DOTA2024:5
Defense of the Ancients
Fifth topic - IP networks

Hugh Anderson

National University of Singapore
School of Computing

July, 2024

IP networks...

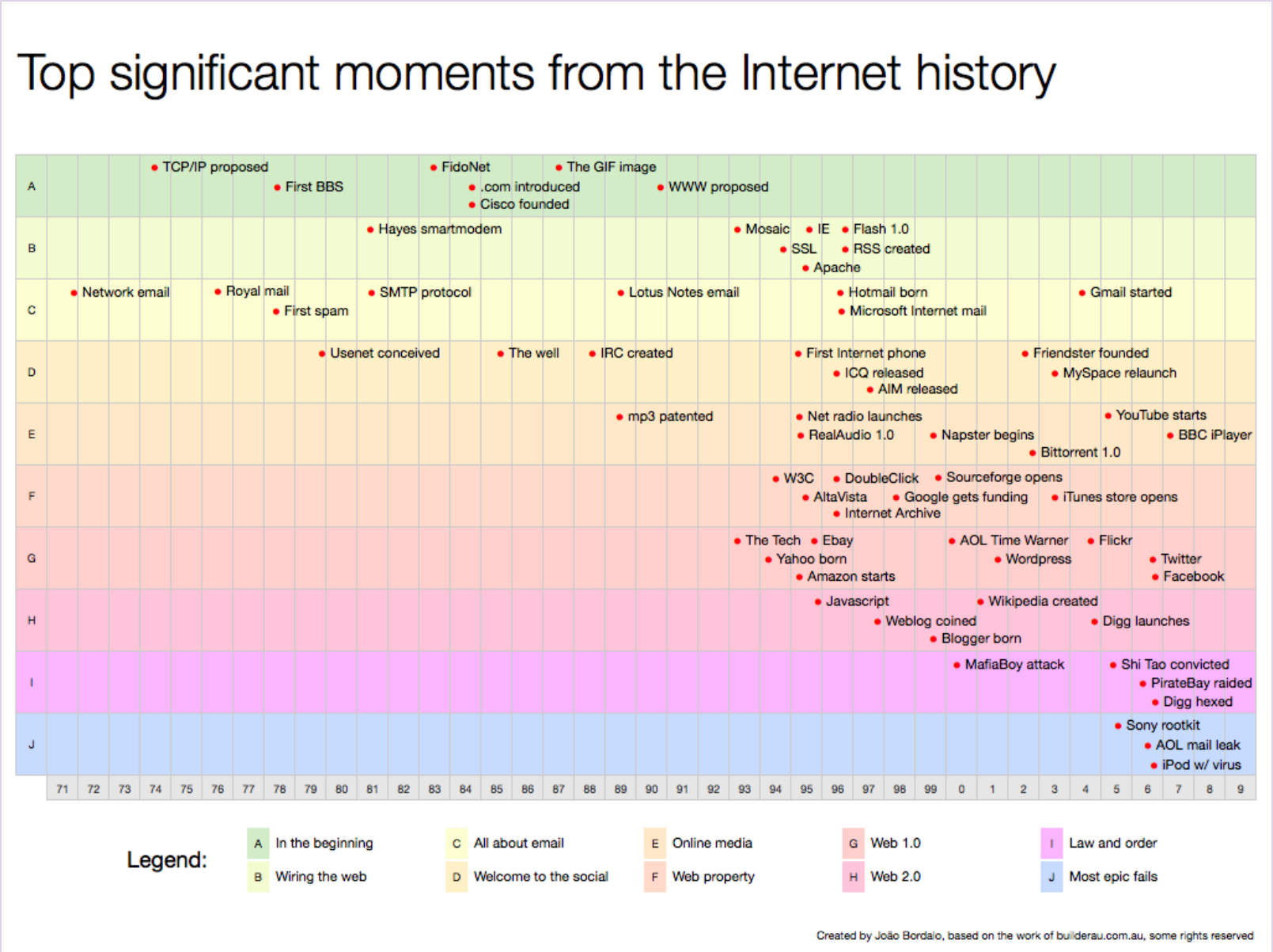
Outline

Outline

Outline

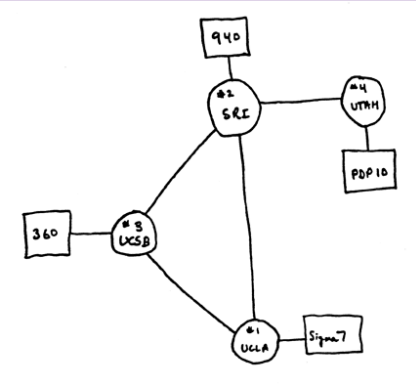
Internet history

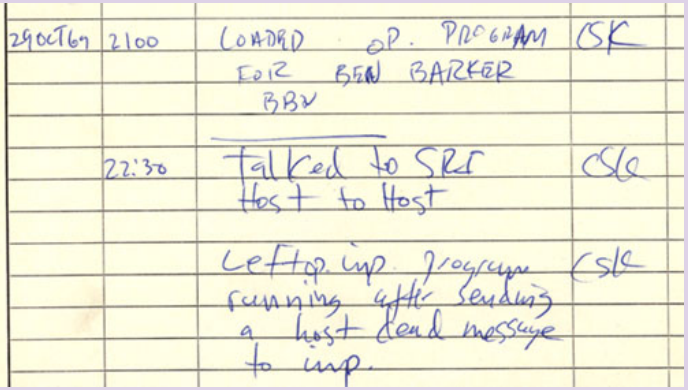
Moments in time...



Objects in space...

