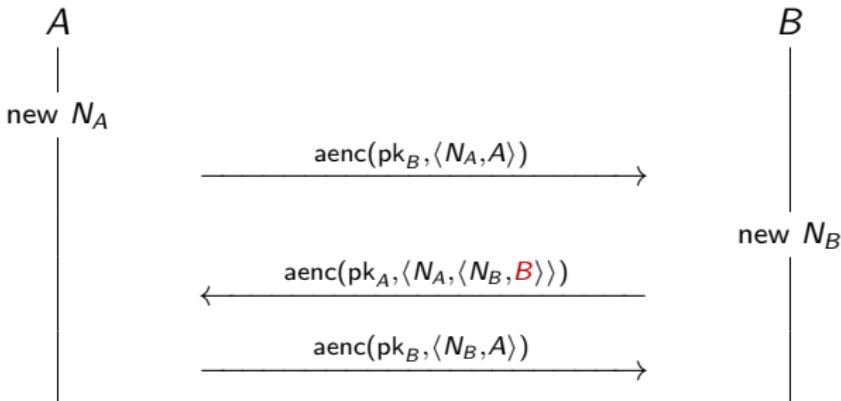
NSPK: Lowe's attack on authentication

Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's fix



$$k_{AB} \leftarrow h(N_A, N_B)$$

$$k_{AB} \leftarrow h(N_A, N_B)$$

Forward secrecy

- ▶ The NSL protocol is secure against an attacker that controls the network.
- ▶ What if Alice's and Bob's private keys get compromised?
- ▶ What if the government forces Alice and Bob to reveal their private keys?
- ▶ Can we still protect confidentiality?

Forward secrecy

A protocol ensures **forward secrecy**, if even if long-term keys are compromised, past sessions of the protocol are still kept confidential, and this even if an attacker actively interferred.

The Station-to-Station (StS) protocol



The Station-to-Station (StS) protocol



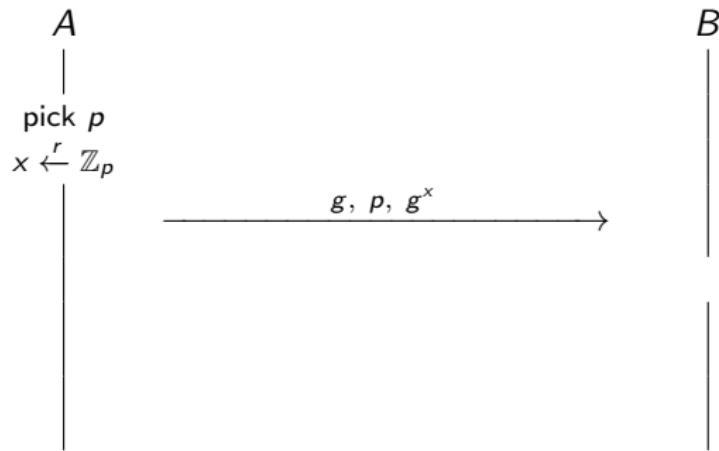
- ▶ where p is a large prime
- ▶ and g a generator of \mathbb{Z}_p^*

The Station-to-Station (StS) protocol



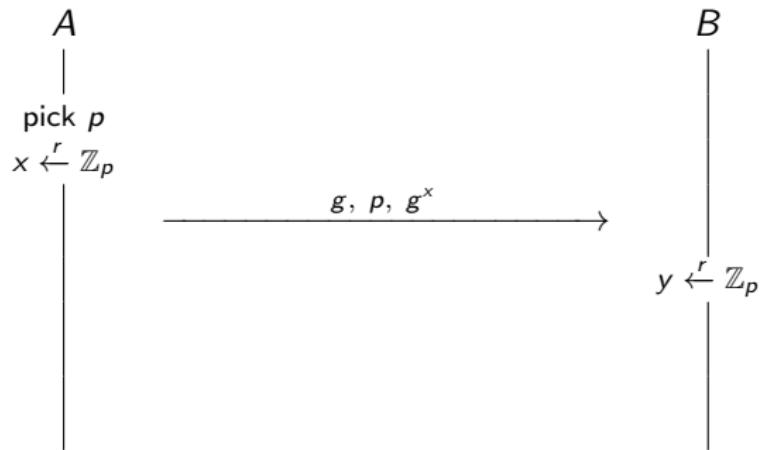
- ▶ where p is a large prime
- ▶ and g a generator of \mathbb{Z}_p^*

The Station-to-Station (StS) protocol



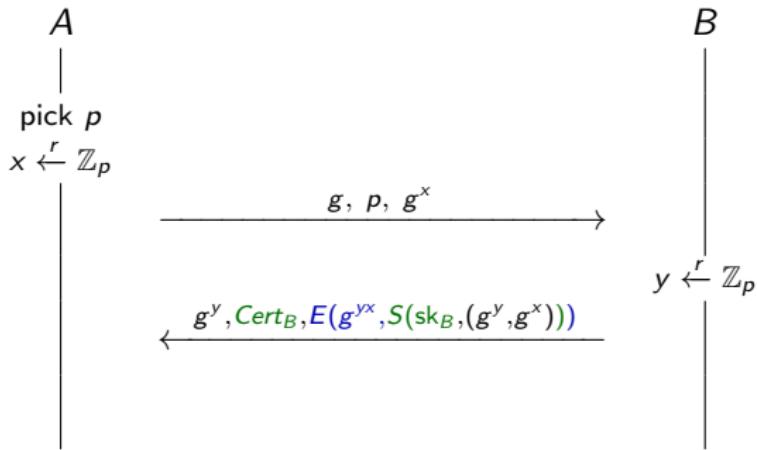
- ▶ where p is a large prime
- ▶ and g a generator of \mathbb{Z}_p^*

The Station-to-Station (StS) protocol



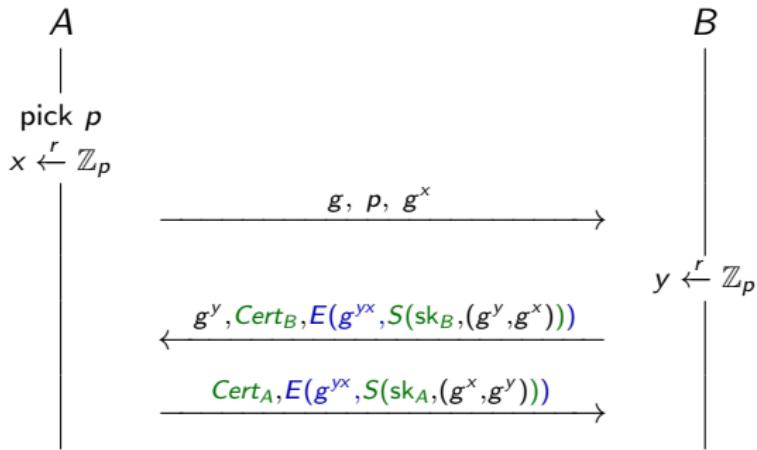
- ▶ where p is a large prime
- ▶ and g a generator of \mathbb{Z}_p^*

The Station-to-Station (StS) protocol



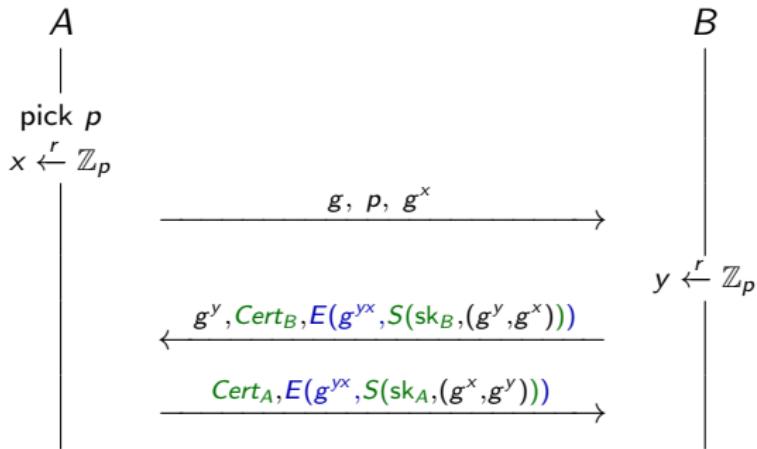
- ▶ where p is a large prime
- ▶ and g a generator of \mathbb{Z}_p^*

The Station-to-Station (StS) protocol



- ▶ where p is a large prime
- ▶ and g a generator of \mathbb{Z}_p^*

The Station-to-Station (StS) protocol



- ▶ where p is a large prime
- ▶ and g a generator of \mathbb{Z}_p^*

The StS ensures mutual authentication, key agreement, and forward secrecy

(More) Cryptographic protocols

Myrto Arapinis
School of Informatics
University of Edinburgh

February 25, 2019

Authentication and key agreement protocols

Authentication and key agreement

- ▶ Long-term keys should be used as little as possible to reduce “attack-srufarce”
- ▶ The use of a key should be restricted to a specific purpose
 - e.g. you shouldn't use the same RSA key both for encryption and signing
- ▶ Public key algorithms tend to be computationally more expensive than symmetric key algorithms
- ~~ Long-term keys are used to establish short-term session keys
 - e.g. TLS over HTTP, AKA for 3G, BAC for epassports, etc.

Needham-Schroeder Public Key (NSPK)

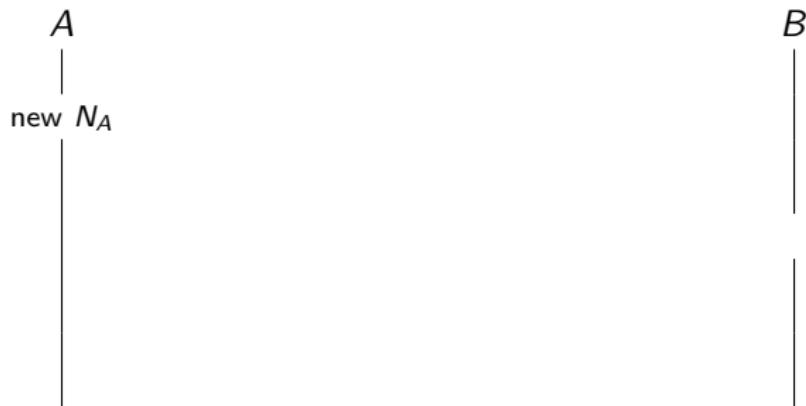
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

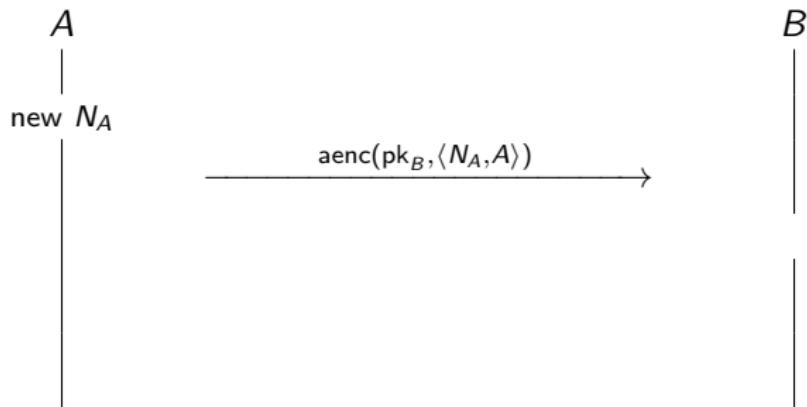
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

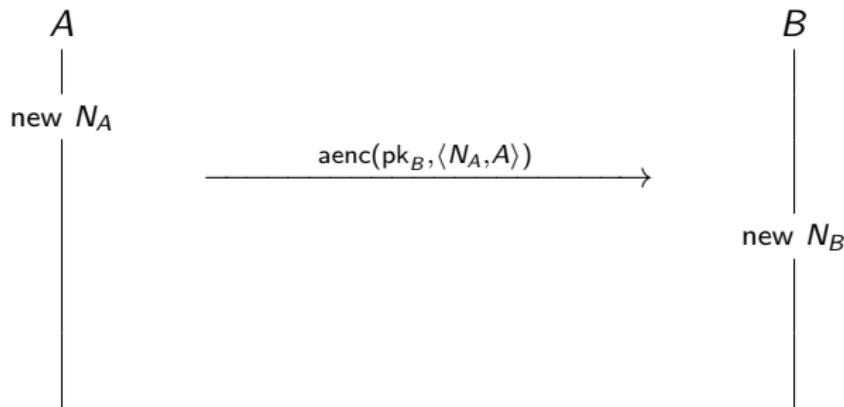
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

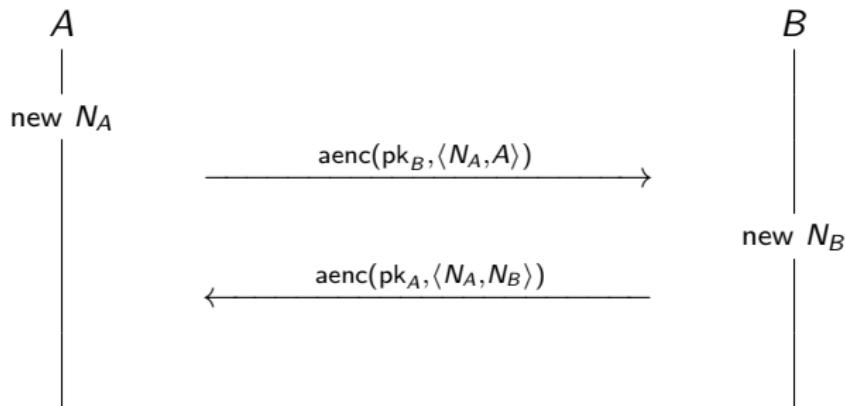
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

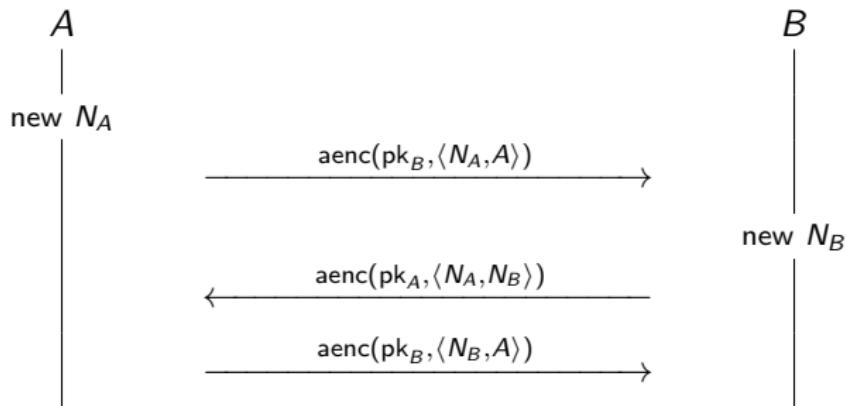
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

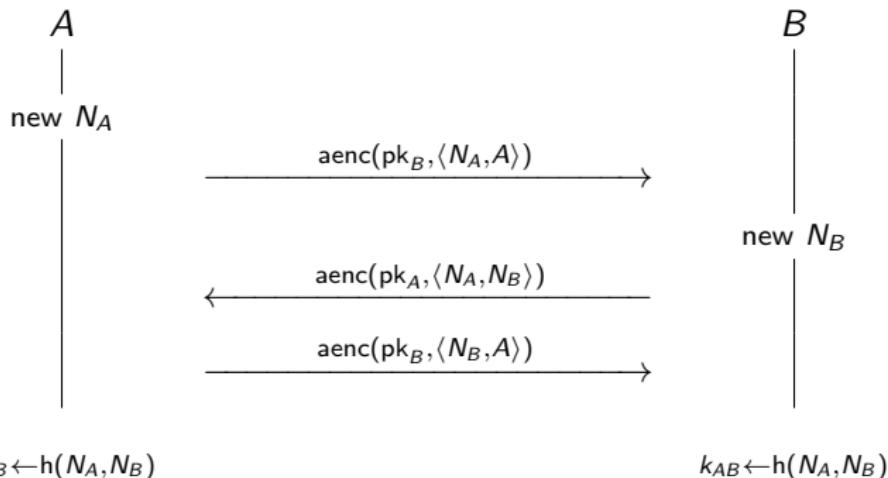
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

NSPK: security requirements

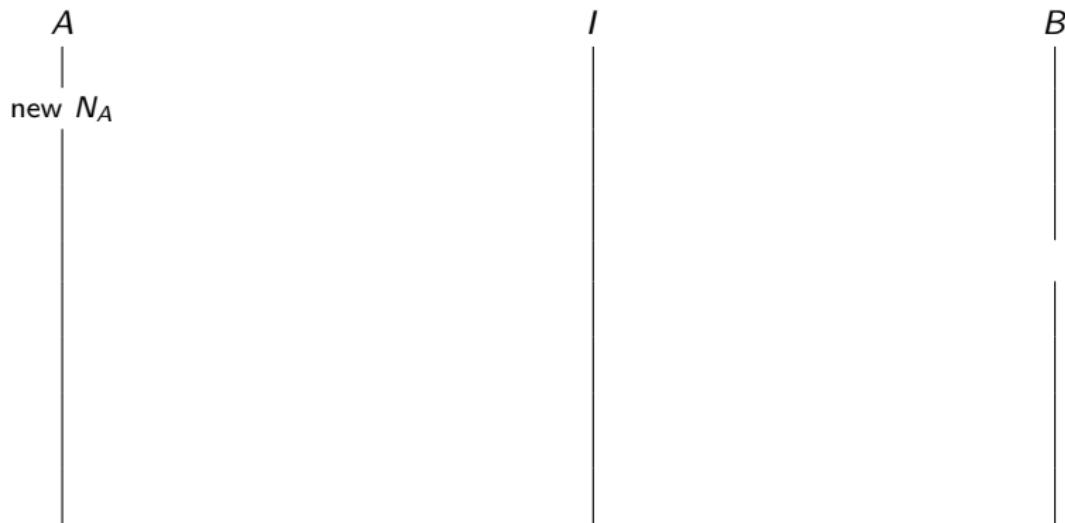
- ▶ **Authentication:** if Alice has completed the protocol, apparently with Bob, then Bob must also have completed the protocol with Alice.
- ▶ **Authentication:** If Bob has completed the protocol, apparently with Alice, then Alice must have completed the protocol with Bob.
- ▶ **Confidentiality:** Messages sent encrypted with the agreed key ($k \leftarrow h(N_A, N_B)$) remain secret.

NSPK: Lowe's attack on authentication

Attack found 17 years after the publication of the NS protocol!!

NSPK: Lowe's attack on authentication

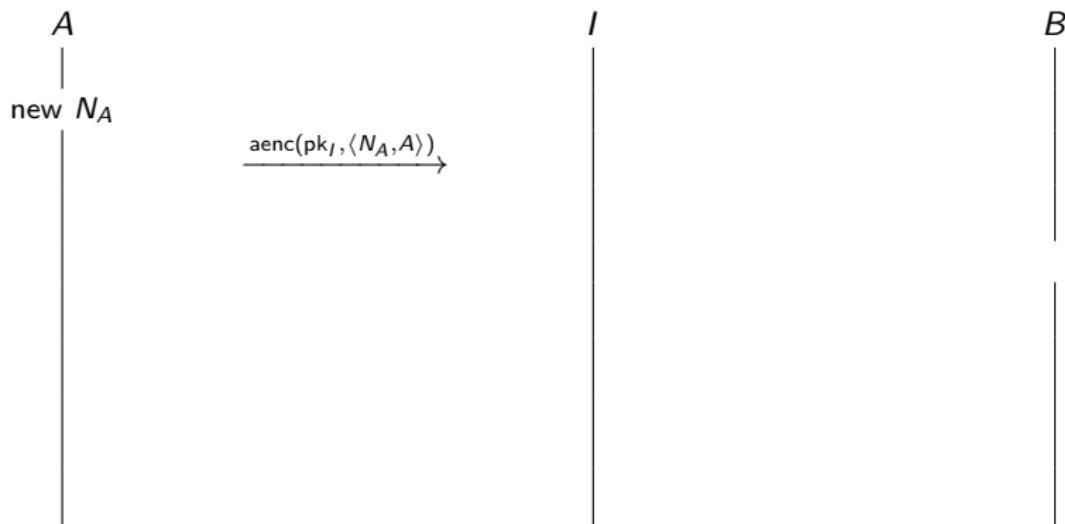
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

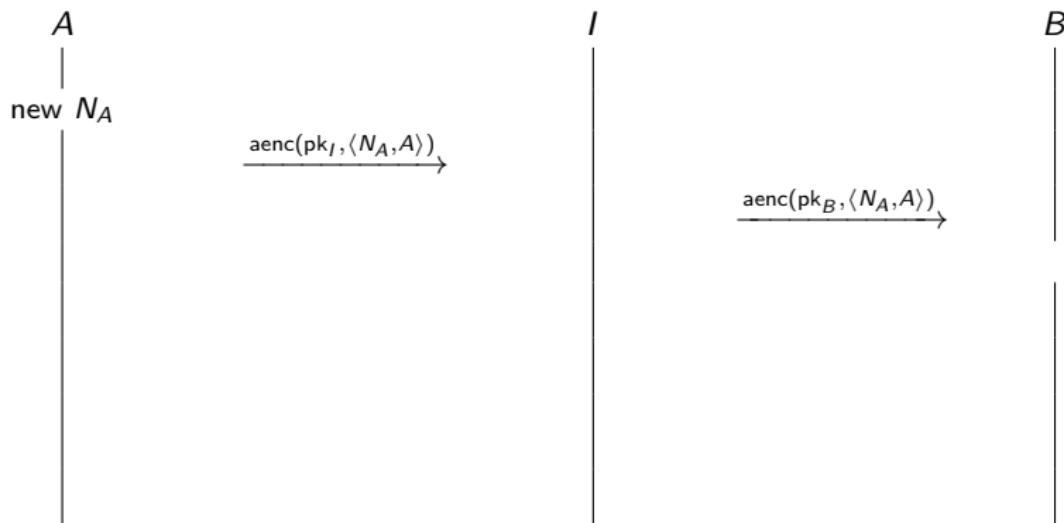
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