The first network, the first router, the first message...

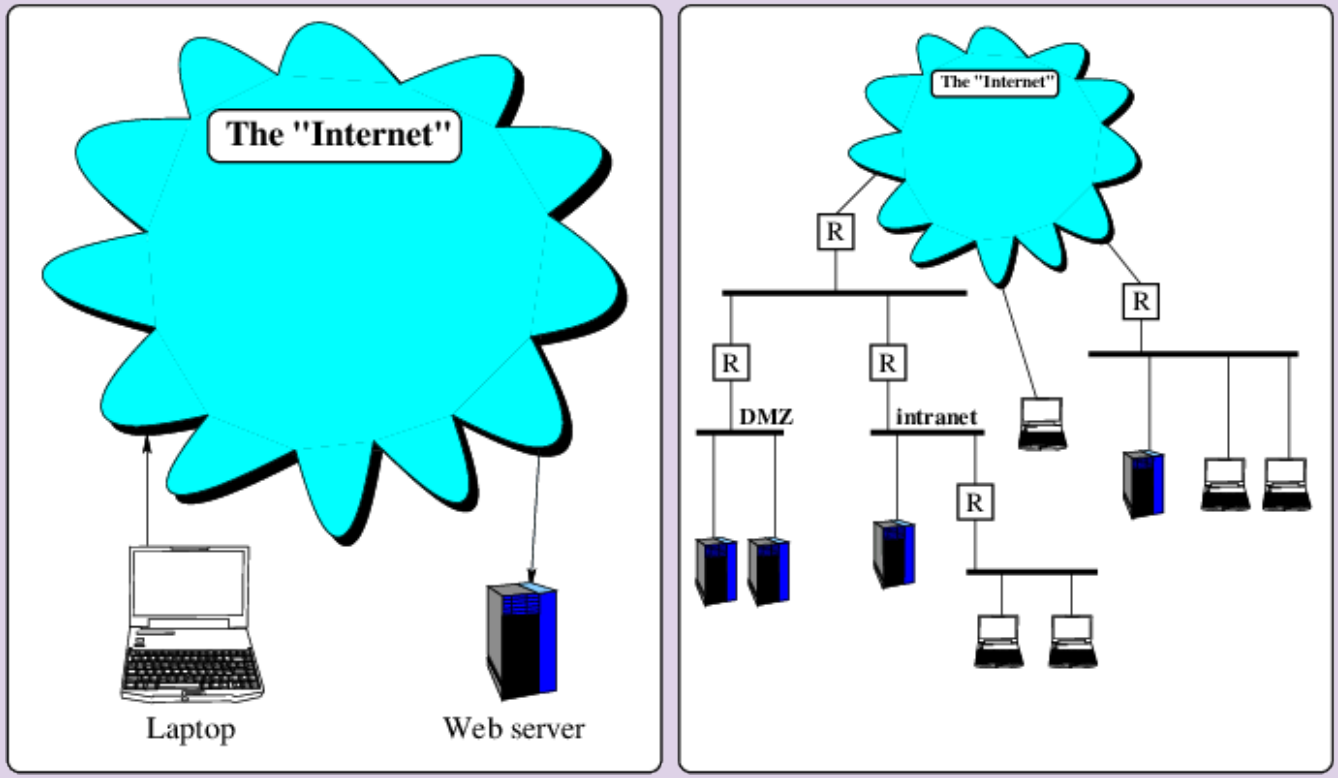




- The original network diagramming tool was a pencil, and the first network had four routers (originally called IMPs), and four nodes (At UCSB, UCLA, SRI and University of Utah).
- The first successful transmission was from UCLA to SRI in September 1969, and was recorded in the UCLA logbook.

Structure of networks now...

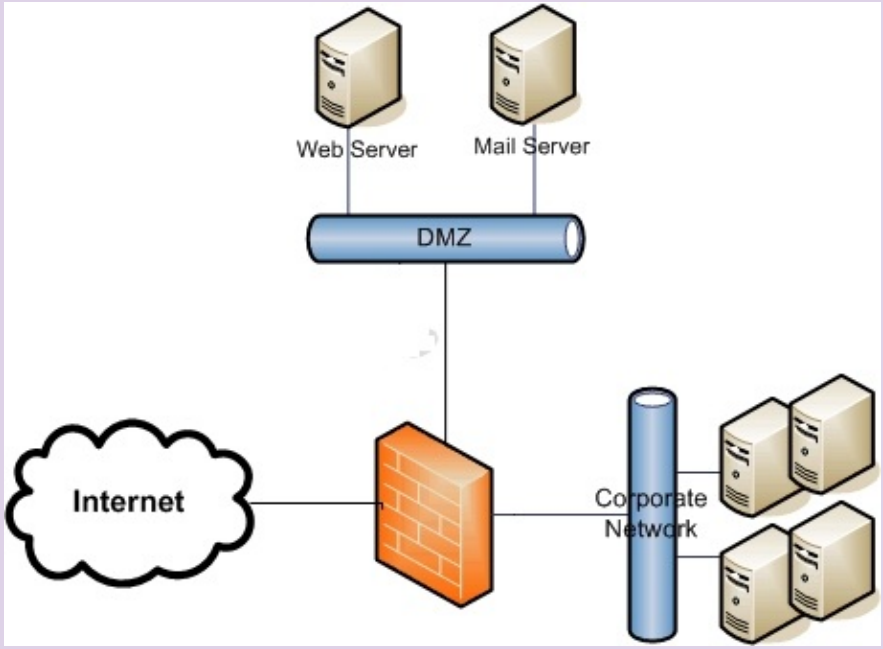
Trees and lattices...



The topology of the Internet, particularly nearer the 'edge' of the Internet, appears more like a tree than a lattice, and we use the routers to control access to and from the smaller local networks.

Firewalls

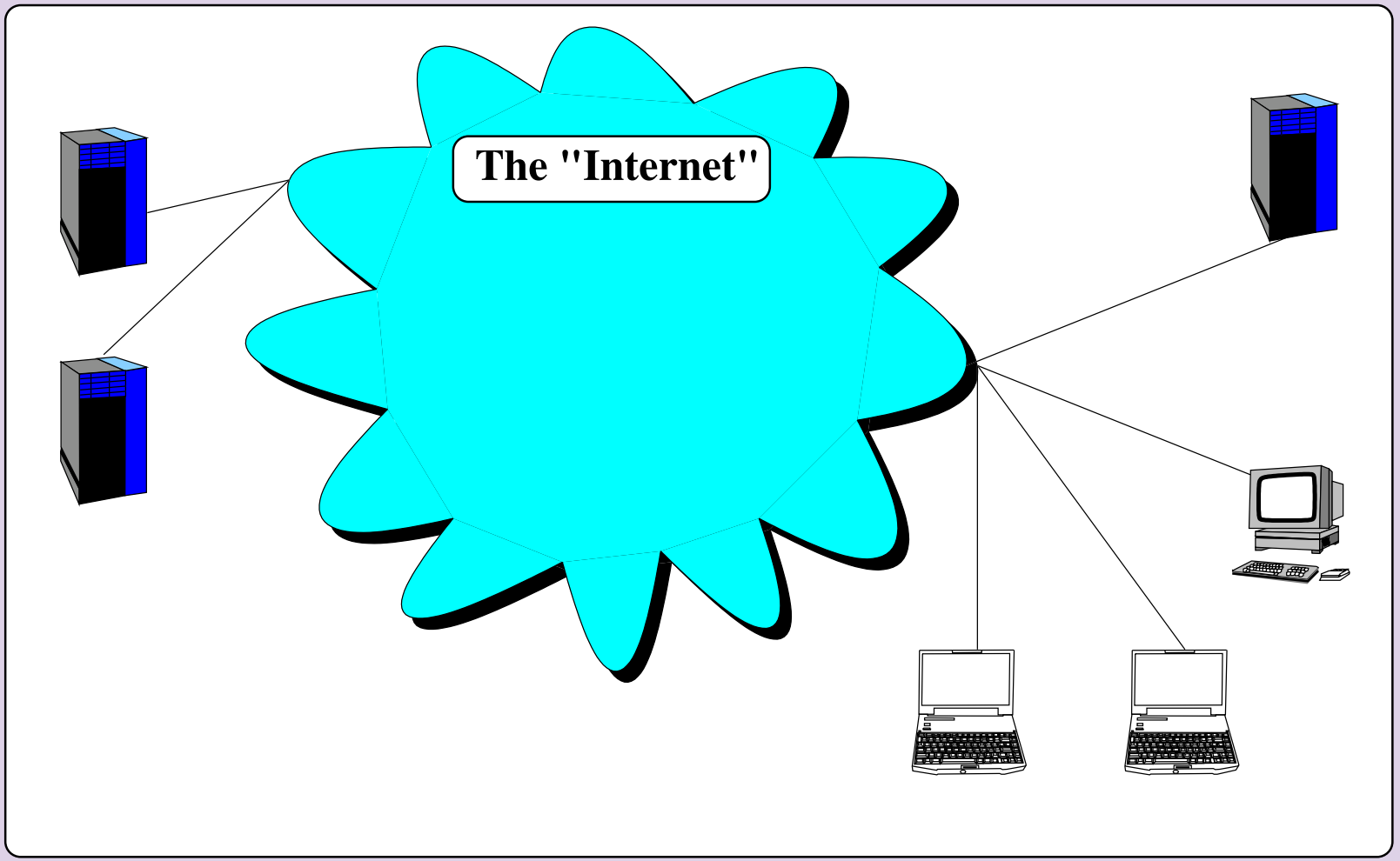
When a router is protecting you...