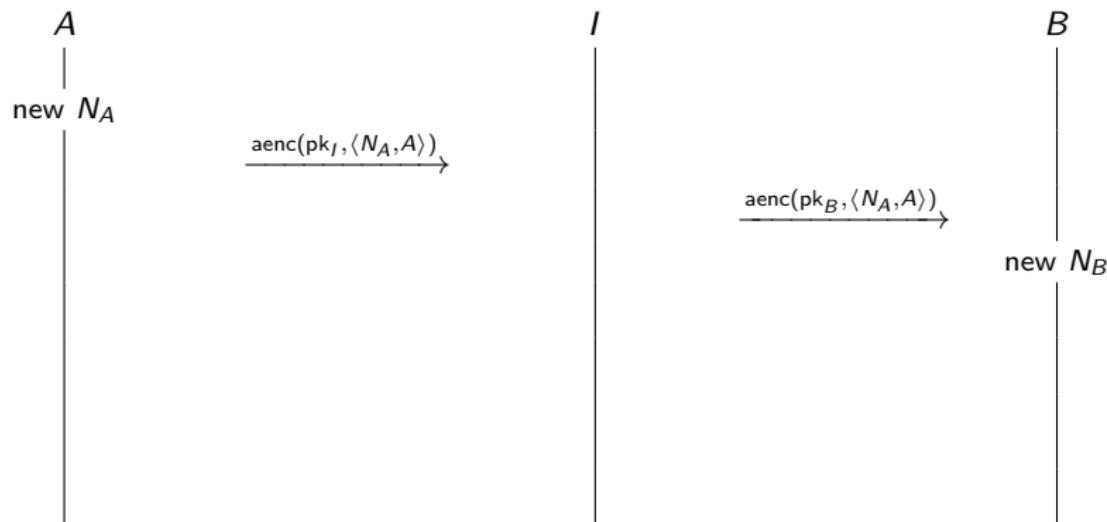
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

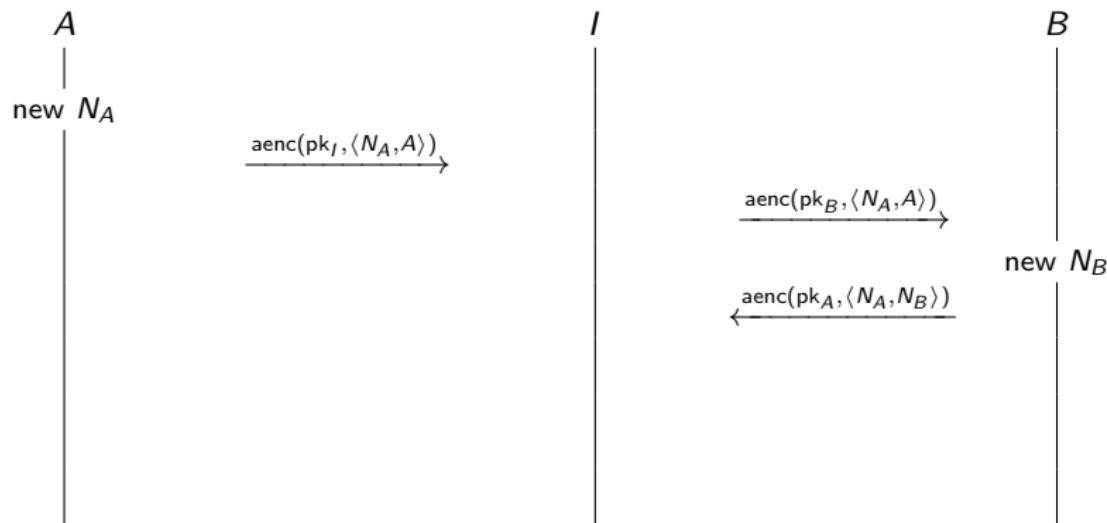
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

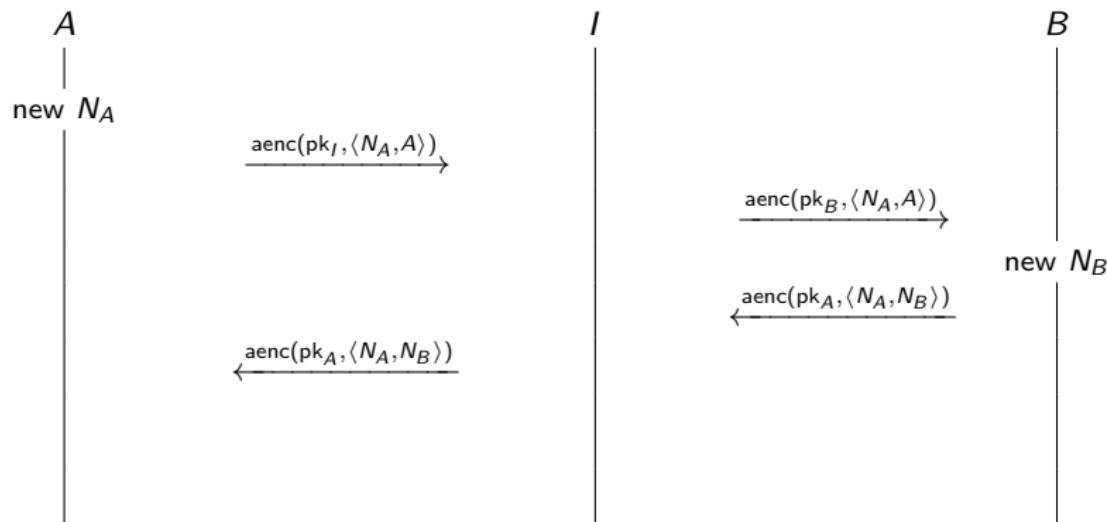
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

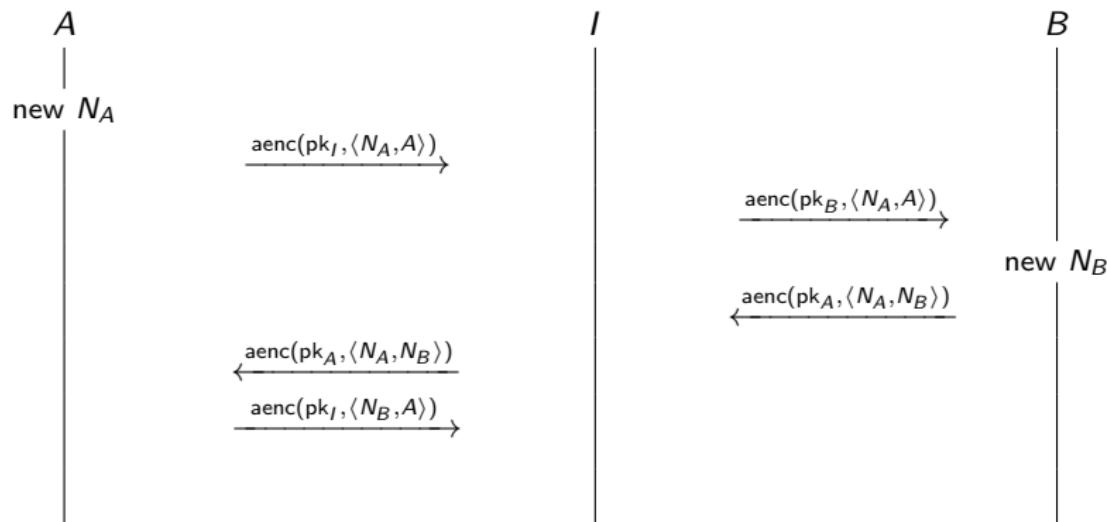
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

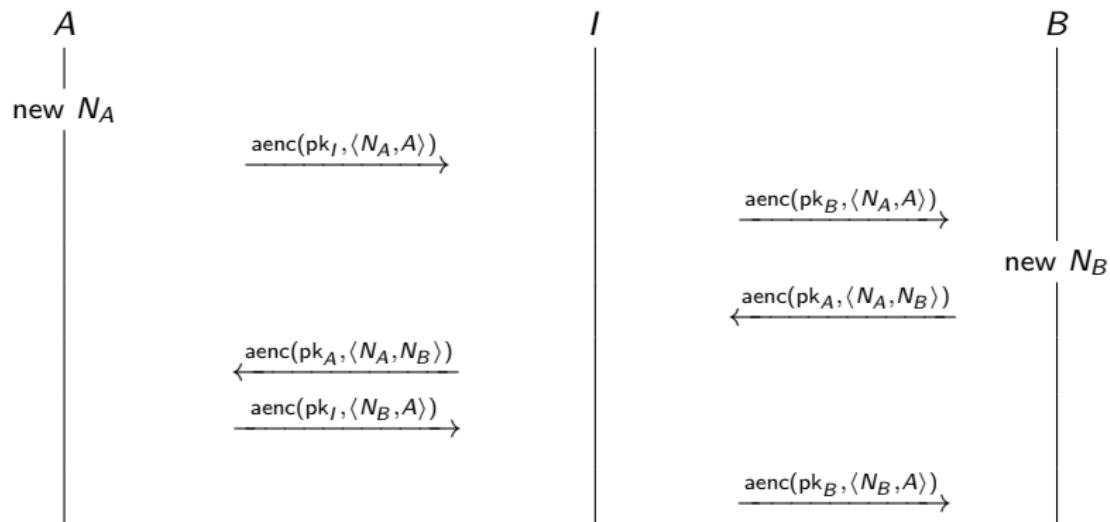
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

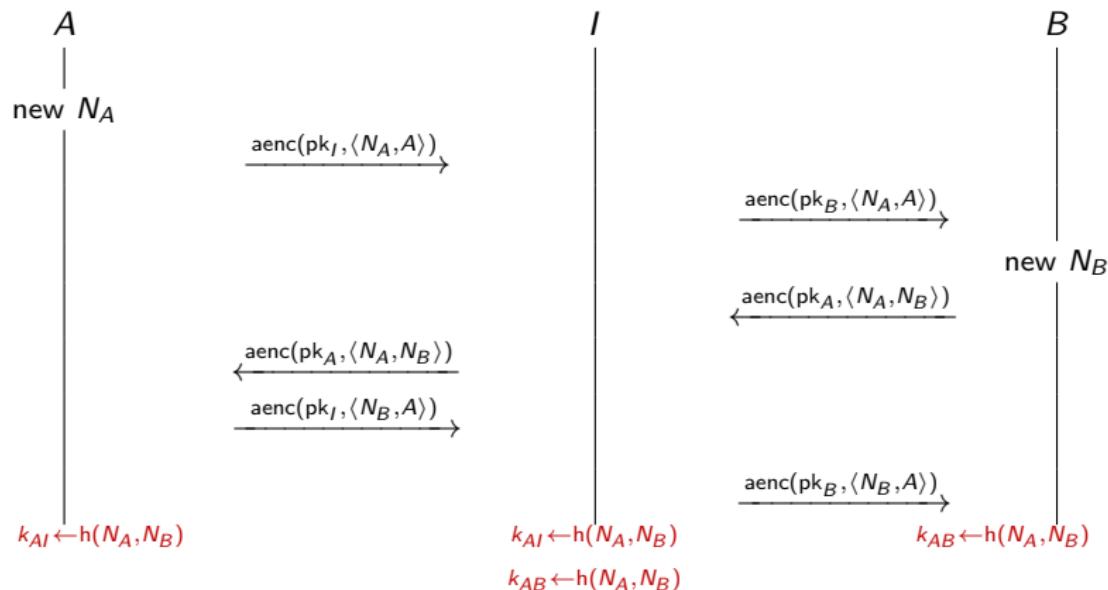
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

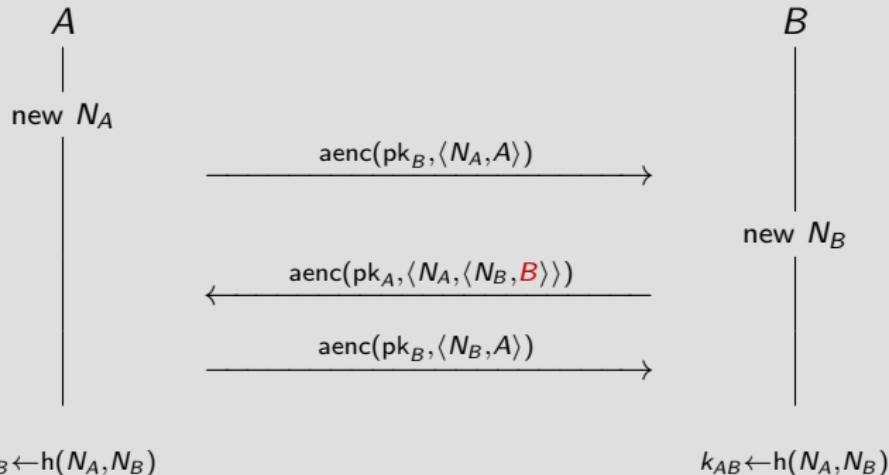
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's fix

The Needham-Schroeder-Lowe (NSL) protocol



Forward secrecy

- ▶ The NSL protocol is secure against an attacker that controls the network.
- ▶ What if the Alice's and Bob's private keys get compromised?
- ▶ What if the government forces Alice and Bob to reveal their private keys?
- ▶ Can we still protect confidentiality?

Forward secrecy

A protocol ensures **forward secrecy**, if even if long-term keys are compromised, past sessions of the protocol are still kept confidential, and this even if an attacker actively interfered.

The Station-to-Station (StS) protocol

A



B



The Station-to-Station (StS) protocol



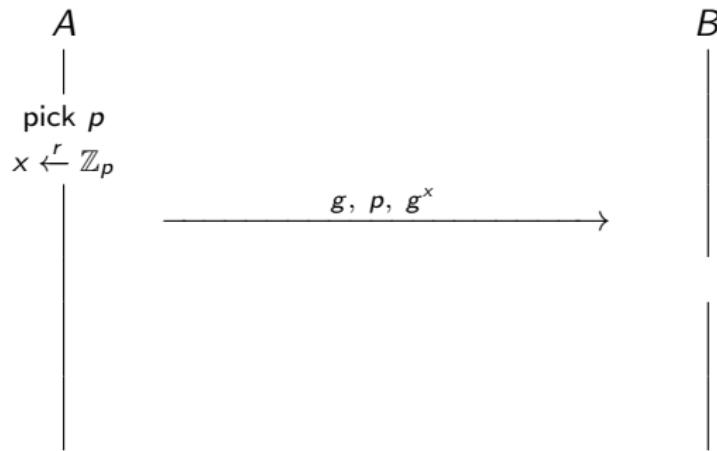
- ▶ where p is a large prime
- ▶ and g a generator of \mathbb{Z}_p^*

The Station-to-Station (StS) protocol



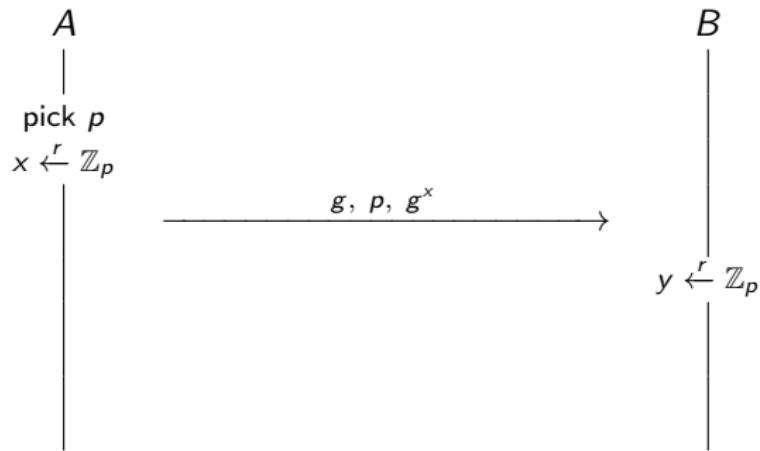
- ▶ where p is a large prime
- ▶ and g a generator of \mathbb{Z}_p^*

The Station-to-Station (StS) protocol



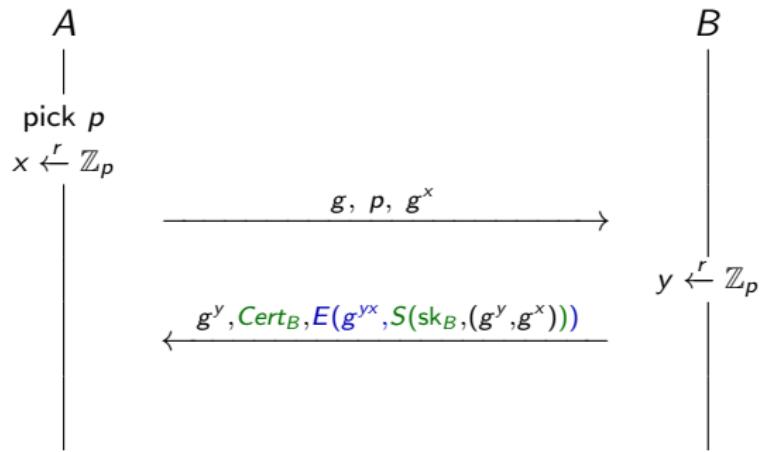
- ▶ where p is a large prime
- ▶ and g a generator of \mathbb{Z}_p^*

The Station-to-Station (StS) protocol



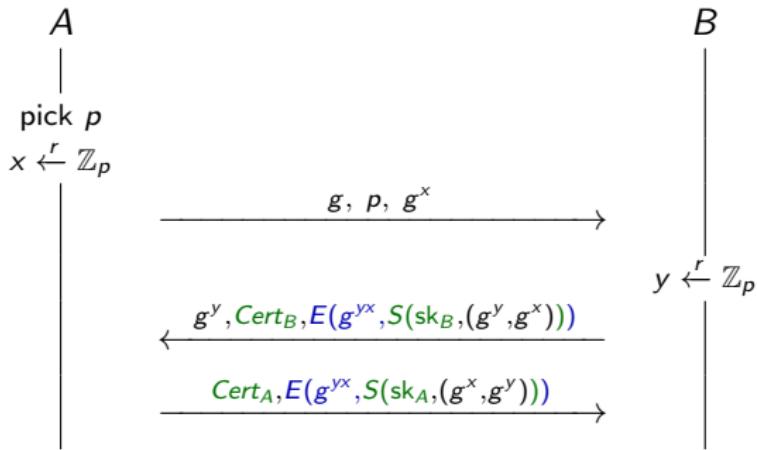
- ▶ where p is a large prime
- ▶ and g a generator of \mathbb{Z}_p^*

The Station-to-Station (StS) protocol



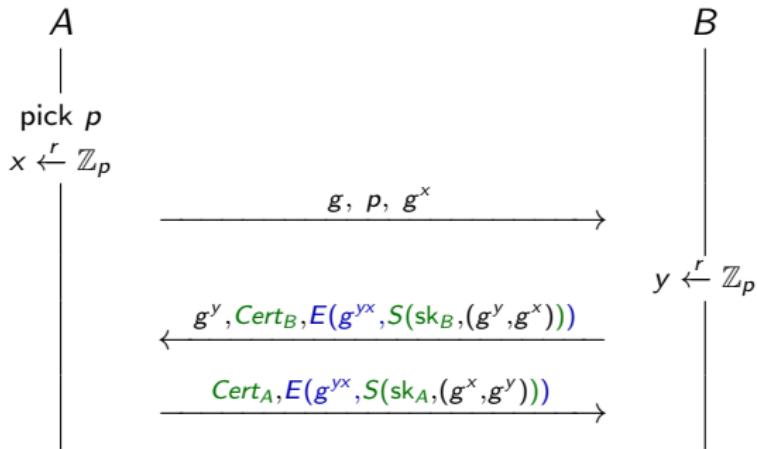
- ▶ where p is a large prime
- ▶ and g a generator of \mathbb{Z}_p^*

The Station-to-Station (StS) protocol



- ▶ where p is a large prime
- ▶ and g a generator of \mathbb{Z}_p^*

The Station-to-Station (StS) protocol



- ▶ where p is a large prime
- ▶ and g a generator of \mathbb{Z}_p^*

The StS ensures mutual authentication, key agreement, and forward secrecy

The Basic Access Control (BAC) protocol

An e-Passport is a passport with an RFID tag embedded in it.