Firewalls are brick walls often found in wooden buildings and are supposed to prevent the spread of fire. In networking we use the same idea - the firewall is a router, which limits access to and from the Internet. We normally imagine that the fire is on the Internet side :)

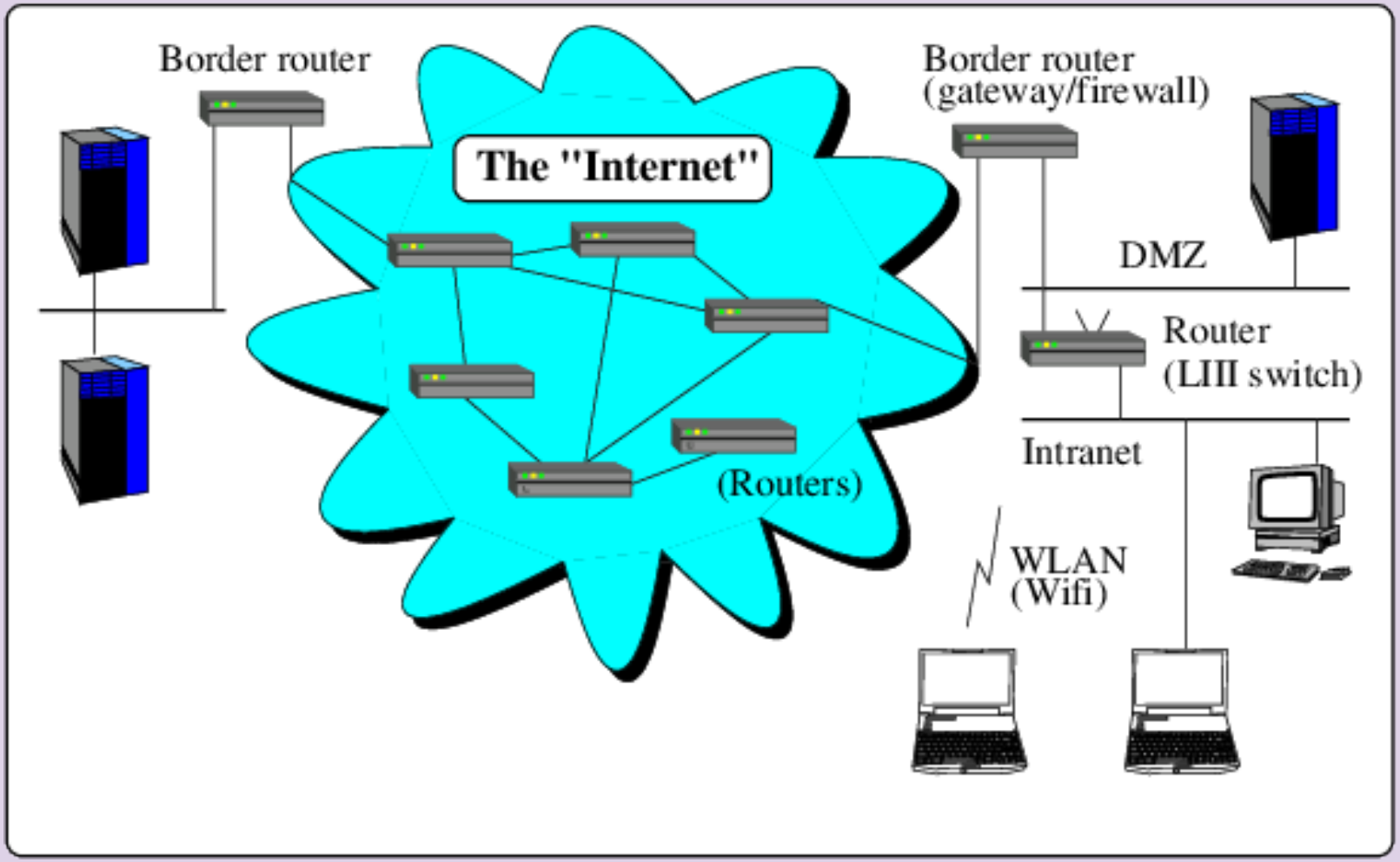
Systems, networks...