The RFID tag stores:

- ▶ the information printed on the passport,
- ▶ a JPEG copy of the picture

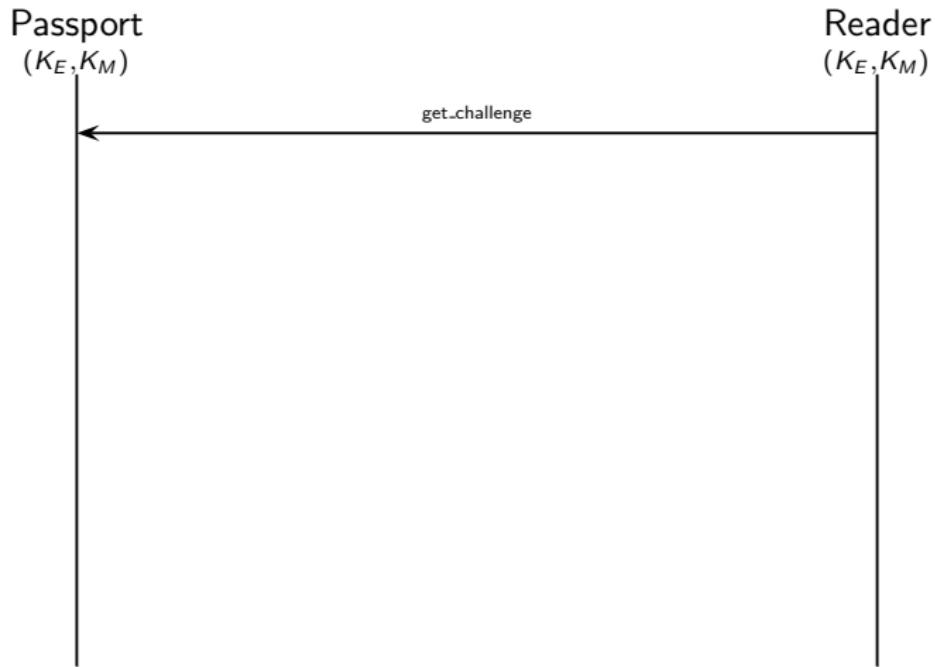
BAC: authentication and key agreement protocol implemented on e-Passports

The Basic Access Control protocol (BAC)

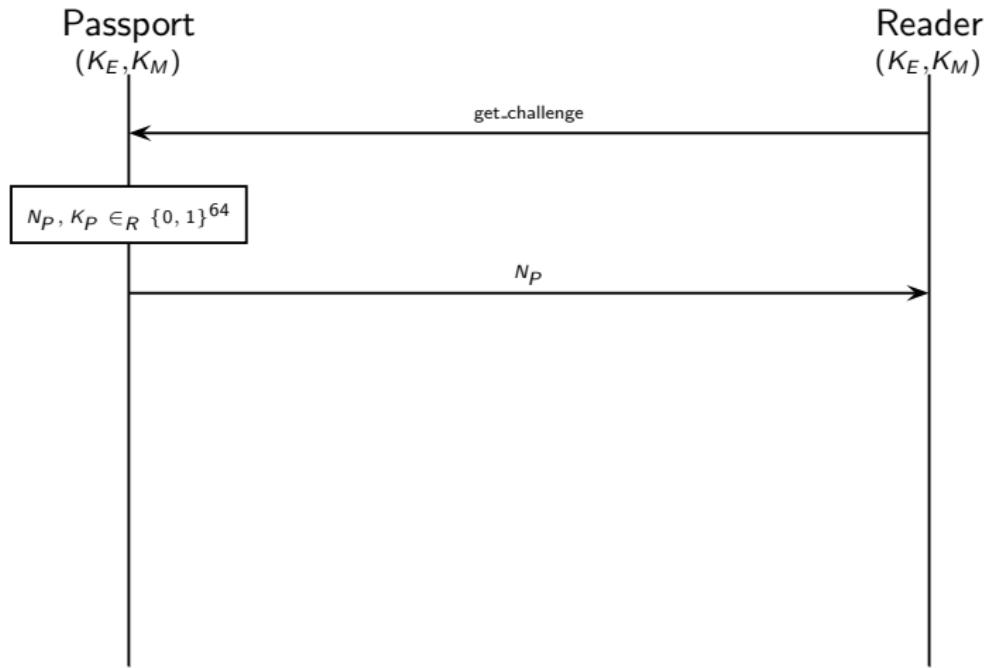
Passport
 (K_E, K_M)

Reader
 (K_E, K_M)

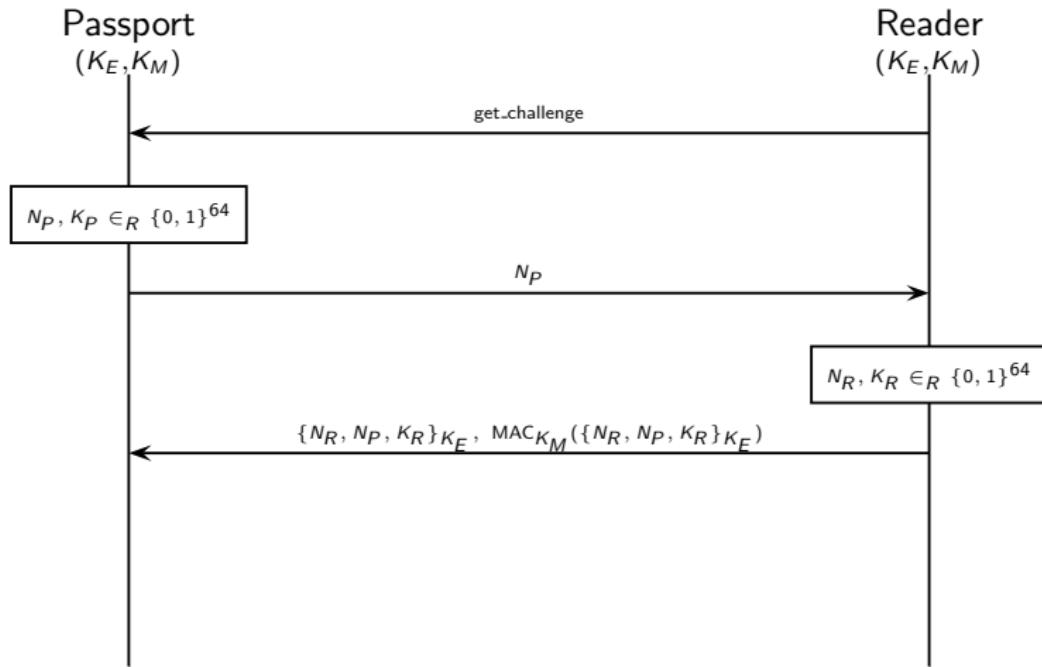
The Basic Access Control protocol (BAC)



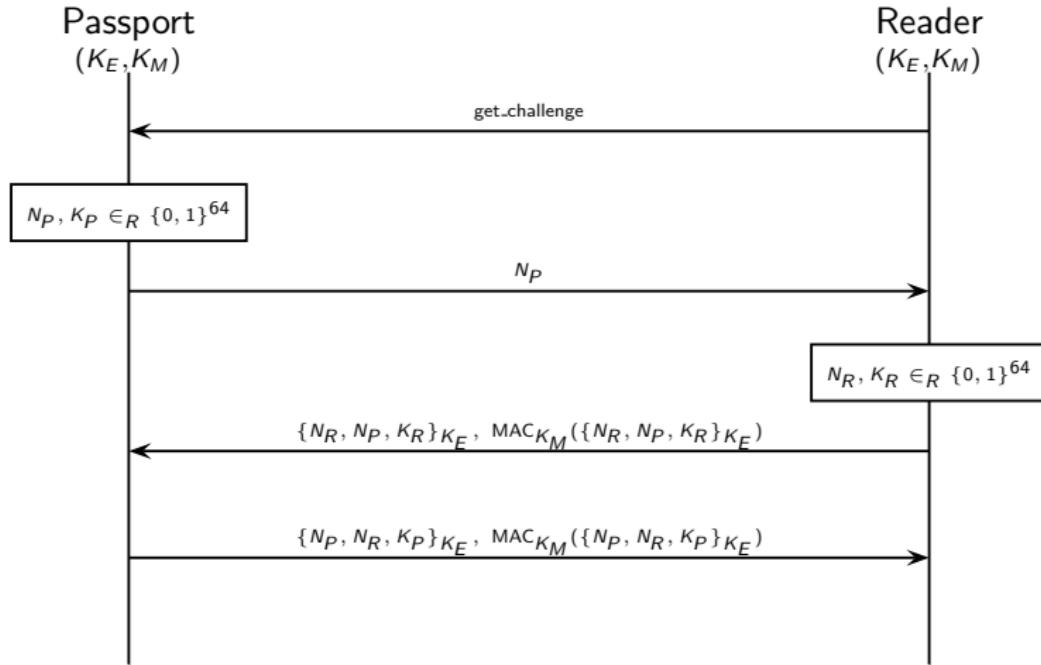
The Basic Access Control protocol (BAC)



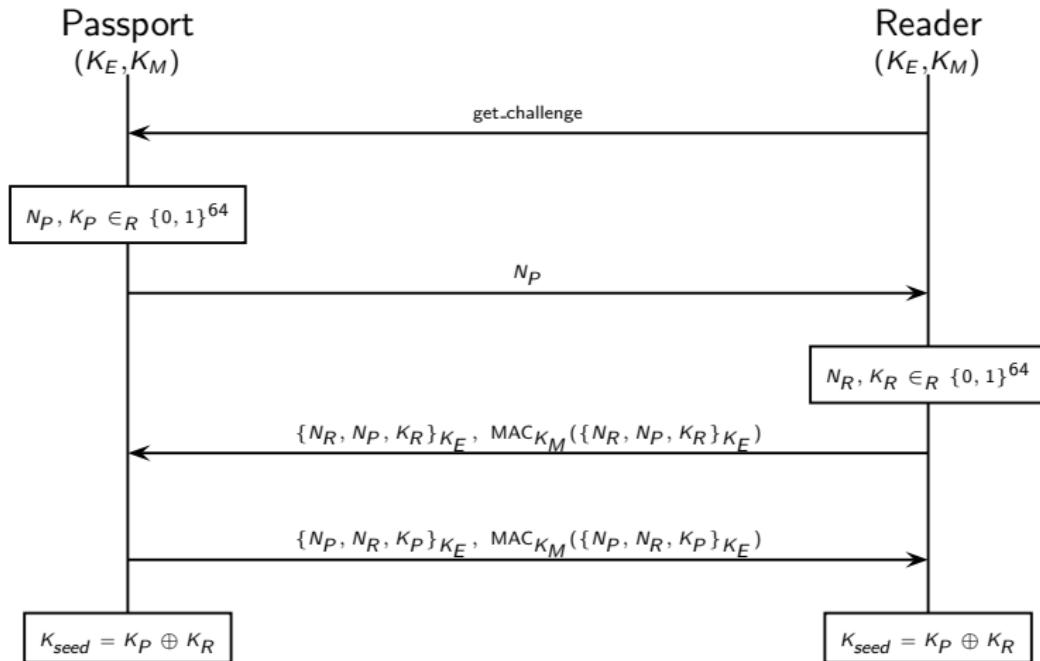
The Basic Access Control protocol (BAC)



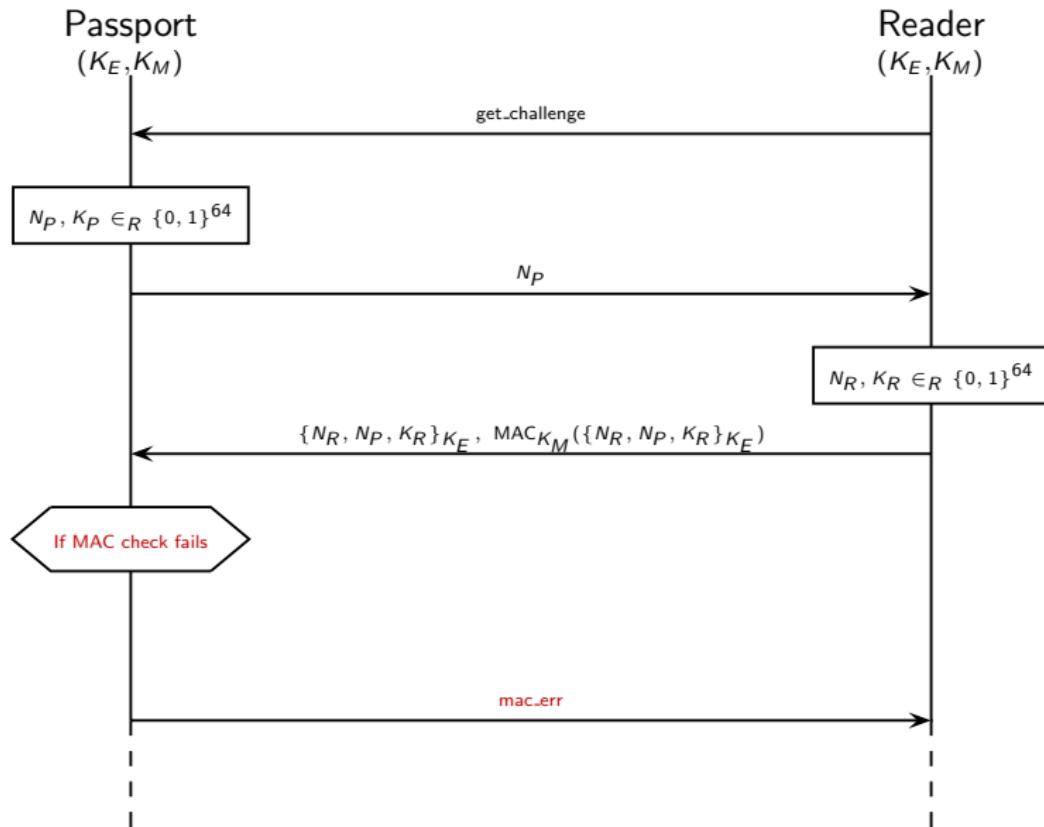
The Basic Access Control protocol (BAC)



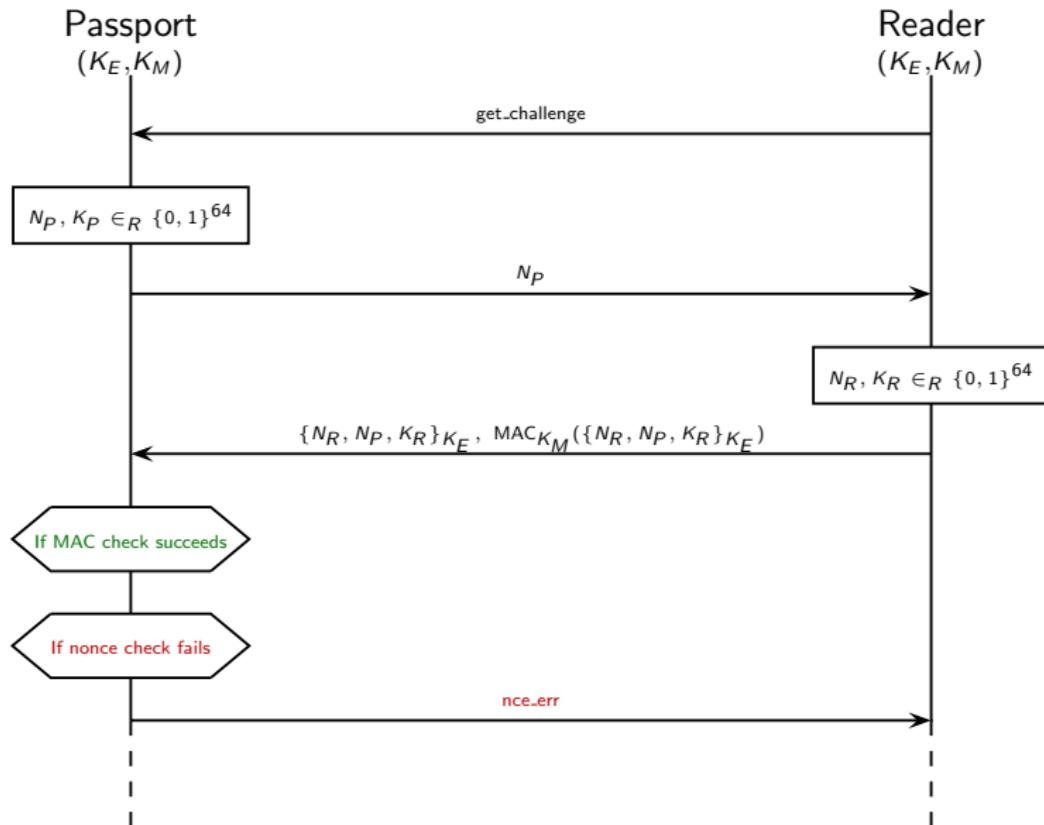
The Basic Access Control protocol (BAC)



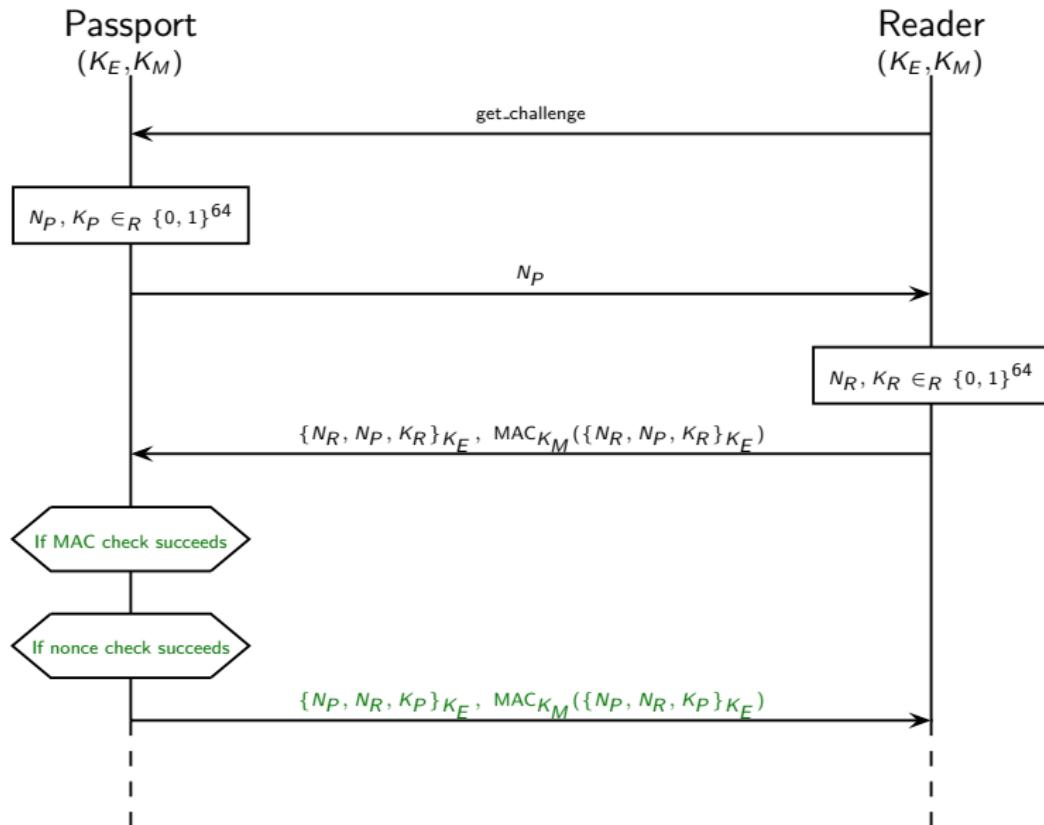
The passport must reply to all received messages



The passport must reply to all received messages



The passport must reply to all received messages



e-Passports and privacy

- ▶ The BAC protocol provides mutual authentication, key agreement, and confidentiality of subsequent communication
- ▶ e-Passports further aim at providing anonymity and unlinkability to their bearers

Definition (ISO 15408)

Anonymity ensures that a user may use of a resource or service without disclosing the user's identity.

Definition (ISO 15408)

Unlinkability ensures that a user may make multiple uses of a resource or service without other users being able to link these uses together.

Different implementations of the BAC protocol

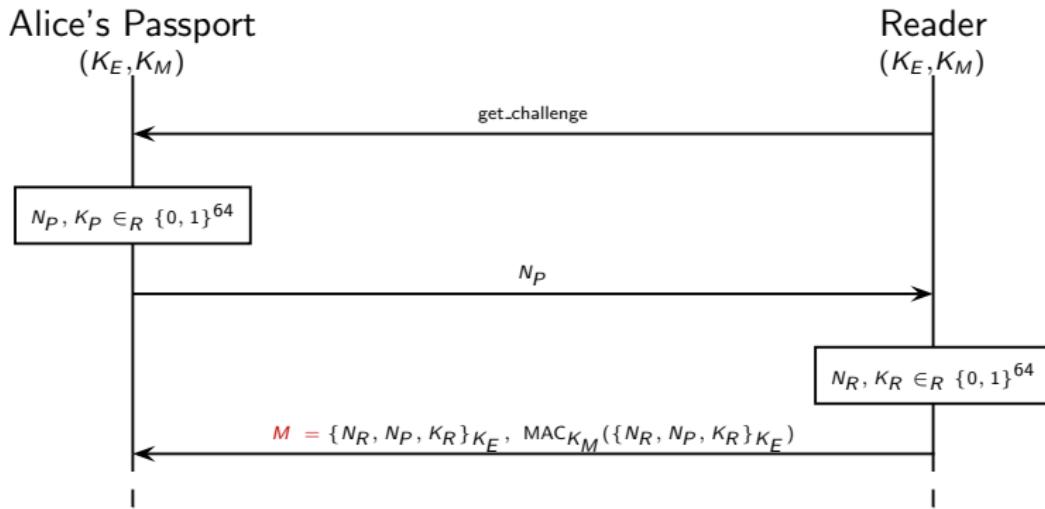
The ICAO e-Passport standard doesn't specify what the error messages should be. Each nation has implemented its own version:

- French e-Passport: $\text{mac_err} \neq \text{nce_err}$
→ French implementation allows an attacker to **track a passport**, provided he has once witnessed a successful authentication.
- British e-Passport: $\text{mac_err} = \text{nce_err}$
→ The British version of the BAC protocol **satisfies unlinkability**.

[T. Chothia, V. Smirnov. "A traceability attack against e-Passports". 14th International Conference on Financial Cryptography and Data Security 2010.]