We all trust the Internet, but who manages it again?



New topic: Systems, networks...

Consider the Internet - lots of room for damage



Internet attack possibilities

One of, or some combination of...

Close to you...

1. Your WiFi link(s)
2. Your PCs
3. Servers local to your organization (file, name servers, etc)
4. Your router(s)

But far away too...

1. Routers on the Internet
2. Remote servers that you trust: For example:
3. DNS for names like gmail.com
4. ebay, Amazon, Sony PSN, your bank's web site...

Who does all of this? Anonymous people with (a) a computer, and (b) an Internet connection. Not too hard.

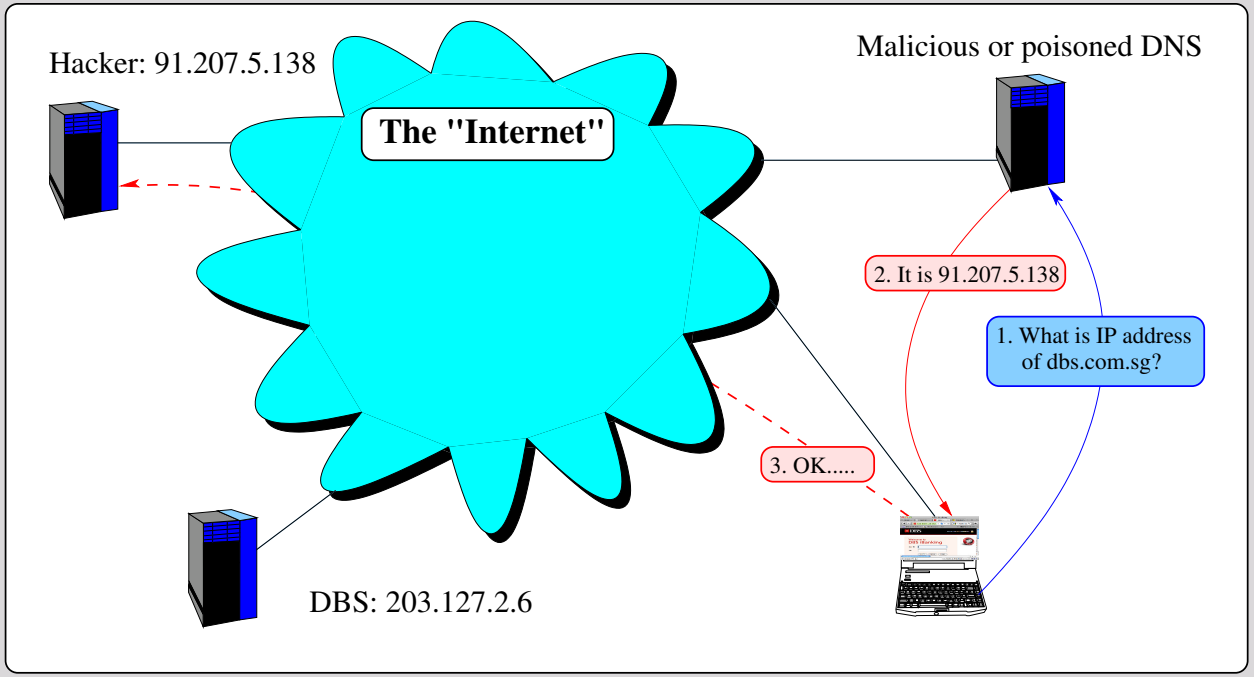
Internet hacking scenario #1

A server you trust lies to you...

You connect to dbs.com.sg to do some banking.

But... A local DNS (Domain name system) server lies to your computer, and connects you to hackers.r.us who have put up a spoof version of the bank's web site.

The spoof version collects your password and login information and proceeds to empty your account.