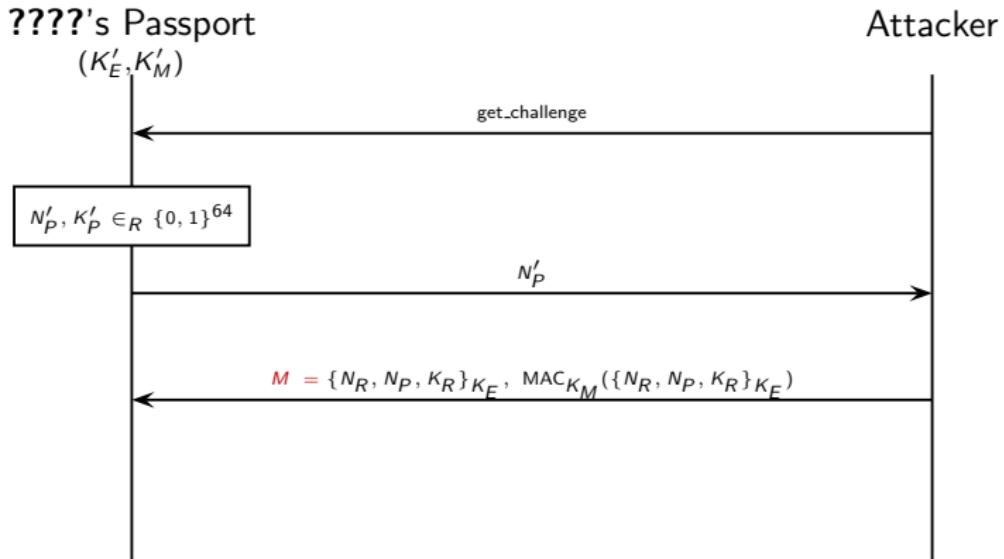
An attack on the French e-Passport (part 1)

The attacker eavesdrop on Alice using her passport

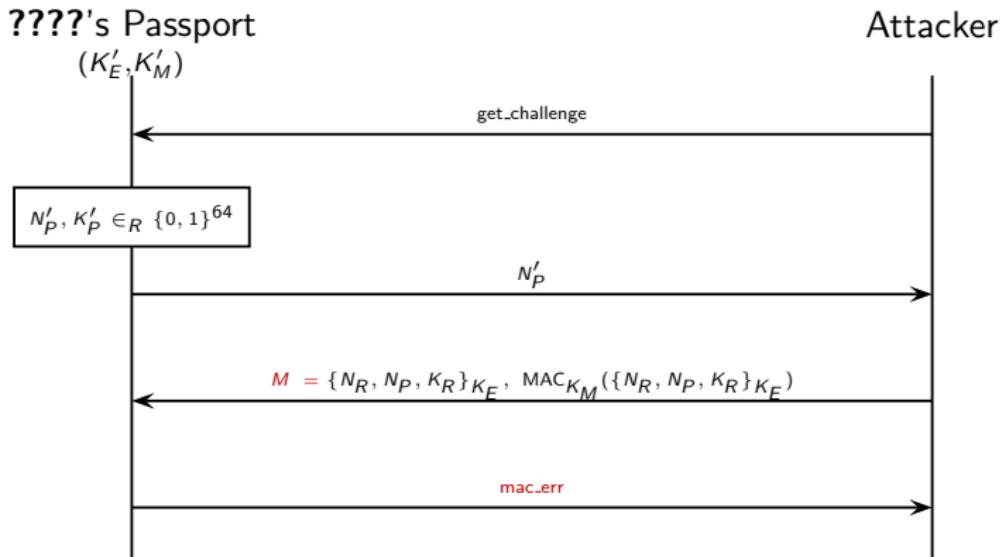


and records message M

An attack on the French e-Passport (part 2)

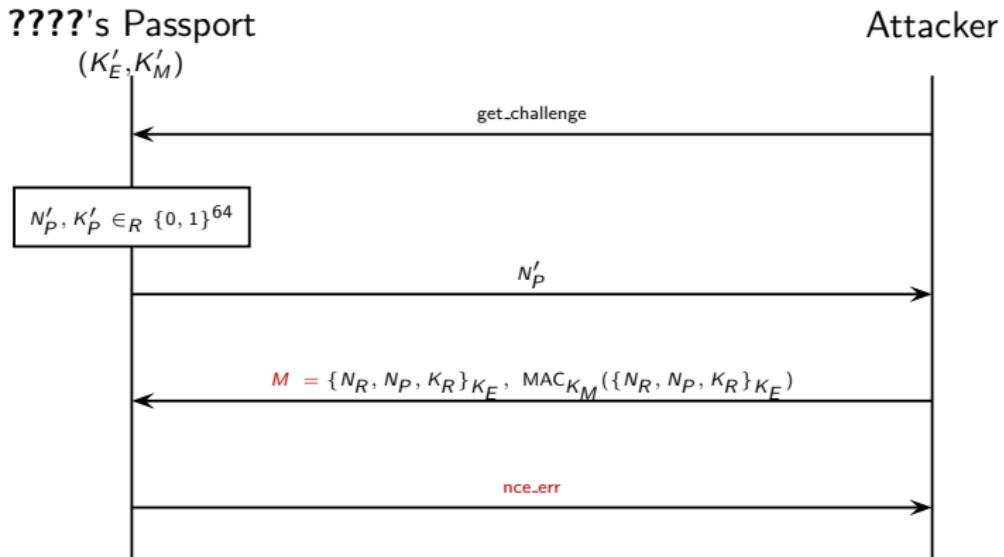


An attack on the French e-Passport (part 2)



\Rightarrow MAC check failed $\Rightarrow K'_M \neq K_M \Rightarrow$ **????** is not Alice

An attack on the French e-Passport (part 2)



\Rightarrow MAC check succeeded $\Rightarrow K'_M = K_M \Rightarrow \text{???? is Alice}$

Timing attack: the failed MAC is rejected sooner

- UK, Greek, German passports return the same error in both situations, but still...

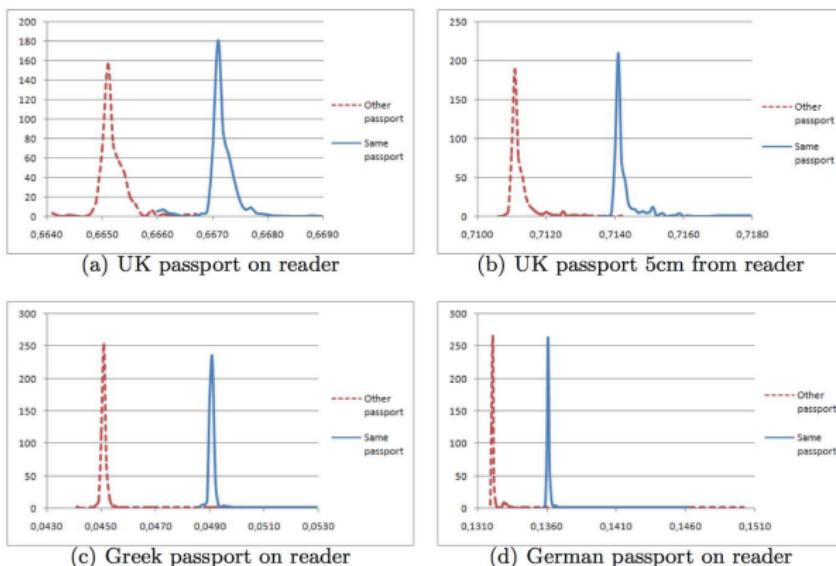


Fig. 4. Sampled Times from Replaying a Message to the Same or a Different Passport

[T. Chothia, V. Smirnov. "A traceability attack against e-Passports". 14th International Conference on Financial Cryptography and Data Security 2010.]

Anonymous communication

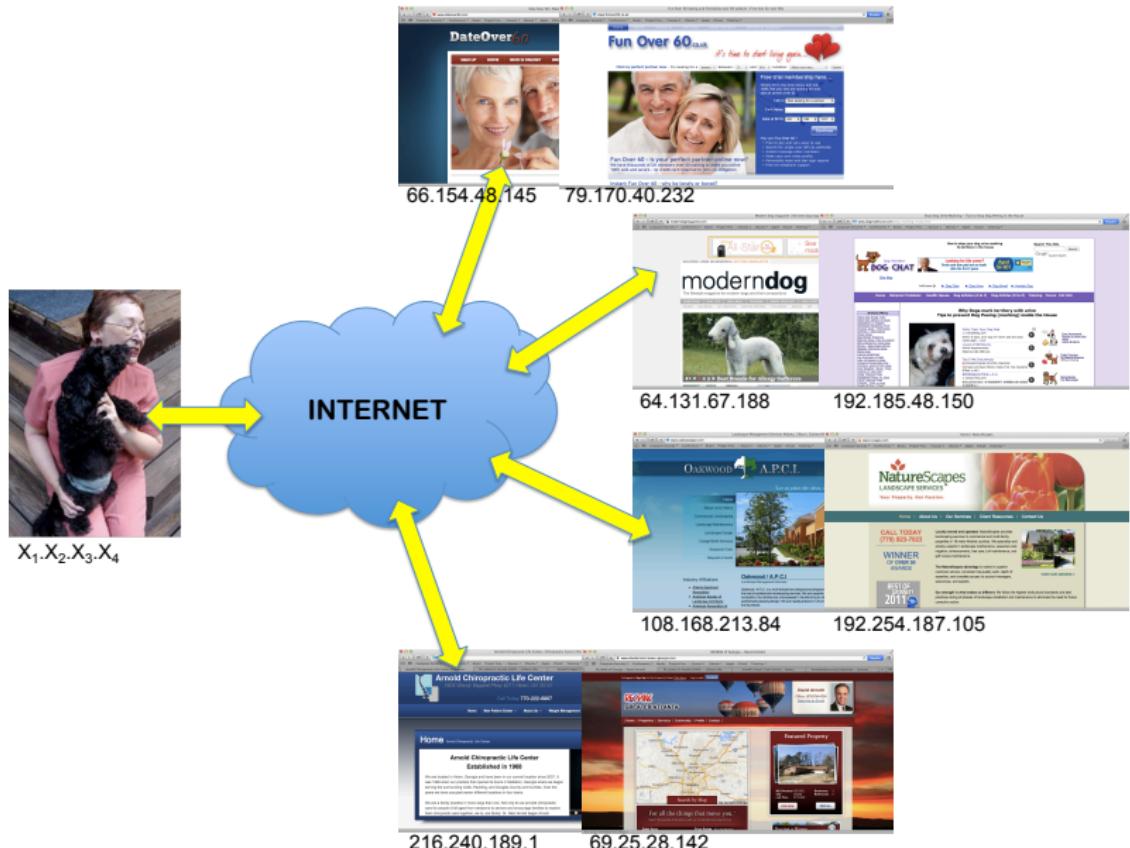
Myrto Arapinis
School of Informatics
University of Edinburgh

February 27, 2019

Context

- ▶ The Internet is a public network:
 - ▶ network routers see all traffic that passes through them
- ▶ Routing information is public:
 - ▶ IP packet headers contain source and destination of packets
- ▶ Encryption does not hide identities:
 - ▶ encryption hides payload, but not routing information

Routing information can reveal who you are!



Routing information can reveal who you are!

A Face Is Exposed for AOL Searcher No. 4417749 – New York Times

www.nytimes.com/2006/08/09/technology/09aol.html?pagewanted=all&_r=0

Computer Security Conferences Books Project Free ... : Season 1 Ubuntu Apple iCloud Tutoring

HOME PAGE MY TIMES TODAY'S PAPER VIDEO MOST POPULAR TIMES TOPICS SUBSCRIBE NOW Log In Register Now

The New York Times Technology

WORLD U.S. N.Y./REGION BUSINESS TECHNOLOGY SCIENCE HEALTH SPORTS OPINION ARTS STYLE TRAVEL JOBS REAL ESTATE AUTOS

CAMCORDERS CAMERAS CELLPHONES COMPUTERS HANDHELDs HOME VIDEO MUSIC PERIPHERALS WI-FI

A Face Is Exposed for AOL Searcher No. 4417749

By MICHAEL BARBARO and TOM ZELLER JR.
Published: August 9, 2006

Buried in a list of 20 million Web search queries collected by AOL and recently released on the Internet is user No. 4417749. The number was assigned by the company to protect the searcher's anonymity, but it was not much of a shield.

 Erik S. Lesser for The New York Times
Thelma Arnold's identity was betrayed by AOL records of her Web searches, like ones for her dog, Dudley, who clearly has a problem.

No. 4417749 conducted hundreds of searches over a three-month period on topics ranging from "numb fingers" to "60 single men" to "dog that urinates on everything."

And search by search, click by click, the identity of AOL user No. 4417749 became easier to discern. There are queries for "landscapers in Lilburn, Ga., several people with the last name Arnold and "homes sold in shadow lake subdivision gwinnett county georgia."

It did not take much investigating to follow that data trail to Thelma Arnold, a 62-year-old widow who lives in Lilburn, Ga., frequently researches her friends' medical ailments and loves her three dogs. "Those are my searches," she said, after a reporter read part of the list to her.

AOL removed the search data from its site over the weekend and apologized for its release, saying it was an AOL removed the search data from its site over the weekend and apologized for its release, saying it was an


The Accenture Digital Difference
Video Gallery Latest Thinking Ad Spotlight
The Accenture Digital Difference
Digital Business is Changing
Accenture Digital—Defining Digital Business

Daily Report: With Cloud Computing, Companies Face a Grid of Tech Choices +
Maps That Use and Encourage Walks + Data +
Detroit, Embracing New Auto Tech Innovations, Sets App Building +
The New York Times The publication of this article is sponsored by Accenture. The editorial staff of The New York Times

Routing information can reveal who you are!

The screenshot shows a web browser window with the address bar displaying "What Is My IP Address? IP Address Tools and More". Below the address bar, the URL "whatismyipaddress.com" is visible. The main content area features a green header with the text "How you connect to the world". Below the header are several navigation links: MY IP, IP LOOKUP, SPEED TEST, BLACKLIST CHECK, TRACE EMAIL, CHANGE IP, HIDE IP, IP TOOLS, LEARN, and COMMUNITY. On the left side, there is a sidebar with various tools: IP Lookup (Know the IP address of another computer), Trace Email (Track down the geographical location and origin of an email you received), Hide IP (Learn how to use a high-deck 'proxyman' to shield your real IP address on the Internet), VPN Comparison (Compare top rated VPN service providers that meet your needs and budget), Blacklist Check (Have you been blacklisted because of the IP address you use? Check to see here.), Speed Test (Is your Internet connection up to speed? Find out for free with a quick click.), and IP Tools (Have the right tool for any job. That goes for your Internet connection, too.). The central main content area displays the user's IPv4 address as 89.241.168.239. It also shows ISP: TalkTalk, City: Edinburgh, Region: Edinburgh, and Country: United Kingdom. A map of the UK highlights the location of Edinburgh. A callout box says "Click for more details about 89.241.168.239". Below the map are links to "Update your IP location" and "Learn More About This IP". To the right of the map, there is a sidebar with the text "This Christmas, people will search for a business like yours." and a "Google AdWords" logo. At the bottom of the page, a blue banner reads "It's not personal — It's just your connection".

Routing information can reveal who you are!



"With your permission, you give us more information about you, about your friends, and we can improve the quality of your searches. We don't need you to type at all. We know where you are. We know where you've been. We can more or less know what you're thinking about."

Eric Schmidt, CEO Google, 2010

Your IP address is your ID

Your IP address is Your ID.



Your IP address leaves behind digital tracks that can be used to identify you and invade your privacy

The McNealy argument



"You have zero privacy anyway. Get over it"

Scott McNealy, CEO Sun Microsystems, 1999

The Schmidt argument



"If you have something that you don't want anyone to know maybe you shouldn't be doing it in the first place"

Eric Schmidt, CEO Google, 2009

Anonymity

Definition (ISO/IEC standard 15408)

A user may use a service or resource without disclosing the user's identity.

Anonymity

Definition (ISO/IEC standard 15408)

A user may use a service or resource without disclosing the user's identity.

→ this can be achieved by hiding one's activities among others' similar activities

- Dinning cryptographers
- Crowds
- Chaum's mix
- Onion routing

Three-party dining cryptographers (3DC) protocol

Three cryptographers are having dinner. Either NSA paid for the dinner, or one of the cryptographers. They want to know if it is the NSA that paid, but without revealing the identity of the cryptographer that paid in the case the NSA did not pay.

3DC protocol:

1. Each cryptographer flips a coin and shows it to his left neighbor:
 - ▶ each cryptographer will see his own coin and his right neighbor's
2. Each cryptographer announces whether the two coins he saw are the same. If he is the payer, he lies
3. odd number of "same" \Rightarrow the NSA paid
even number of "same" \Rightarrow one of the cryptographers paid
 - ▶ only the payer knows he is the one who paid

Superposed sending

- ▶ 3DC protocol generalises to any group size n (nDC)
- ▶ Sender wants to anonymously broadcast a message m :
 1. for each bit of the m , every user generates a random bit and sends it to his left neighbor
 - ▶ every user learns two bits: his own, and his right neighbor's
 2. each user (except the sender) announces (own_bit XOR neighbor's_bit)
 3. the sender announces (own_bit XOR neighbor's_bit XOR message_bit)
 4. XOR of all announcements = message_bit
 - ▶ every randomly generated bit occurs in this sum twice (and is canceled by XOR)
 - ▶ message_bit occurs only once

Limitations of the DC protocol

The DC protocol is impractical:

- ▶ Requires pair-wise shared secret keys (secure channels) between the participants (to share random bits)
- ▶ Requires large amounts of randomness

Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

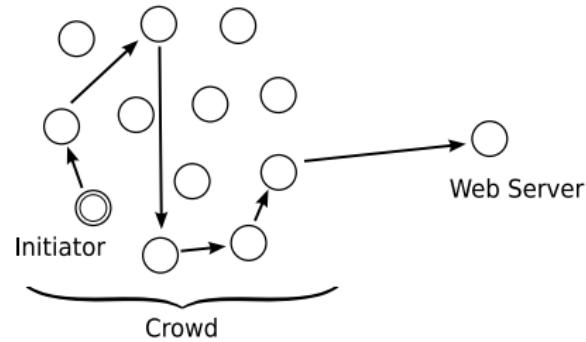
Idea: randomly route the request through a crowd of users

Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- ▶ a crowd is a group of m users; c out of m users may be corrupted

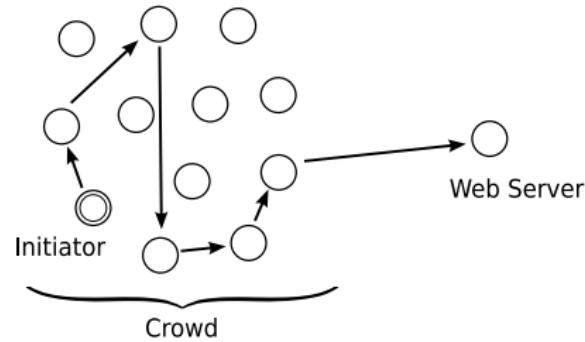


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- ▶ a crowd is a group of m users; c out of m users may be corrupted
- ▶ an initiator that wants to request a webpage creates a path between him and the server:

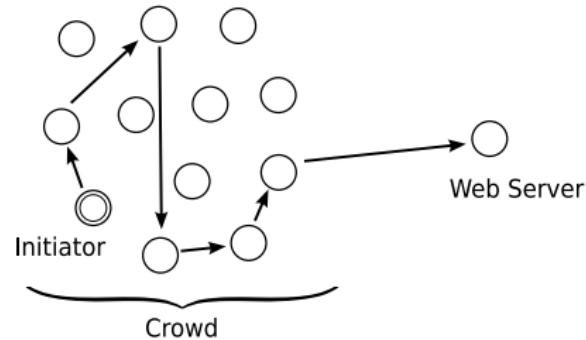


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- ▶ a crowd is a group of m users; c out of m users may be corrupted
- ▶ an initiator that wants to request a webpage creates a path between him and the server:
 1. the initiator selects a forwarder from the crowd and sends him his request

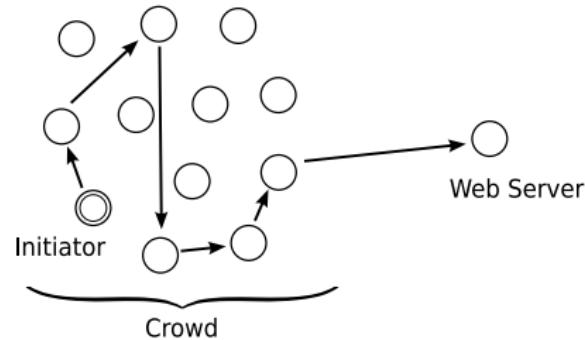


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- ▶ a crowd is a group of m users; c out of m users may be corrupted
- ▶ an initiator that wants to request a webpage creates a path between him and the server:
 1. the initiator selects a forwarder from the crowd and sends him his request
 2. a forwarder delivers the request directly to the server with probability $1 - p_f$; he forwards the request to a randomly selected new forwarder from the crowd with probability p_f ; the new forwarder repeats the procedure

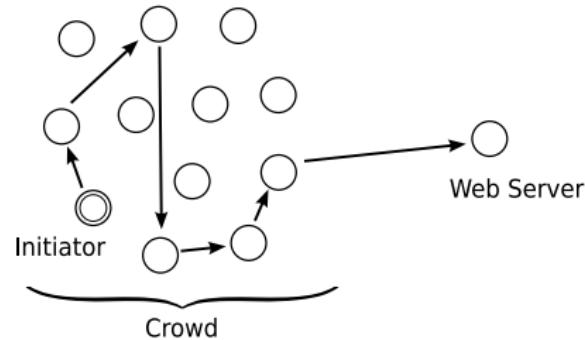


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- ▶ a crowd is a group of m users; c out of m users may be corrupted
- ▶ an initiator that wants to request a webpage creates a path between him and the server:
 1. the initiator selects a forwarder from the crowd and sends him his request
 2. a forwarder delivers the request directly to the server with probability $1 - p_f$; he forwards the request to a randomly selected new forwarder from the crowd with probability p_f ; the new forwarder repeats the procedure
 3. the response from the server follows same route in opposite direction

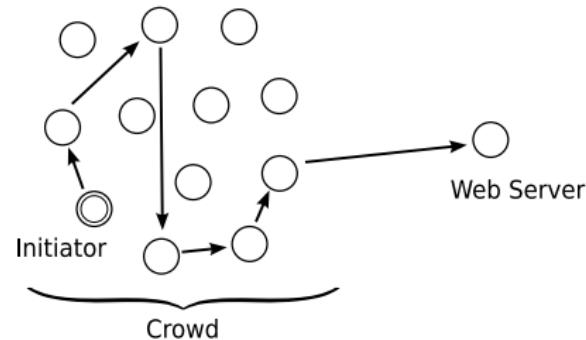


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

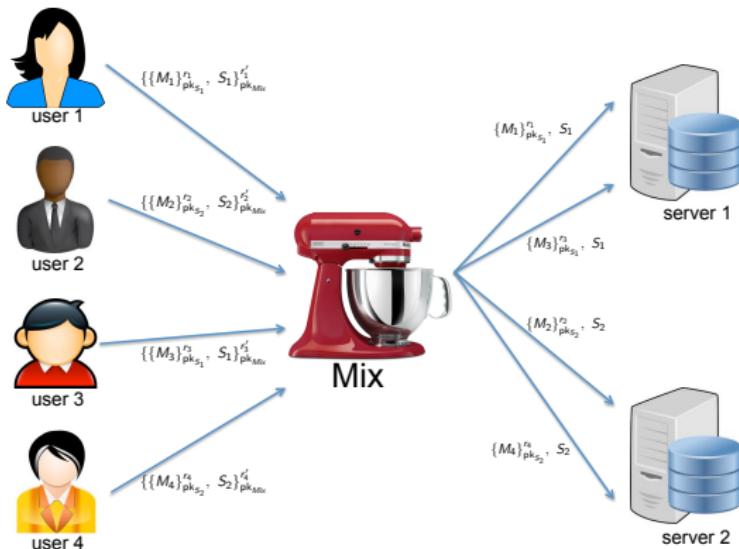
- ▶ a crowd is a group of m users; c out of m users may be corrupted
- ▶ an initiator that wants to request a webpage creates a path between him and the server:
 1. the initiator selects a forwarder from the crowd and sends him his request
 2. a forwarder delivers the request directly to the server with probability $1 - p_f$; he forwards the request to a randomly selected new forwarder from the crowd with probability p_f ; the new forwarder repeats the procedure
 3. the response from the server follows same route in opposite direction



Crowd IS NOT resistant
against an attacker that sees
the whole network traffic!