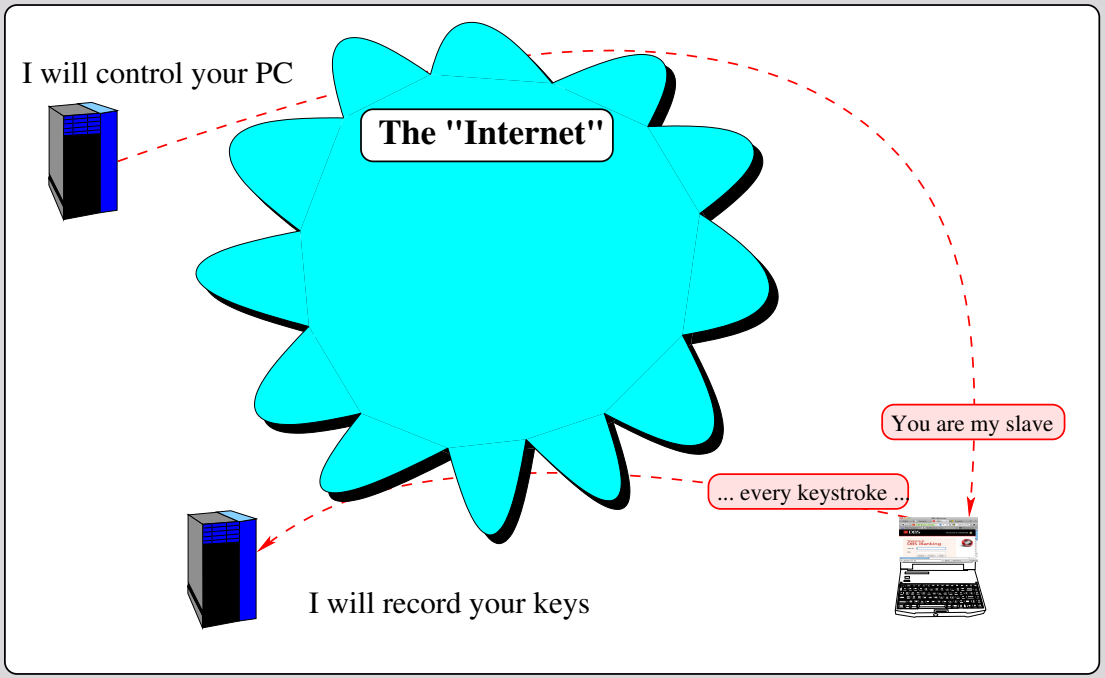


Internet hacking scenario #2

A server you trust lies to you...

A web site fastdownloads.org.uk says it will improve your Internet experience by 225%! You say yes, and it installs a *useful* bit of software.

Actually - it installs something which sends every key you type to hackers.r.us, and/or allows someone in Guatemala to control your machine...

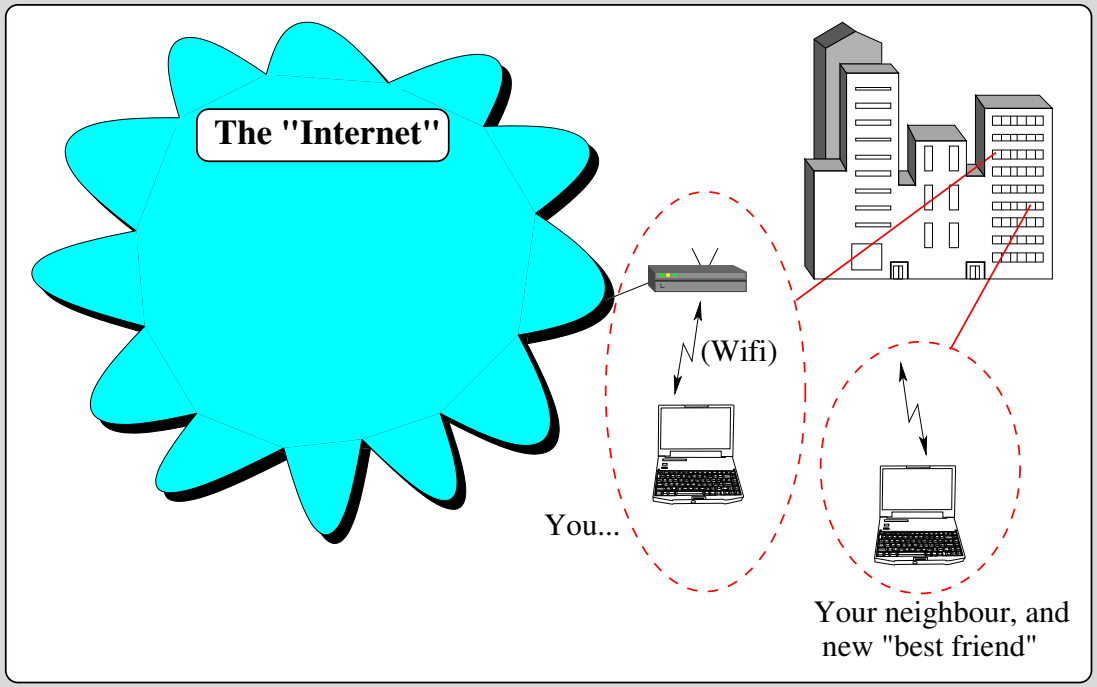


Internet hacking scenario #3

You overestimate how safe your home is...

Your next door neighbour observes your wifi network.

Your neighbour hacks your network authentication password, and from then on, uses your Internet link, and monitors your traffic to and from the Internet, retrieving your passwords etc...