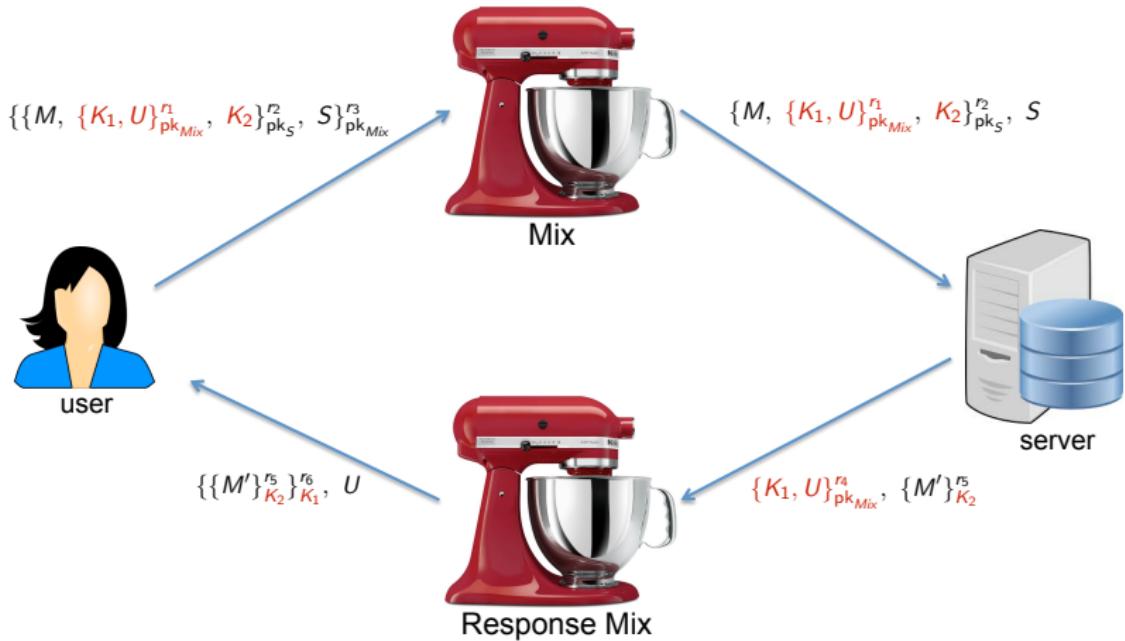
Chaum's mix

[D. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms", Communications of the ACM, February 1981.]

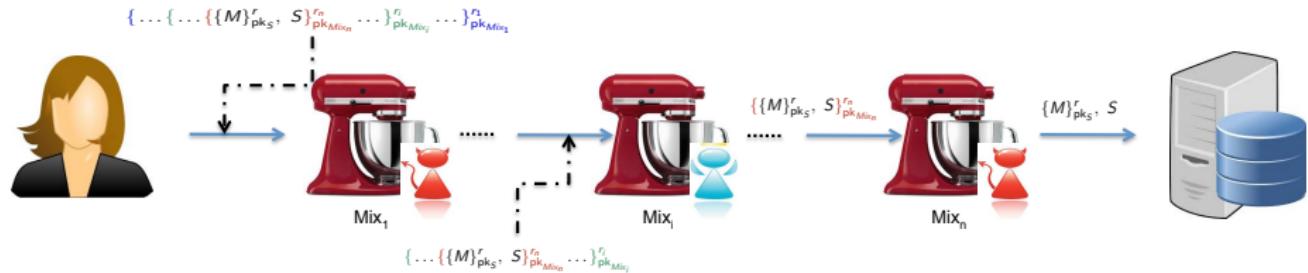


- ▶ **message padding** and **buffering** to avoid time correlation attacks
- ▶ **dummy messages** are generated by the mixes themselves to prevent an attacker sending $n - 1$ messages to a mix with capacity n , allowing him to then link the sender of the n^{th} message with its recipient

Anonymous return addresses



Mix cascade



- ▶ messages are sent through a sequence of mixes
- ▶ some of the mixes may be corrupted
- ▶ a single honest mix guarantees anonymity against an attacker controlling the whole network provided it applies:
 - ▶ message padding
 - ▶ buffering
 - ▶ dummy messages

Limitations of Chaum's mixnets

- ▶ Asymmetric encryption is not efficient
- ▶ Dummy messages are inefficient
- ▶ Buffering is not efficient

Onion routing

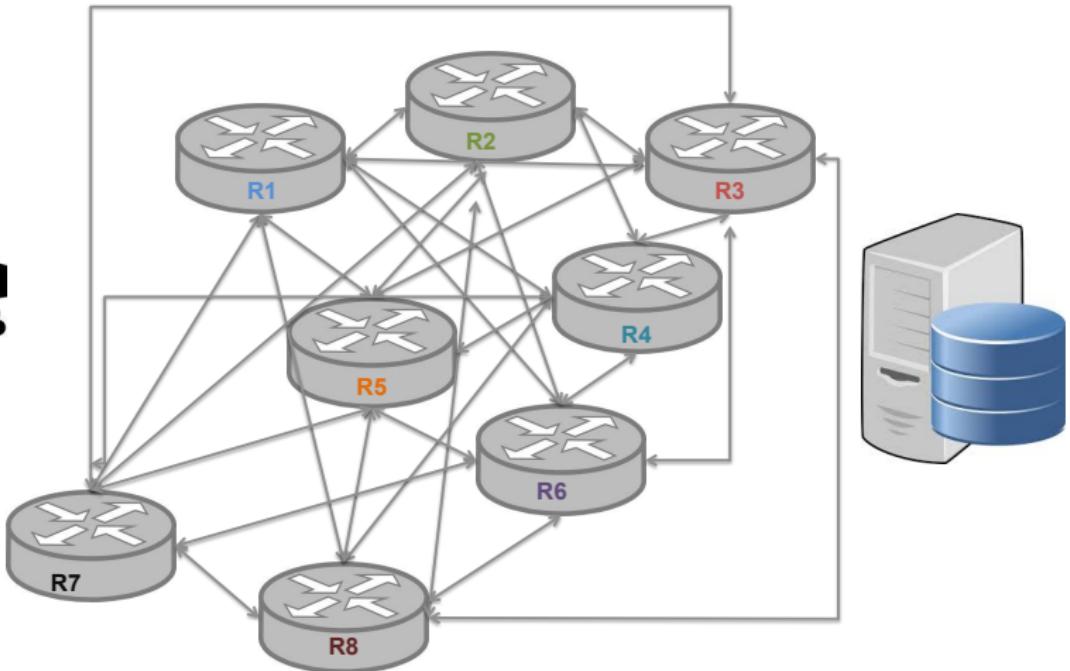
[R. Dingledine, N. Mathewson, and P. F. Syverson: "Tor: The Second-Generation Onion Router", USENIX Security Symposium 2004]

Idea: combine advantages of mixes and proxies

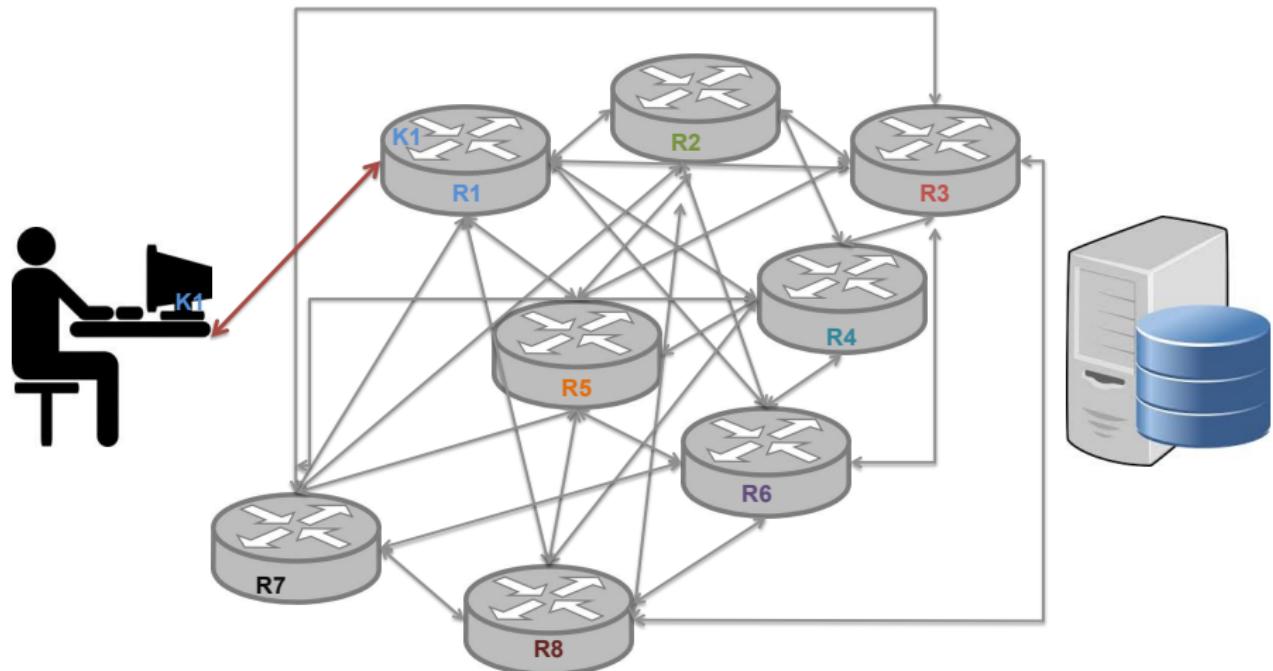
- ▶ use public-key crypto only to establish circuit
- ▶ use symmetric-key crypto to exchange data
- ▶ distribute trust like mixes

But does not defend against attackers that control the hole network

TOR circuit setup

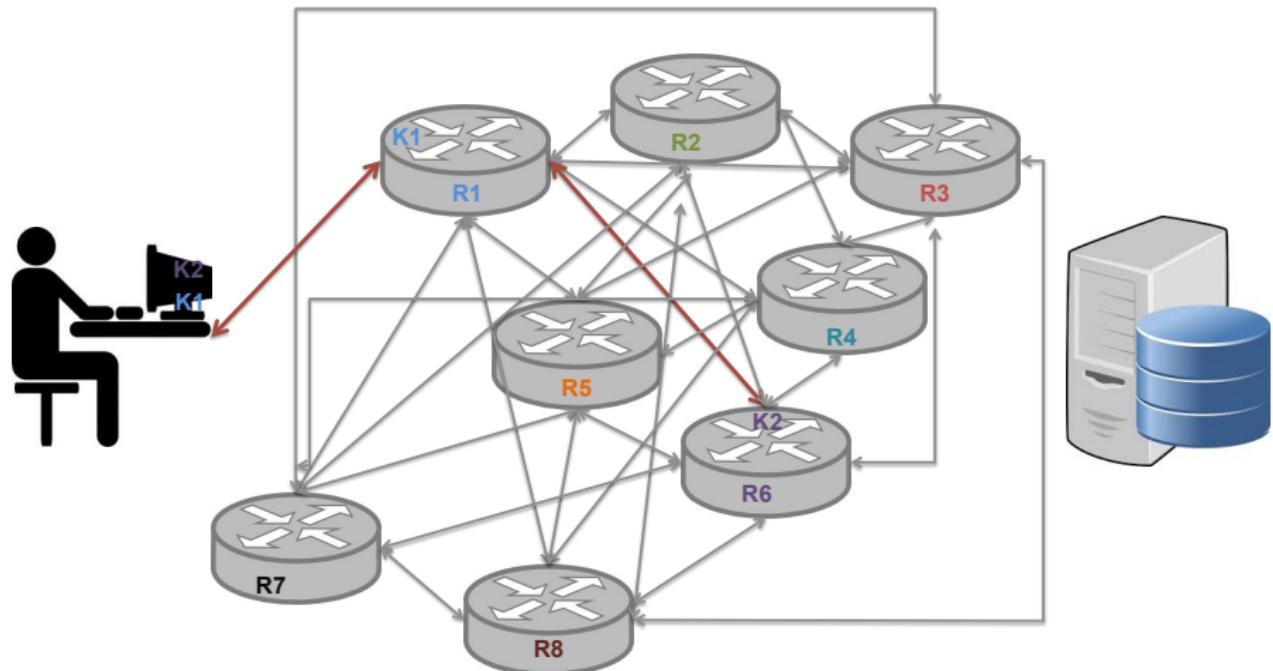


TOR circuit setup



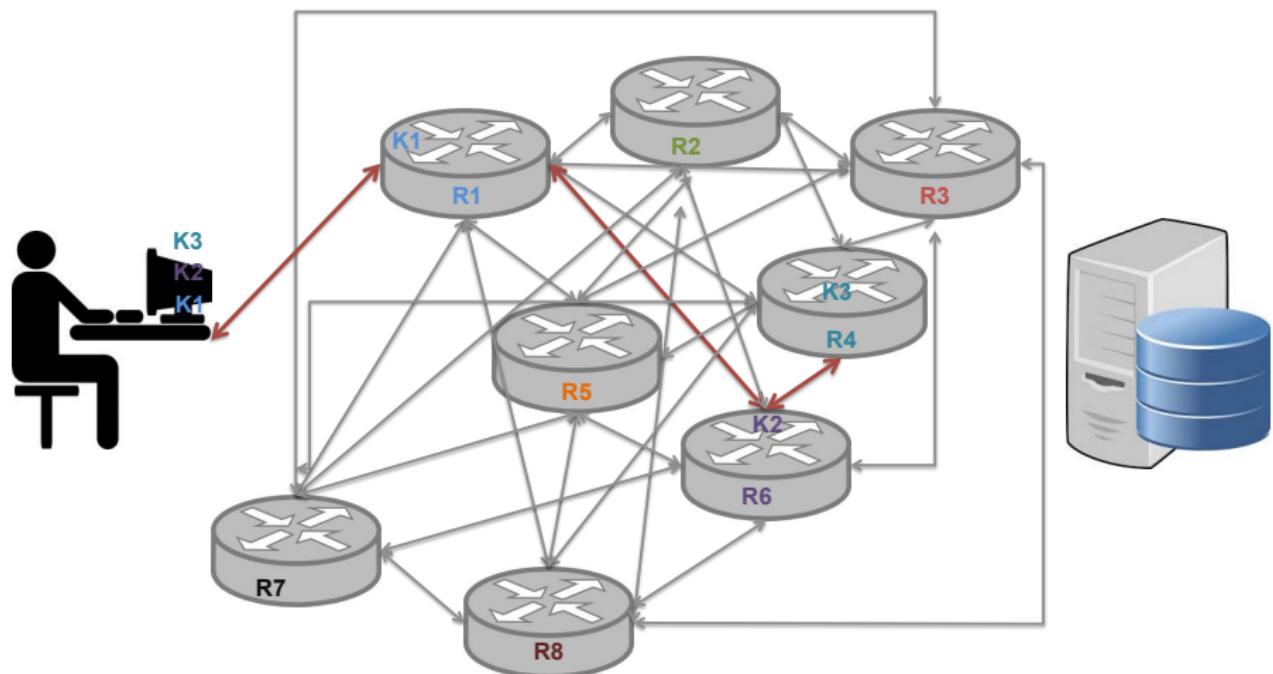
- ▶ client establishes session key **K1** and circuit with Onion Router **R1**

TOR circuit setup



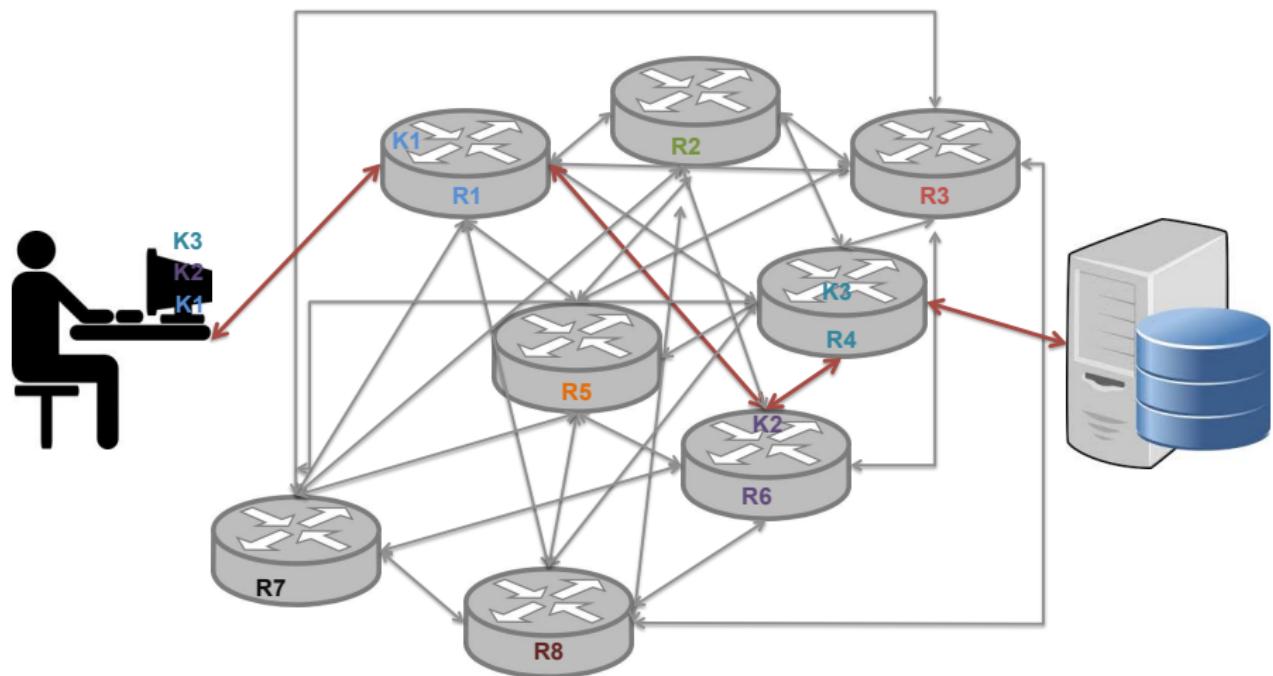
- ▶ client tunnels through that circuit to extend to Onion Router **R6**

TOR circuit setup



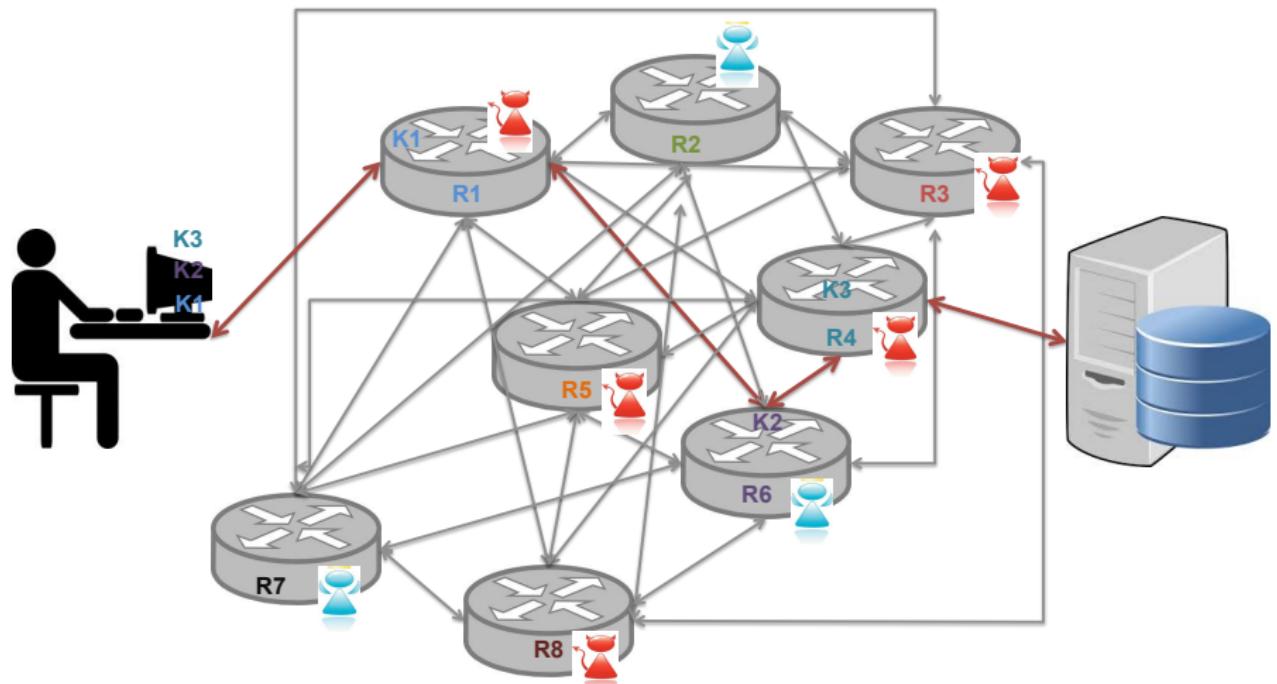
- ▶ client tunnels through that extended circuit to extend to Onion Router **R4**