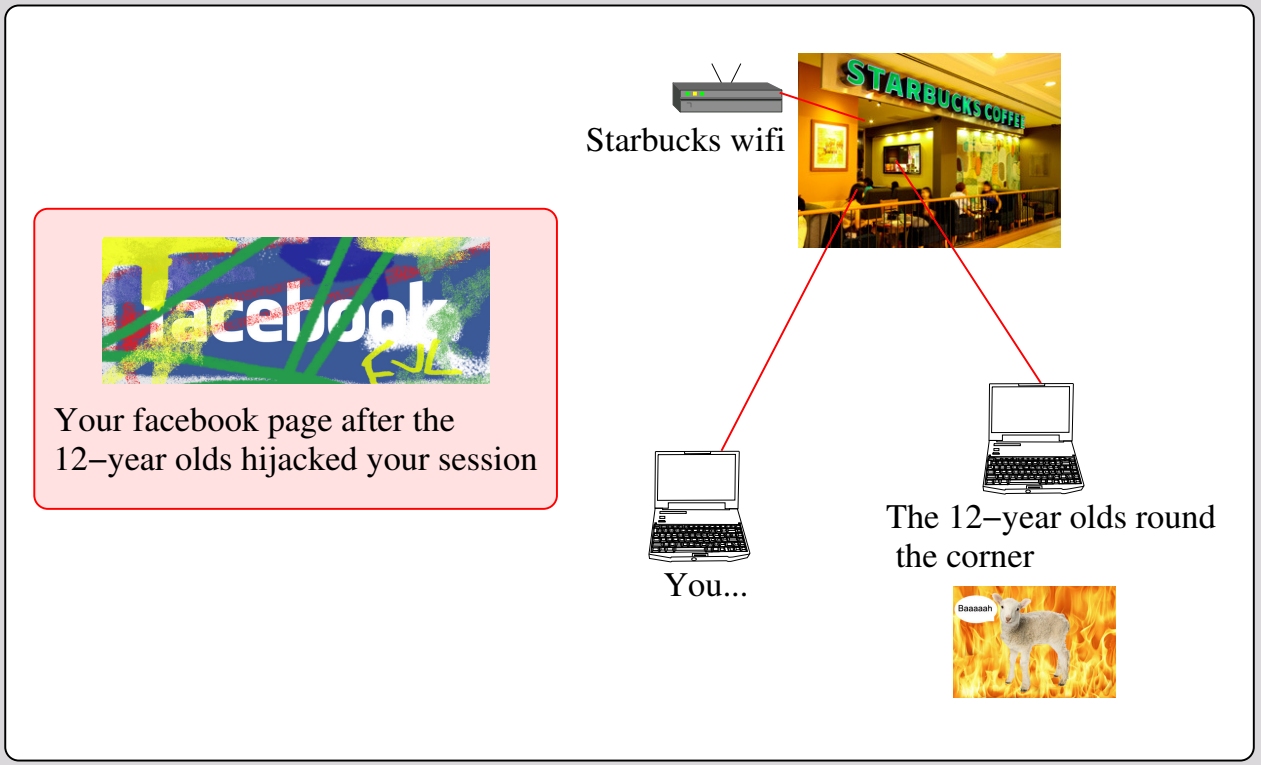


Internet hacking scenario #4

You overestimate how safe Starbucks is...

You look at facebook to see whats happening.

A cluster of 12 yr-olds there are using firesheep, and steal a cookie and take over your facebook session, posting all sorts of “funny” stuff[a].