TOR circuit setup



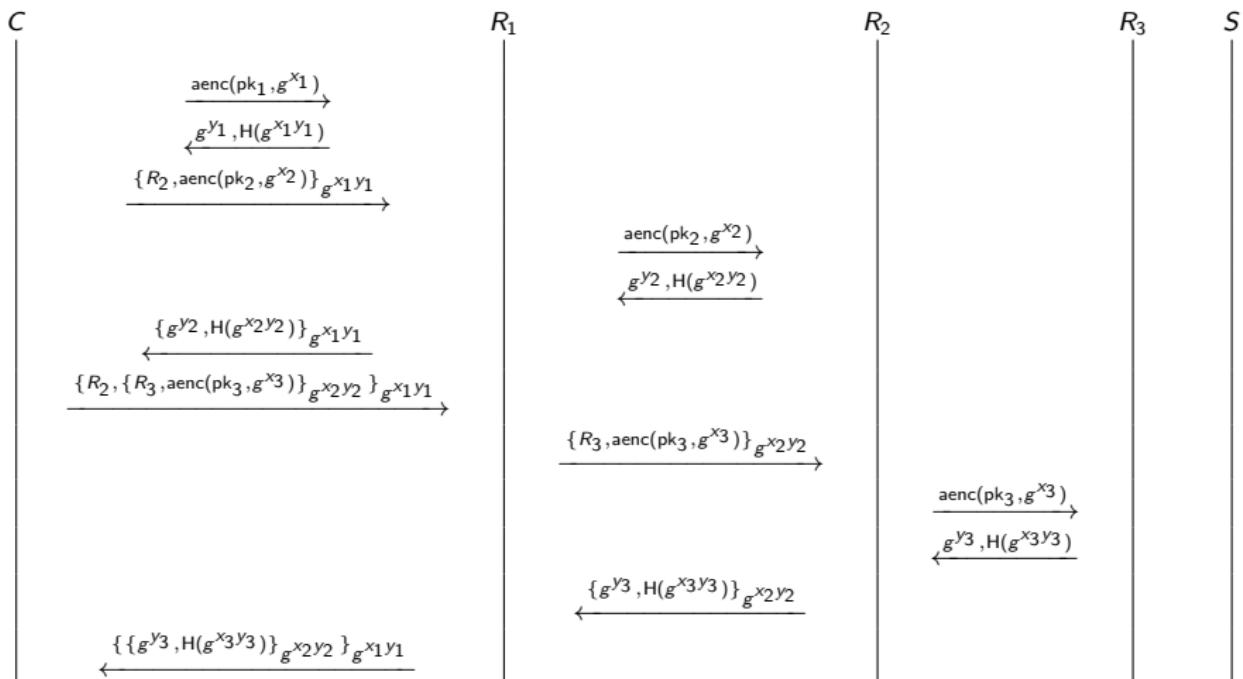
- ▶ client applications connect and communicate over established TOR circuit

TOR circuit setup

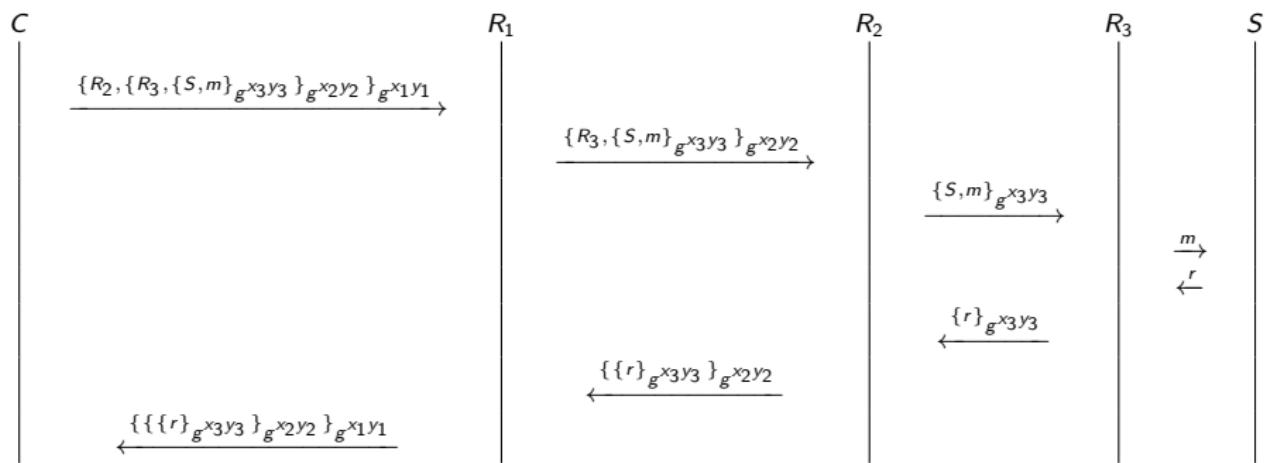


a single honest Onion Router on the TOR circuit guarantees anonymity against an attacker controlling some Onion Routers

The (simplified) TOR message flow - circuit setup



The (simplified) TOR message flow - actual communication



TOR only provides privacy - not confidentiality

- ▶ TOR anonymises the origin of the traffic
- ▶ TOR encrypts everything inside the TOR network
- ▶ but TOR **DOES NOT** encrypt all traffic through the Internet
- ▶ for confidentiality you still need to use end-to-end encryption such as SSL/TLS

TOR takes care of DNS resolution

- ▶ TOR only anonymises TCP streams
- ▶ But, DNS resolution is executed over UDP
- ▶ So, DNS resolution if handled by the client browser defeats the purpose of using TOR
- ▶ To avoid privacy breaches due to DNS resolution, the TOR browser delegates DNS resolution to the exit node

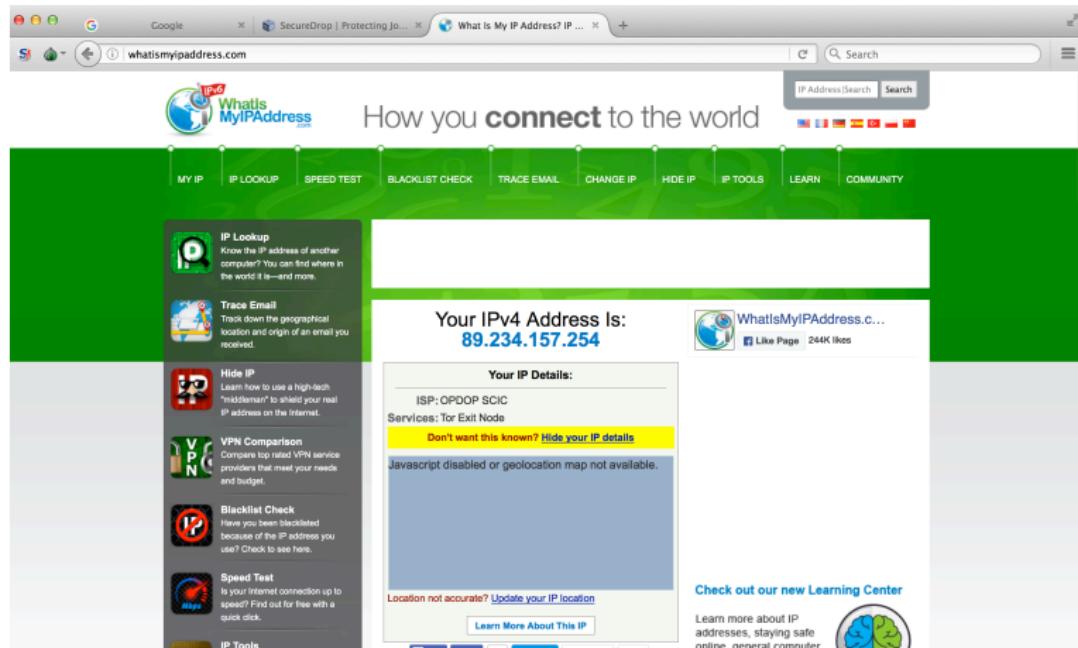
Avoiding censorship

- ▶ TOR relays are listed on the public TOR directory
- ▶ So your local ISP can observe that you are communicating with TOR nodes
- ▶ ISPs and governments can try to block access to the TOR network by blocking TOR relays
- ▶ TOR bridge relays are relays not listed on the public TOR directory
- ▶ Entering the TOR network through a TOR bridge relay can prevent ISPs and governments blocking access to the TOR network

Limitations of TOR

- ▶ TOR does not provide protection against end-to-end timing attacks
- ▶ If the attacker can see both ends of the communication channel, he can correlate volume and timing information on the two sides

whatismyipaddress.com cannot tell where am I using TOR



The screenshot shows a web browser window with multiple tabs open. The active tab is for whatismyipaddress.com. The page displays the IPv4 address **89.234.157.254**. The interface includes a navigation bar with links like MY IP, IP LOOKUP, SPEED TEST, BLACKLIST CHECK, TRACE EMAIL, CHANGE IP, HIDE IP, IP TOOLS, LEARN, and COMMUNITY. On the left, there's a sidebar with icons for IP Lookup, Trace Email, Hide IP, VPN Comparison, Blacklist Check, Speed Test, and IP Tools. A central box highlights the IP address, and another box below it shows "Your IP Details" with fields for ISP, Services, and a note about being a Tor Exit Node. A yellow button at the bottom of this section says "Don't want this known? Hide your IP details". To the right, there's a "Learning Center" section with a brain icon and a "Like Page" button for the Facebook page, which has 244K likes.

How you **connect** to the world

IPV6

What's MyIPAddress.com

MY IP IP LOOKUP SPEED TEST BLACKLIST CHECK TRACE EMAIL CHANGE IP HIDE IP IP TOOLS LEARN COMMUNITY

IP Lookup Know the IP address of another computer? You can find where in the world it is—and more.

Trace Email Track down the geographical location and origin of an email you received.

Hide IP Learn how to use a high-tech "middleman" to shield your real IP address on the Internet.

VPN Comparison Compare top rated VPN service providers that meet your needs and budget.

Blacklist Check Have you been blacklisted because of the IP address you use? Check to see here.

Speed Test Is your Internet connection up to speed? Find out for free with a quick click.

IP Tools

Your IPv4 Address Is:
89.234.157.254

Your IP Details:

ISP: OPDOP SCIC
Services: Tor Exit Node

Don't want this known? Hide your IP details

Javascript disabled or geolocation map not available.

Location not accurate? Update your IP location

Learn More About This IP

Facebook Like Page 244K likes

Check out our new Learning Center

Learn more about IP addresses, staying safe online, general computer

google.com thinks I'm in the Netherlands using TOR

A screenshot of a Mac OS X desktop showing a Tor Browser window. The title bar says "Tor Browser". The main content area shows a Google search results page for "What Is My IP Address?". A context menu is open over the search bar, with the "Tor circuit for this site" option selected. A tooltip from this option reads: "To track you. We recommend that you leave Tor Browser windows in their original default size." Below the menu, a dropdown shows the current circuit: "This browser", "United Kingdom (163.172.21.117)", "France (91.121.23.100)", "Netherlands (46.166.148.177)", and "Internet". The main search results page features the classic Google logo with a play button icon.

New Identity ⌘U
New Tor Circuit for this Site ⌘L

Tor circuit for this site
(google.de):

- This browser
- United Kingdom (163.172.21.117)
- France (91.121.23.100)
- Netherlands (46.166.148.177)
- Internet

OK X

Google

SecureDrop | Protecting jo... | What Is My IP Address? IP ... +

Search

Gmail Bilder

Anmelden

Google-Suche Auf gut Glück!

Google.de angeboten auf: English

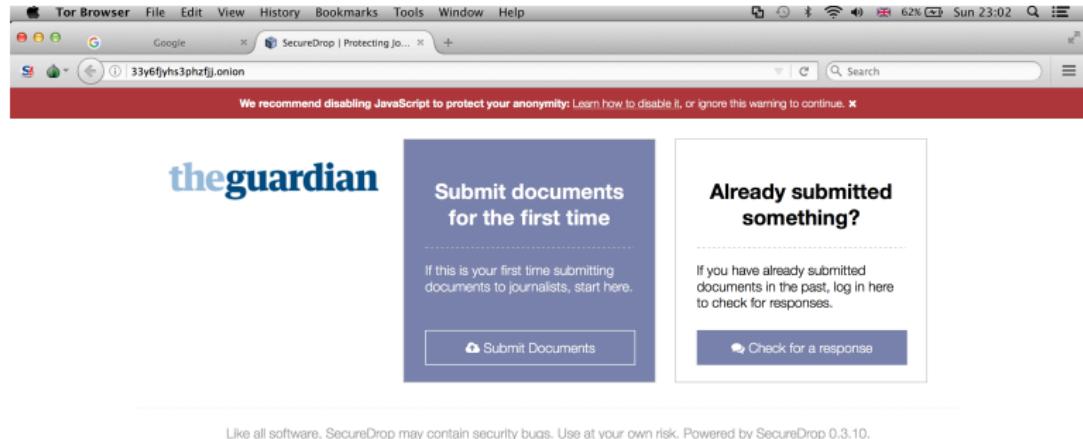
SPÄTER ERINNERN JETZT ANSEHEN

Hinweise zum Datenschutz bei Google

Werbeprogramme Unternehmen Über Google

Datenschutzerklärung Nutzungsbedingungen Einstellungen

TOR hidden services



- ▶ TOR can also provide anonymity to websites and servers
- ▶ www.torproject.org/docs/hidden-services.html

Anonymous communication

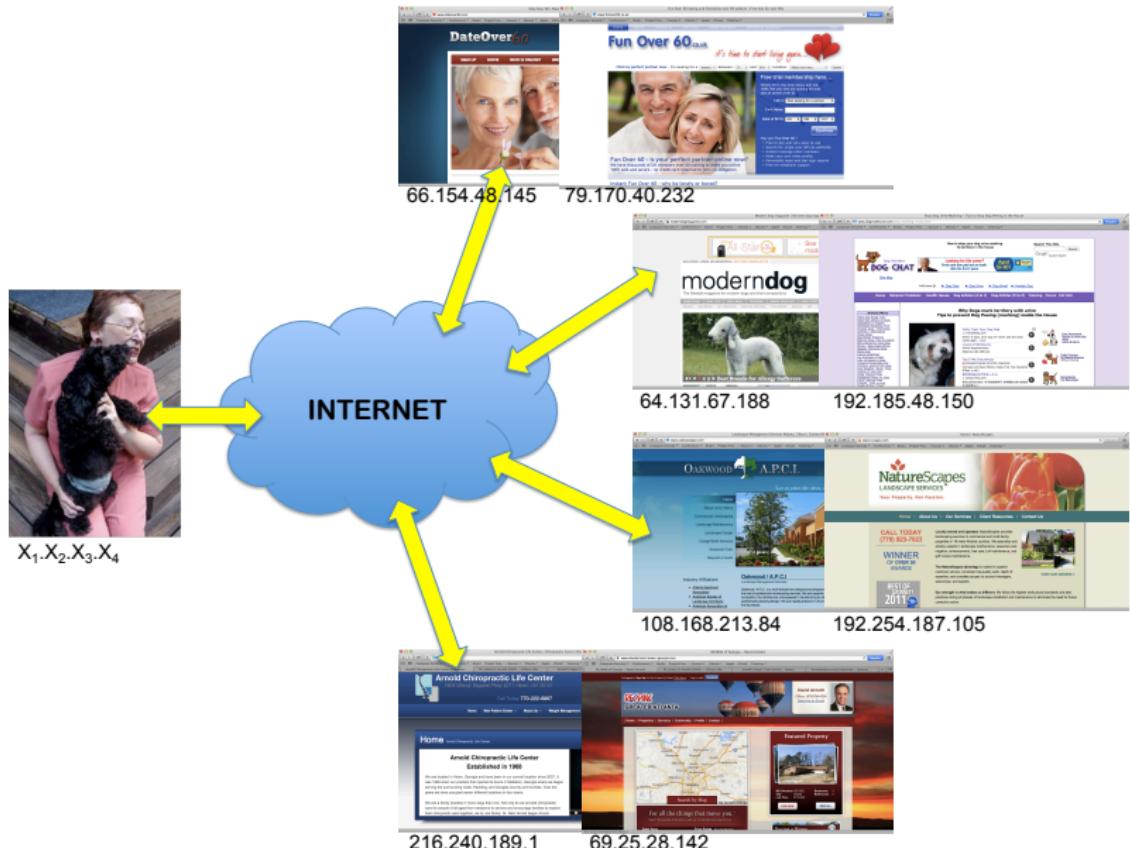
Myrto Arapinis
School of Informatics
University of Edinburgh

February 27, 2019

Context

- ▶ The Internet is a public network:
 - ▶ network routers see all traffic that passes through them
- ▶ Routing information is public:
 - ▶ IP packet headers contain source and destination of packets
- ▶ Encryption does not hide identities:
 - ▶ encryption hides payload, but not routing information

Routing information can reveal who you are!



Routing information can reveal who you are!

A Face Is Exposed for AOL Searcher No. 4417749 – New York Times

www.nytimes.com/2006/08/09/technology/09aol.html?pagewanted=all&_r=0

Computer Security Conferences Books Project Fre... Season 1 Ubuntu Apple iCloud Tutoring

HOME PAGE MY TIMES TODAY'S PAPER VIDEO MOST POPULAR TIMES TOPICS SUBSCRIBE NOW Log In Register Now

The New York Times Technology

WORLD U.S. N.Y./REGION BUSINESS TECHNOLOGY SCIENCE HEALTH SPORTS OPINION ARTS STYLE TRAVEL JOBS REAL ESTATE AUTOS

CAMCORDERS CAMERAS CELLPHONES COMPUTERS HANDHELDs HOME VIDEO MUSIC PERIPHERALS WI-FI

A Face Is Exposed for AOL Searcher No. 4417749

By MICHAEL BARBARO and TOM ZELLER JR.
Published: August 9, 2006

Buried in a list of 20 million Web search queries collected by AOL and recently released on the Internet is user No. 4417749. The number was assigned by the company to protect the searcher's anonymity, but it was not much of a shield.

 Erik S. Lesser for The New York Times
Thelma Arnold's identity was betrayed by AOL records of her Web searches, like ones for her dog, Dudley, who clearly has a problem.

No. 4417749 conducted hundreds of searches over a three-month period on topics ranging from "numb fingers" to "60 single men" to "dog that urinates on everything."

And search by search, click by click, the identity of AOL user No. 4417749 became easier to discern. There are queries for "landscapers in Lilburn, Ga., several people with the last name Arnold and "homes sold in shadow lake subdivision gwinnett county georgia."

It did not take much investigating to follow that data trail to Thelma Arnold, a 62-year-old widow who lives in Lilburn, Ga., frequently researches her friends' medical ailments and loves her three dogs. "Those are my searches," she said, after a reporter read part of the list to her.

AOL removed the search data from its site over the weekend and apologized for its release, saying it was an AOL removed the search data from its site over the weekend and apologized for its release, saying it was an


The Accenture Digital Difference
Video Gallery Latest Thinking Ad Spotlight
The Accenture Digital Difference
Digital Business is Changing
Accenture Digital—Defining Digital Business

Daily Report: With Cloud Computing, Companies Face a Grid of Tech Choices +
Maps That Use and Encourage Walks + Data +
Detroit, Embracing New Auto Tech Innovations, Sets App Building +
The New York Times The publication of this article is sponsored by Accenture. The editorial staff of The New York Times

Routing information can reveal who you are!

The screenshot shows a Safari browser window on a Mac OS X desktop. The title bar reads "Safari File Edit View History Bookmarks Develop Window Help". The address bar shows the URL "whatismyipaddress.com". The page content is from the "What's My IP Address?" website, which has a green header with the tagline "How you connect to the world". The main content area displays the user's IPv4 address as 89.241.168.239. It includes a map of the UK with a red dot indicating the location of the IP address. A sidebar on the left lists various tools: IP Lookup, Trace Email, Hide IP, VPN Comparison, Blacklist Check, Speed Test, and IP Tools. A sidebar on the right features a Google AdWords banner and a "Check out our new Learning Center" section with a lightbulb icon.

What Is My IP Address? IP Address Tools and More

ComputerSecurity SimSec CSExam La cryptogra... ts dévoilés Conferences ResearchProfiles Security-Club Teaching Tutoring

IP AddressSearch Search

How you **connect** to the world

MY IP IP LOOKUP SPEED TEST BLACKLIST CHECK TRACE EMAIL CHANGE IP HIDE IP IP TOOLS LEARN COMMUNITY

IP Lookup Know the IP address of another computer? You can find where in the world it is—and more.

Trace Email Track down the geographical location and origin of an email you received.

Hide IP Learn how to use a high-dec "proxyserver" to shield your real IP address on the Internet.

VPN Comparison Compare top rated VPN service providers that meet your needs and budget.

Blacklist Check Have you been blacklisted because of the IP address you use? Check to see here.

Speed Test Is your Internet connection up to speed? Find out for free with a quick click.

IP Tools Have the right tool for any job. That goes for your Internet connection, too.

Your IPv4 Address Is:
89.241.168.239

ISP: TalkTalk
City: Edinburgh
Region: Edinburgh
Country: United Kingdom

Don't want this known? Hide your IP details

Click for more details about 89.241.168.239

Leaflet | OpenStreetMap Tiles

Location not accurate? Update your IP location

Learn More About This IP

Twitter Share 6.2k

This Christmas, people will search for a business like yours.

Google AdWords

Check out our new Learning Center

Learn more about IP addresses, staying safe online, general computer topics and more, including a look at IPv6.

Start Here

If it's not personal — it's just your connection

Routing information can reveal who you are!



"With your permission, you give us more information about you, about your friends, and we can improve the quality of your searches. We don't need you to type at all. We know where you are. We know where you've been. We can more or less know what you're thinking about."

Eric Schmidt, CEO Google, 2010

Your IP address is your ID

Your IP address is Your ID.



Your IP address leaves behind digital tracks that can be used to identify you and invade your privacy

The McNealy argument



"You have zero privacy anyway. Get over it"

Scott McNealy, CEO Sun Microsystems, 1999

The Schmidt argument



"If you have something that you don't want anyone to know maybe you shouldn't be doing it in the first place"

Eric Schmidt, CEO Google, 2009

Anonymity

Definition (ISO/IEC standard 15408)

A user may use a service or resource without disclosing the user's identity.

Anonymity

Definition (ISO/IEC standard 15408)

A user may use a service or resource without disclosing the user's identity.

→ this can be achieved by hiding one's activities among others' similar activities

- Dinning cryptographers
- Crowds
- Chaum's mix
- Onion routing

Three-party dining cryptographers (3DC) protocol

Three cryptographers are having dinner. Either NSA paid for the dinner, or one of the cryptographers. They want to know if it is the NSA that paid, but without revealing the identity of the cryptographer that paid in the case the NSA did not pay.

3DC protocol:

1. Each cryptographer flips a coin and shows it to his left neighbor:
 - ▶ each cryptographer will see his own coin and his right neighbor's
2. Each cryptographer announces whether the two coins he saw are the same. If he is the payer, he lies
3. odd number of "same" \Rightarrow the NSA paid
even number of "same" \Rightarrow one of the cryptographers paid
 - ▶ only the payer knows he is the one who paid

Superposed sending

- ▶ 3DC protocol generalises to any group size n (nDC)
- ▶ Sender wants to anonymously broadcast a message m :
 1. for each bit of the m , every user generates a random bit and sends it to his left neighbor
 - ▶ every user learns two bits: his own, and his right neighbor's
 2. each user (except the sender) announces (own_bit XOR neighbor's_bit)
 3. the sender announces (own_bit XOR neighbor's_bit XOR message_bit)
 4. XOR of all announcements = message_bit
 - ▶ every randomly generated bit occurs in this sum twice (and is canceled by XOR)
 - ▶ message_bit occurs only once

Limitations of the DC protocol

The DC protocol is impractical:

- ▶ Requires pair-wise shared secret keys (secure channels) between the participants (to share random bits)
- ▶ Requires large amounts of randomness

Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

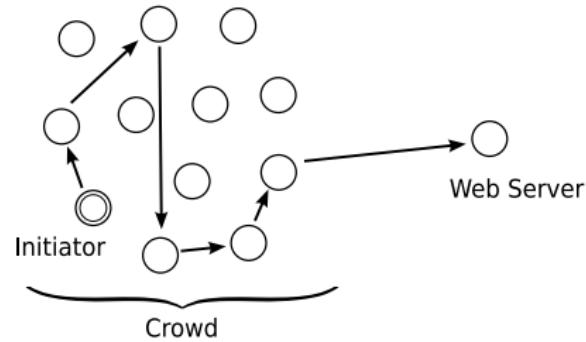
Idea: randomly route the request through a crowd of users

Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- ▶ a crowd is a group of m users; c out of m users may be corrupted

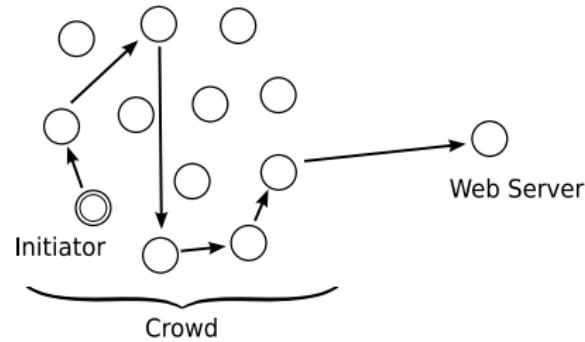


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- ▶ a crowd is a group of m users; c out of m users may be corrupted
- ▶ an initiator that wants to request a webpage creates a path between him and the server:

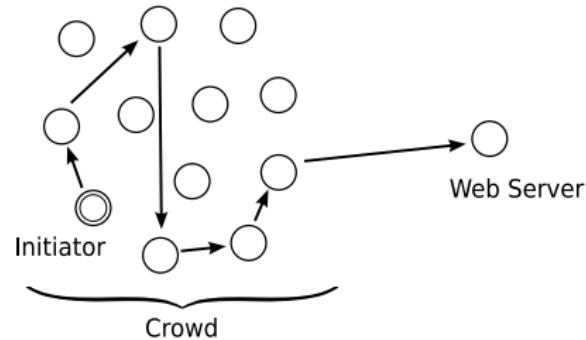


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- ▶ a crowd is a group of m users; c out of m users may be corrupted
- ▶ an initiator that wants to request a webpage creates a path between him and the server:
 1. the initiator selects a forwarder from the crowd and sends him his request

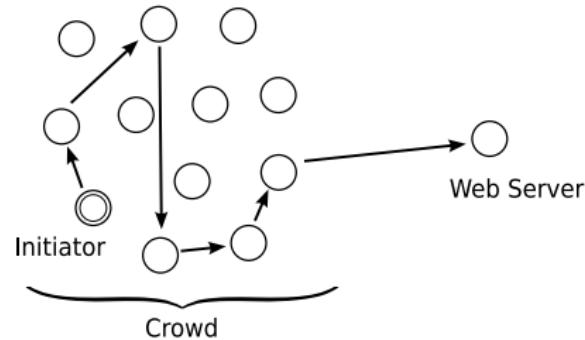


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- ▶ a crowd is a group of m users; c out of m users may be corrupted
- ▶ an initiator that wants to request a webpage creates a path between him and the server:
 1. the initiator selects a forwarder from the crowd and sends him his request
 2. a forwarder delivers the request directly to the server with probability $1 - p_f$; he forwards the request to a randomly selected new forwarder from the crowd with probability p_f ; the new forwarder repeats the procedure

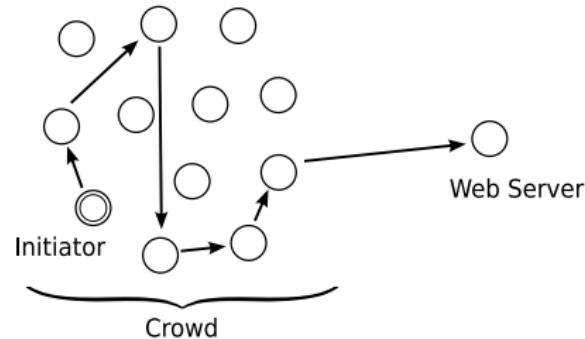


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- ▶ a crowd is a group of m users; c out of m users may be corrupted
- ▶ an initiator that wants to request a webpage creates a path between him and the server:
 1. the initiator selects a forwarder from the crowd and sends him his request
 2. a forwarder delivers the request directly to the server with probability $1 - p_f$; he forwards the request to a randomly selected new forwarder from the crowd with probability p_f ; the new forwarder repeats the procedure
 3. the response from the server follows same route in opposite direction

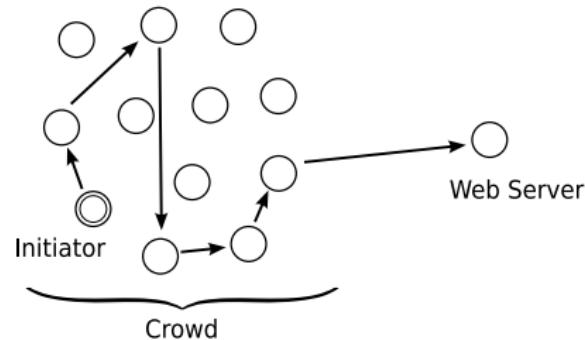


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

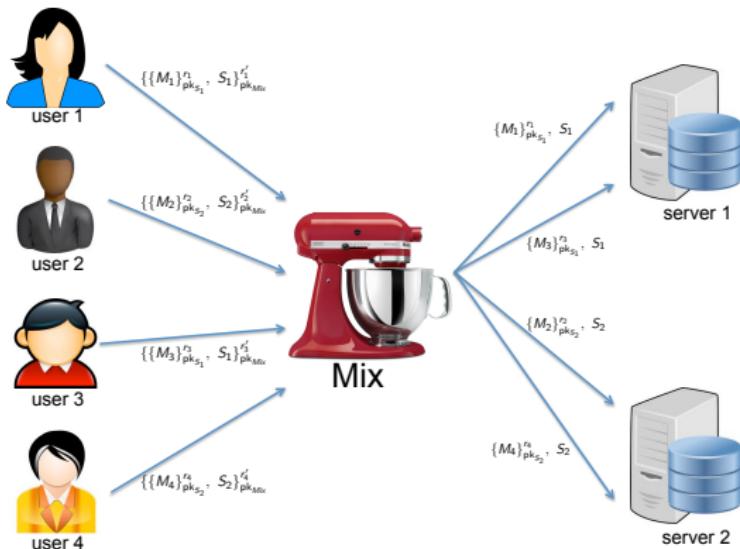
- ▶ a crowd is a group of m users; c out of m users may be corrupted
- ▶ an initiator that wants to request a webpage creates a path between him and the server:
 1. the initiator selects a forwarder from the crowd and sends him his request
 2. a forwarder delivers the request directly to the server with probability $1 - p_f$; he forwards the request to a randomly selected new forwarder from the crowd with probability p_f ; the new forwarder repeats the procedure
 3. the response from the server follows same route in opposite direction



Crowd IS NOT resistant
against an attacker that sees
the whole network traffic!

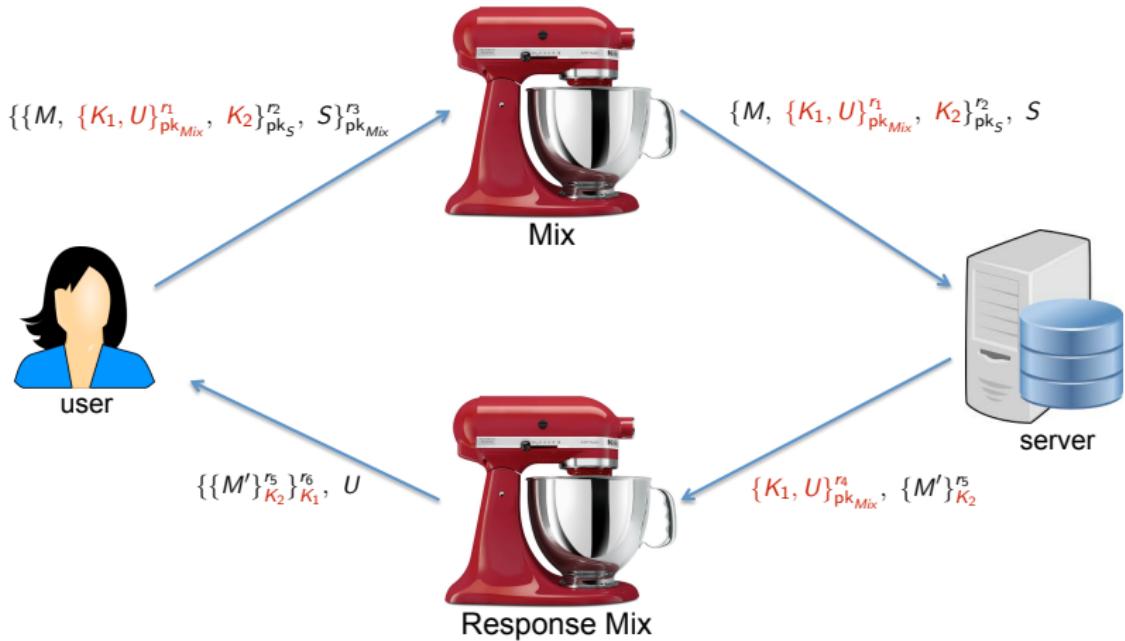
Chaum's mix

[D. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms", Communications of the ACM, February 1981.]

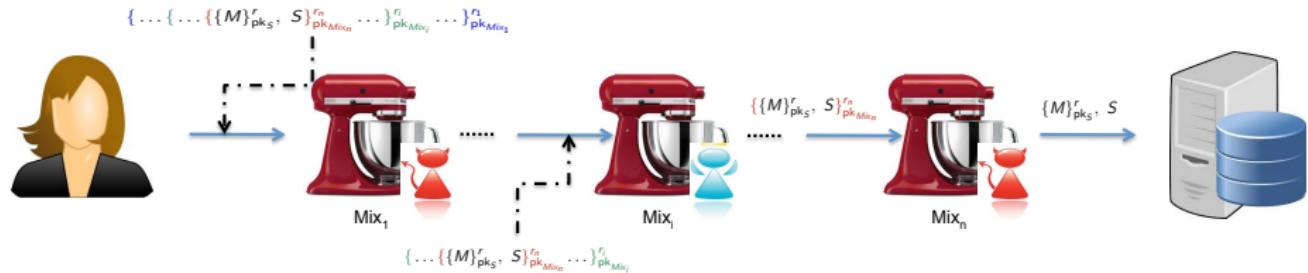


- ▶ **message padding** and **buffering** to avoid time correlation attacks
- ▶ **dummy messages** are generated by the mixes themselves to prevent an attacker sending $n - 1$ messages to a mix with capacity n , allowing him to then link the sender of the n^{th} message with its recipient

Anonymous return addresses



Mix cascade



- ▶ messages are sent through a sequence of mixes
- ▶ some of the mixes may be corrupted
- ▶ a single honest mix guarantees anonymity against an attacker controlling the whole network provided it applies:
 - ▶ message padding
 - ▶ buffering
 - ▶ dummy messages

Limitations of Chaum's mixnets

- ▶ Asymmetric encryption is not efficient
- ▶ Dummy messages are inefficient
- ▶ Buffering is not efficient

Onion routing

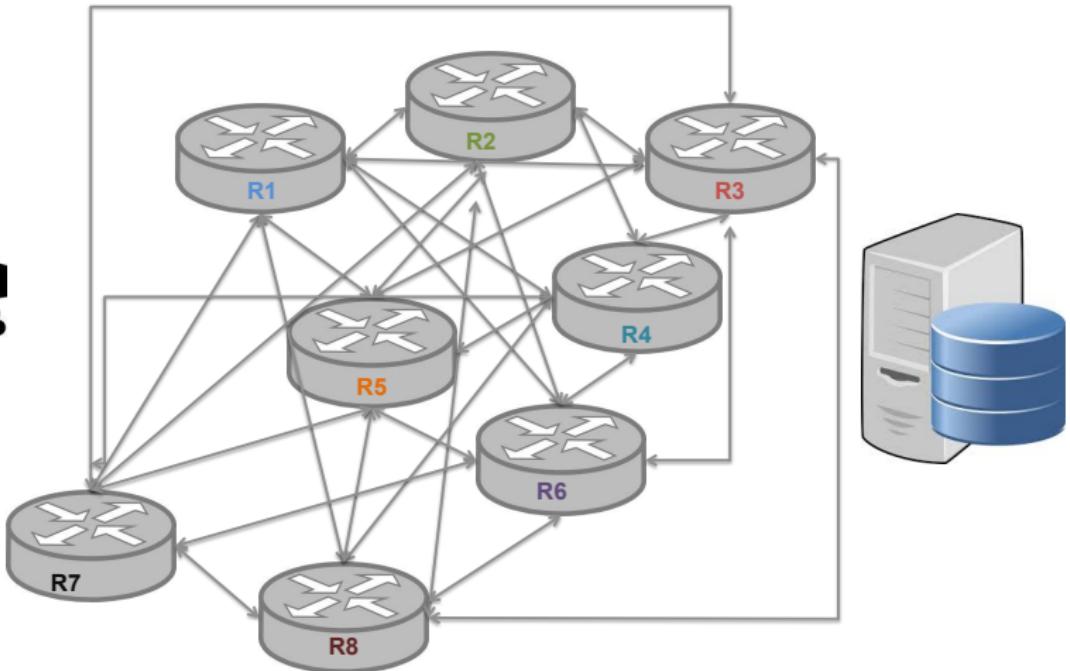
[R. Dingledine, N. Mathewson, and P. F. Syverson: "Tor: The Second-Generation Onion Router", USENIX Security Symposium 2004]

Idea: combine advantages of mixes and proxies

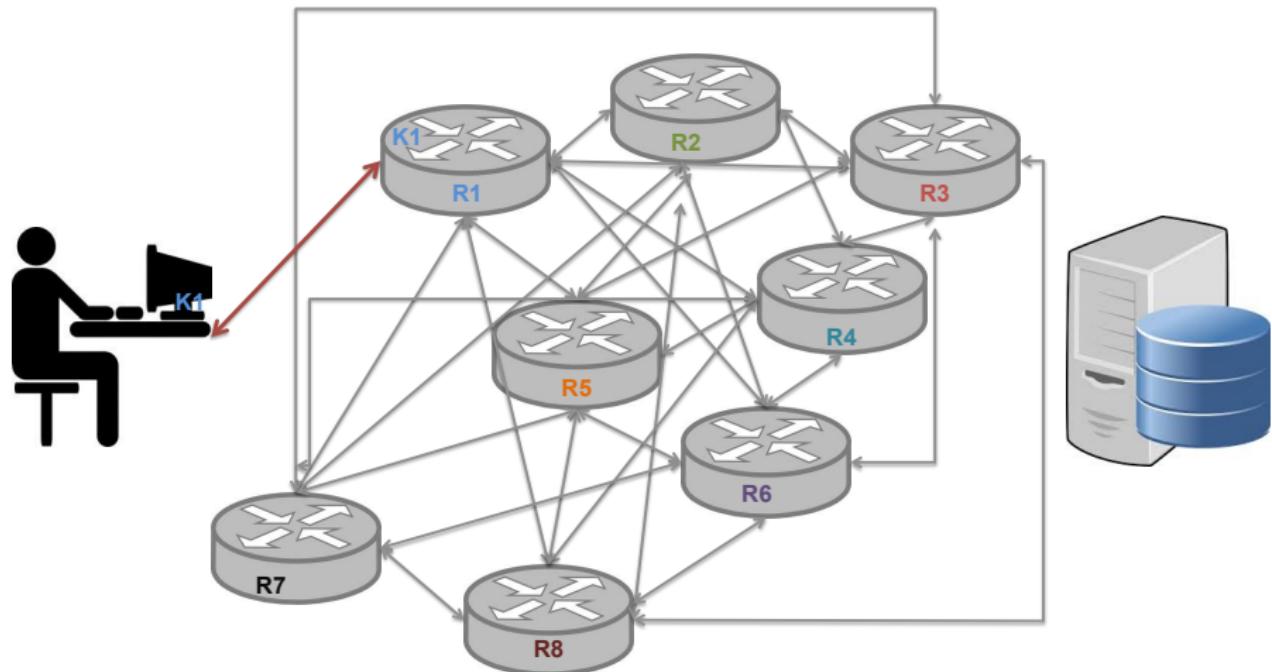
- ▶ use public-key crypto only to establish circuit
- ▶ use symmetric-key crypto to exchange data
- ▶ distribute trust like mixes

But does not defend against attackers that control the hole network

TOR circuit setup

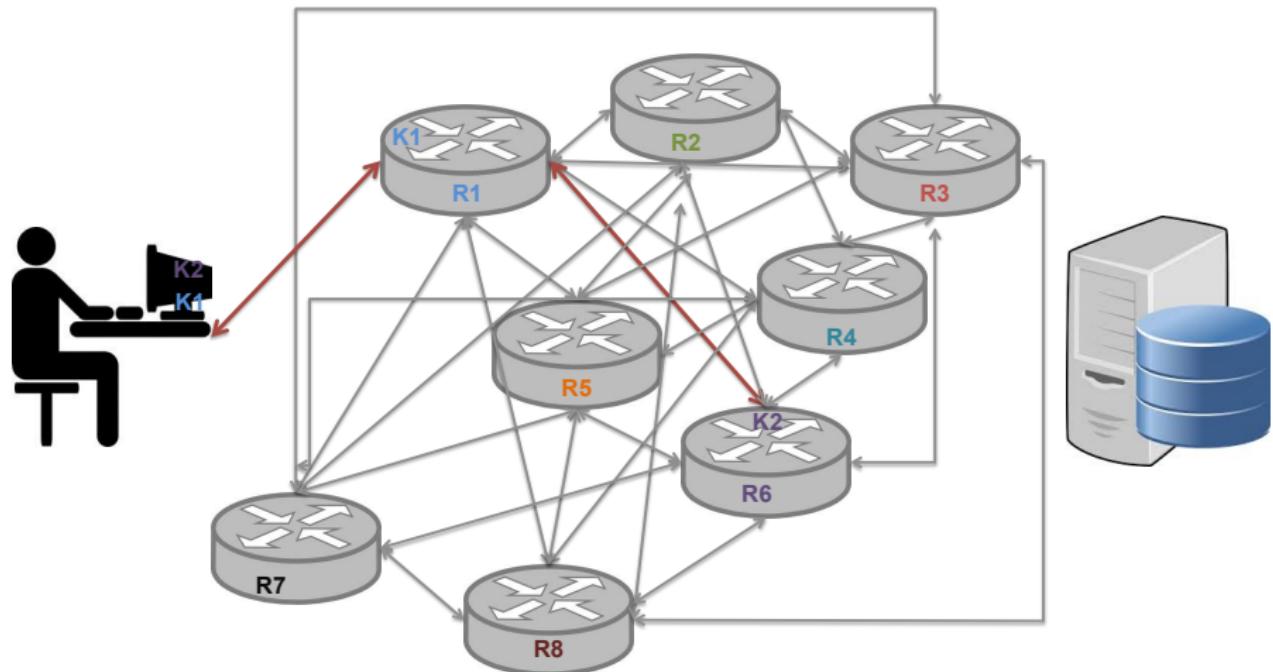


TOR circuit setup



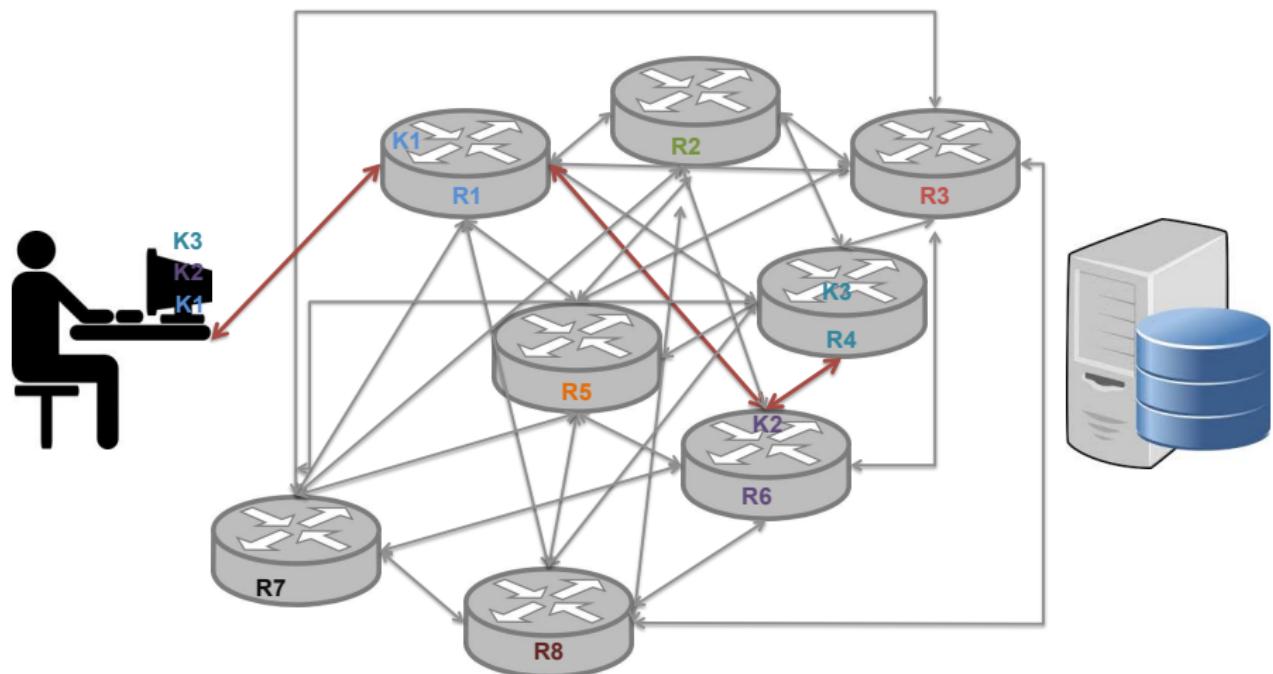
- ▶ client establishes session key **K1** and circuit with Onion Router **R1**

TOR circuit setup



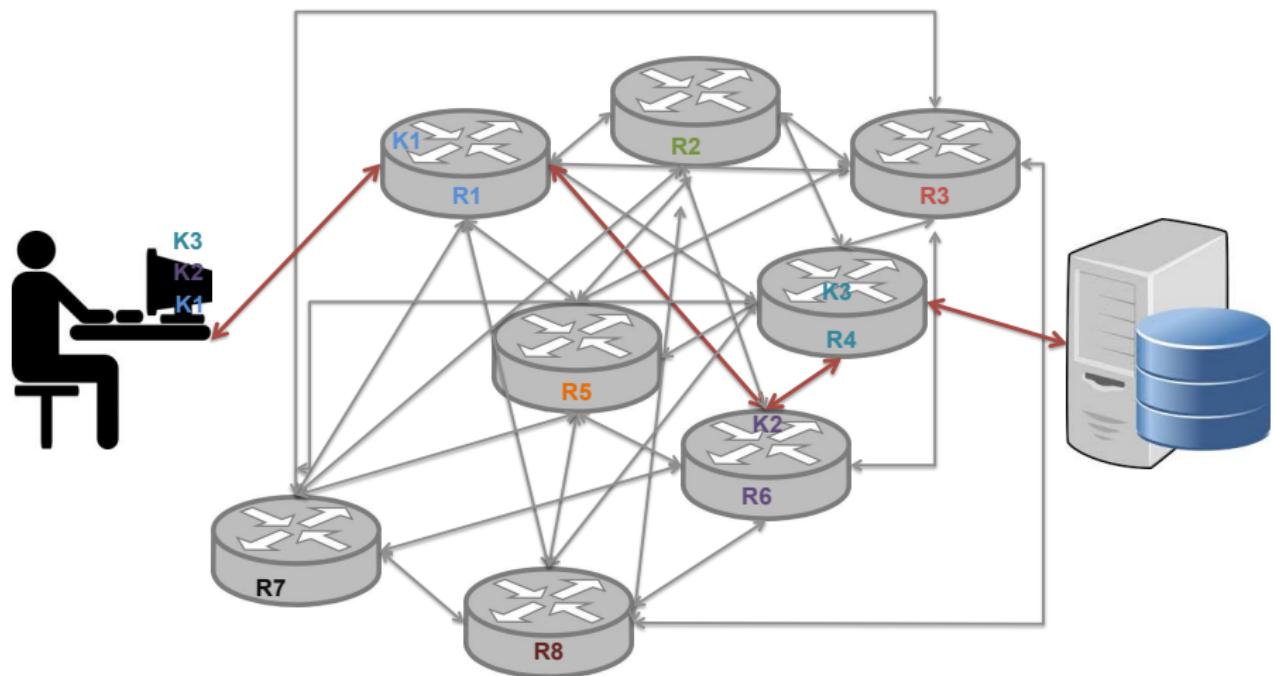
- ▶ client tunnels through that circuit to extend to Onion Router **R6**

TOR circuit setup



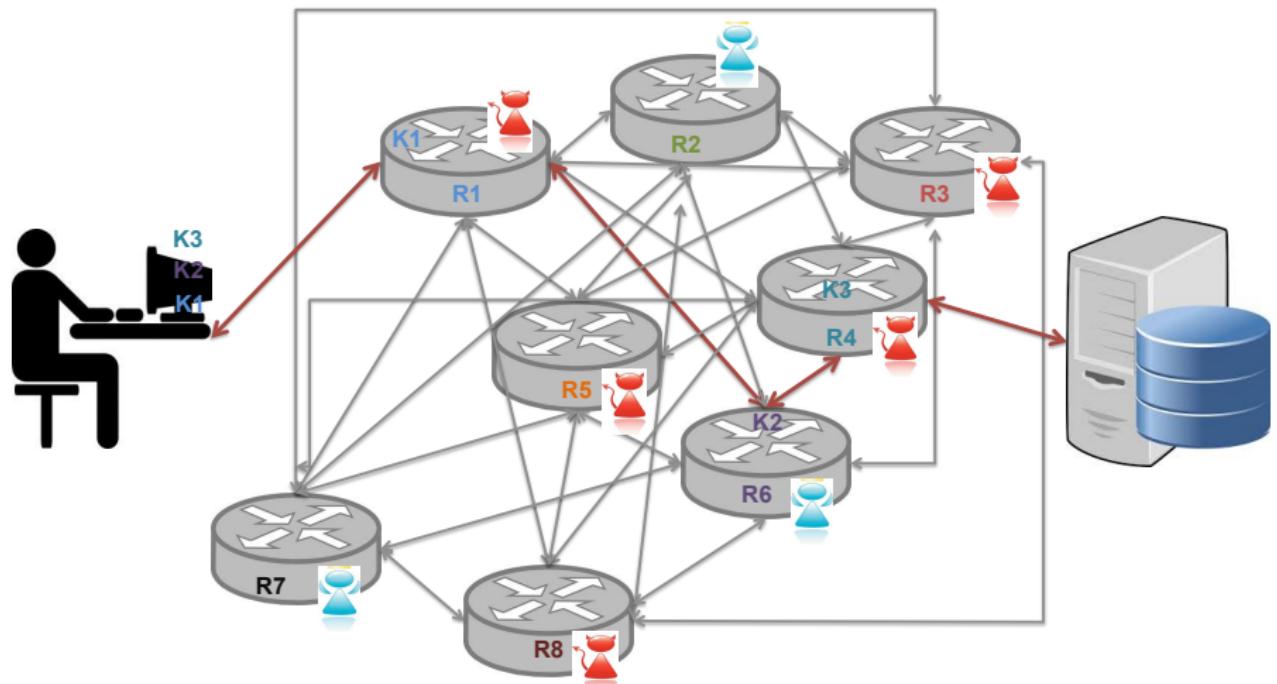
- ▶ client tunnels through that extended circuit to extend to Onion Router **R4**

TOR circuit setup



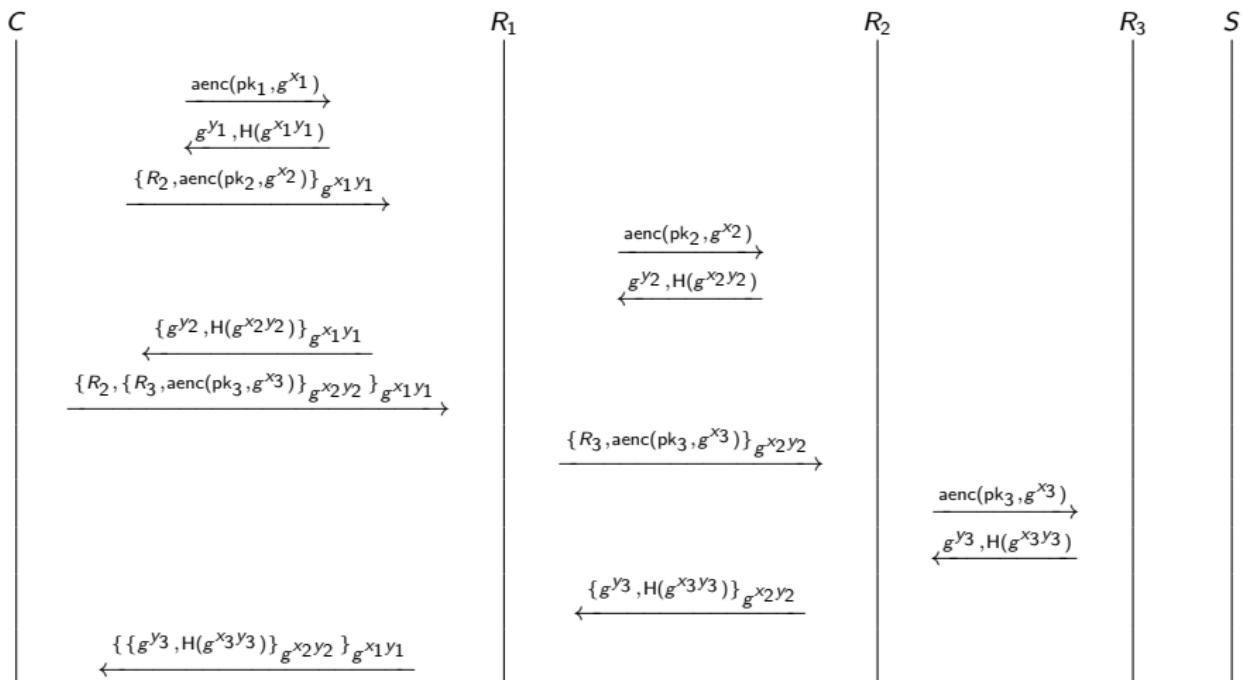
- ▶ client applications connect and communicate over established TOR circuit

TOR circuit setup

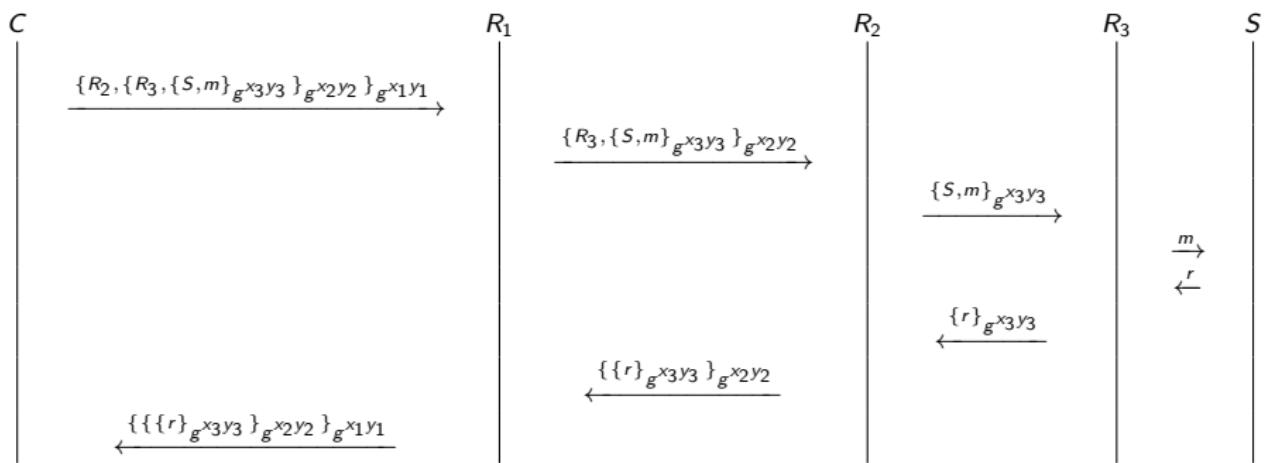


a single honest Onion Router on the TOR circuit guarantees anonymity against an attacker controlling some Onion Routers

The (simplified) TOR message flow - circuit setup



The (simplified) TOR message flow - actual communication



TOR only provides privacy - not confidentiality

- ▶ TOR anonymises the origin of the traffic
- ▶ TOR encrypts everything inside the TOR network
- ▶ but TOR **DOES NOT** encrypt all traffic through the Internet
- ▶ for confidentiality you still need to use end-to-end encryption such as SSL/TLS

TOR takes care of DNS resolution

- ▶ TOR only anonymises TCP streams
- ▶ But, DNS resolution is executed over UDP
- ▶ So, DNS resolution if handled by the client browser defeats the purpose of using TOR
- ▶ To avoid privacy breaches due to DNS resolution, the TOR browser delegates DNS resolution to the exit node

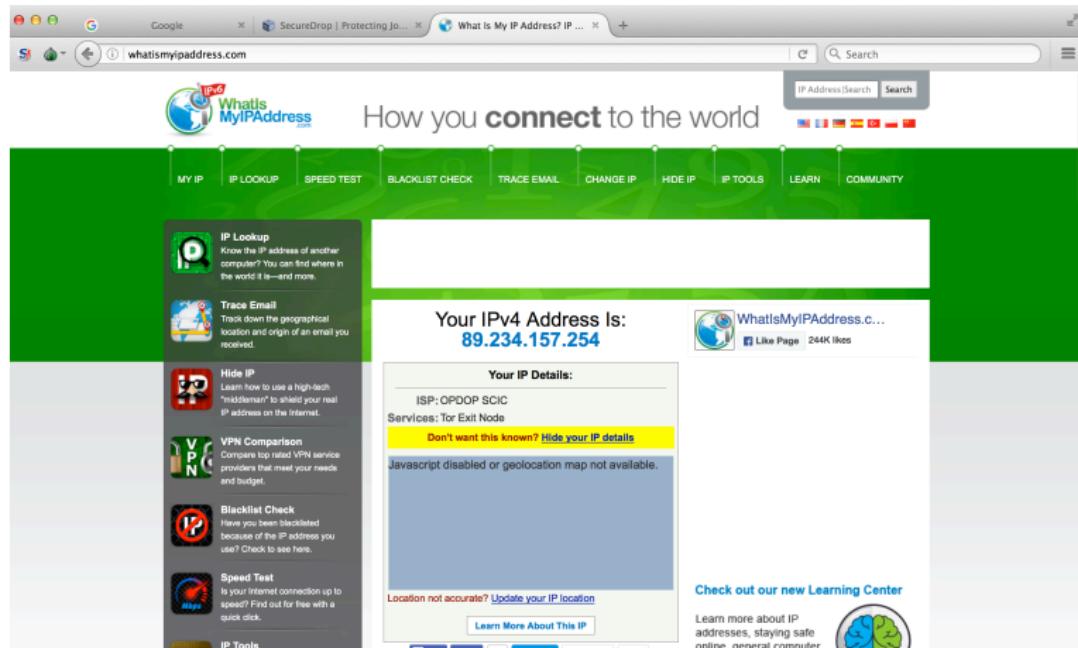
Avoiding censorship

- ▶ TOR relays are listed on the public TOR directory
- ▶ So your local ISP can observe that you are communicating with TOR nodes
- ▶ ISPs and governments can try to block access to the TOR network by blocking TOR relays
- ▶ TOR bridge relays are relays not listed on the public TOR directory
- ▶ Entering the TOR network through a TOR bridge relay can prevent ISPs and governments blocking access to the TOR network

Limitations of TOR

- ▶ TOR does not provide protection against end-to-end timing attacks
- ▶ If the attacker can see both ends of the communication channel, he can correlate volume and timing information on the two sides

whatismyipaddress.com cannot tell where am I using TOR



The screenshot shows a web browser window with the URL whatismyipaddress.com in the address bar. The page itself has a green header with the text "How you **connect** to the world". Below the header is a navigation menu with links: MY IP, IP LOOKUP, SPEED TEST, BLACKLIST CHECK, TRACE EMAIL, CHANGE IP, HIDE IP, IP TOOLS, LEARN, and COMMUNITY. On the left side, there is a sidebar with several icons and their descriptions: IP Lookup (Know the IP address of another computer? You can find where in the world it is—end where), Trace Email (Track down the geographical location and origin of an email you received), Hide IP (Learn how to use a high-tech "hiddenman" to shield your real IP address on the Internet), VPN Comparison (Compare top rated VPN service providers that meet your needs and budget), Blacklist Check (Have you been blacklisted because of the IP address you use? Check to see here), Speed Test (Is your Internet connection up to speed? Find out for free with a quick click), and IP Tools. The main content area displays the text "Your IPv4 Address Is: 89.234.157.254". Below this, there is a section titled "Your IP Details:" which includes the ISP (OPDOP SCIC) and Services (Tor Exit Node). A yellow button labeled "Don't want this known? Hide your IP details" is present. Further down, a message states "Javascript disabled or geolocation map not available." At the bottom of the main content area, there is a link "Location not accurate? Update your IP location" and a "Learn More About This IP" button. To the right of the main content area, there is a sidebar with the text "Check out our new Learning Center" and a brain icon. At the very bottom of the page, there is some footer text and small icons.

google.com thinks I'm in the Netherlands using TOR

A screenshot of a Mac OS X desktop showing a Tor Browser window. The title bar says "Tor Browser". The main content area shows a Google search results page for "What Is My IP Address?". A context menu is open over the search bar, with the "Tor circuit for this site" option selected. A tooltip from this option reads: "To track you. We recommend that you leave Tor Browser windows in their original default size." Below the menu, a dropdown shows the current circuit: "This browser", "United Kingdom (163.172.21.117)", "France (91.121.23.100)", "Netherlands (46.166.148.177)", and "Internet". The main search results page features the classic Google logo with a play button icon.

New Identity ⌘U
New Tor Circuit for this Site ⌘L

Tor circuit for this site
(google.de):

- This browser
- United Kingdom (163.172.21.117)
- France (91.121.23.100)
- Netherlands (46.166.148.177)
- Internet

OK X

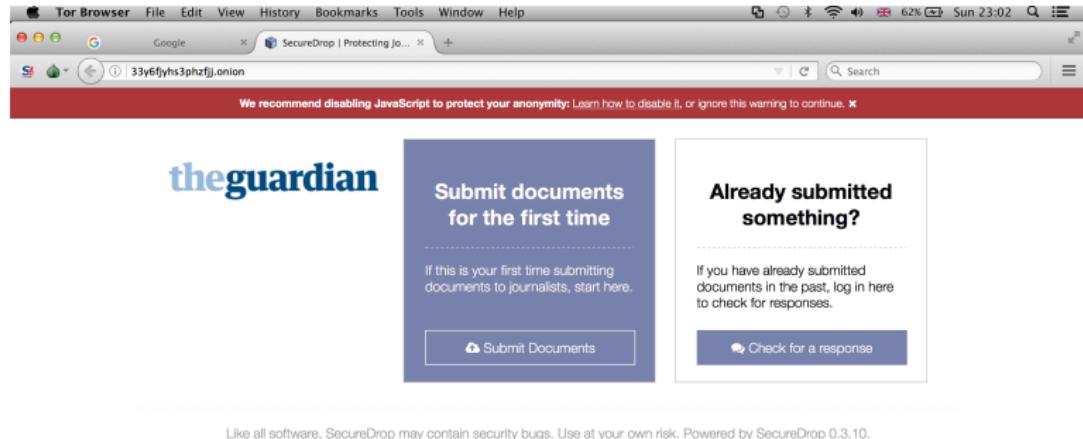
Google-Suche Auf gut Glück!

Google.de angeboten auf: English

SPÄTER ERINNERN JETZT ANSEHEN

Werbeprogramme Unternehmen Über Google Datenschutzerklärung Nutzungsbedingungen Einstellungen

TOR hidden services



- ▶ TOR can also provide anonymity to websites and servers
- ▶ www.torproject.org/docs/hidden-services.html