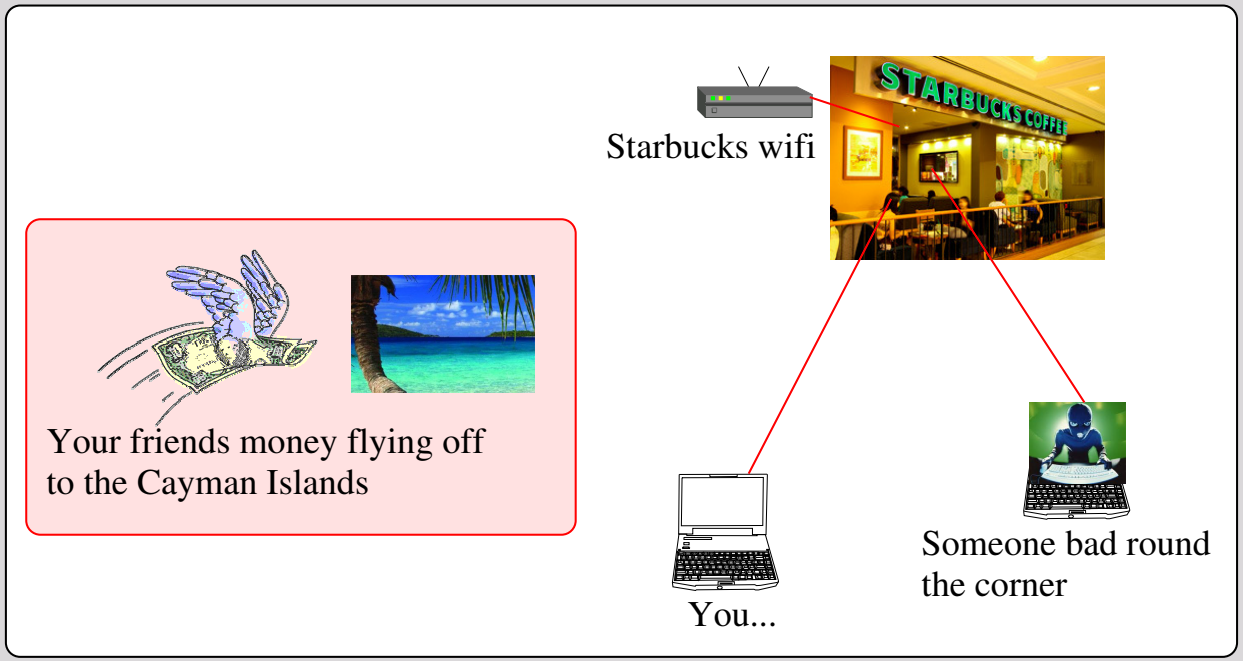
[a]This led to facebook changing to https-by-default two years ago

Internet hacking scenario #5

You overestimate how safe Starbucks is...

Back at the coffee house, you look at gmail.

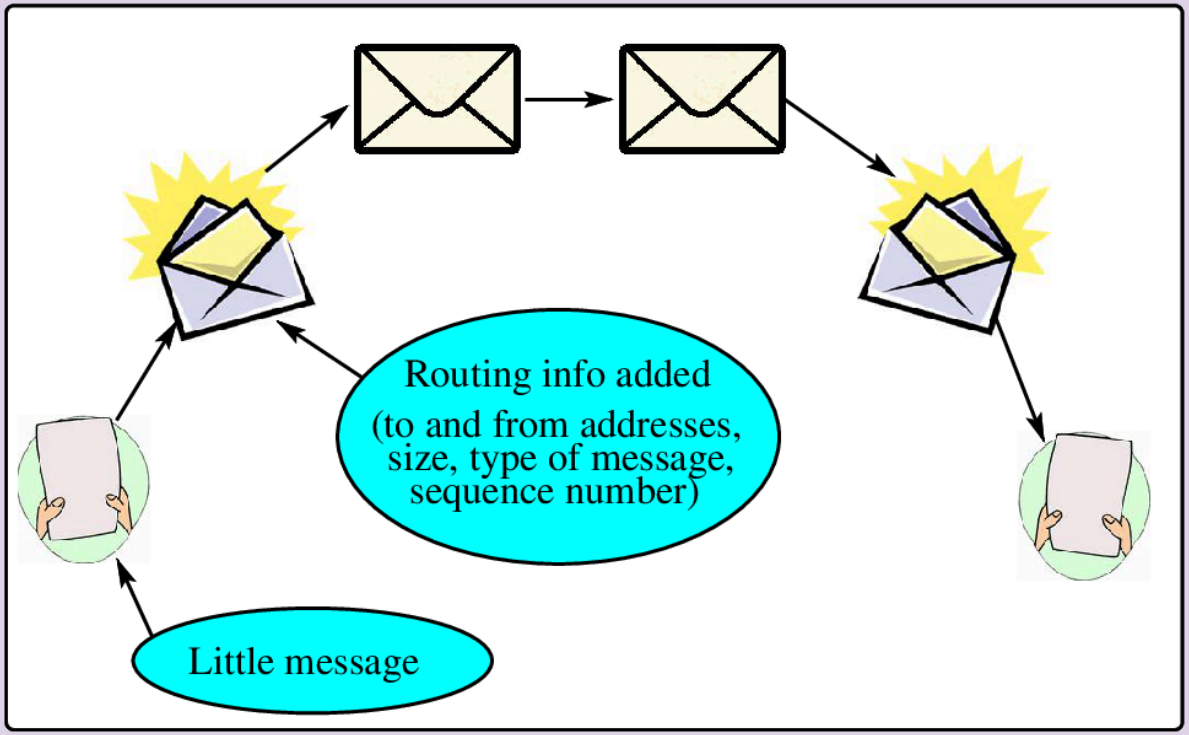
Over the next three days, a bunch of your friends and family wire large amounts of money to an account in the Cayman Islands, to "*help you out*". The money goes to fund someone's next holiday (in the Cayman Islands! Spooky eh?)



Outline

Internet basics: routing and packets...

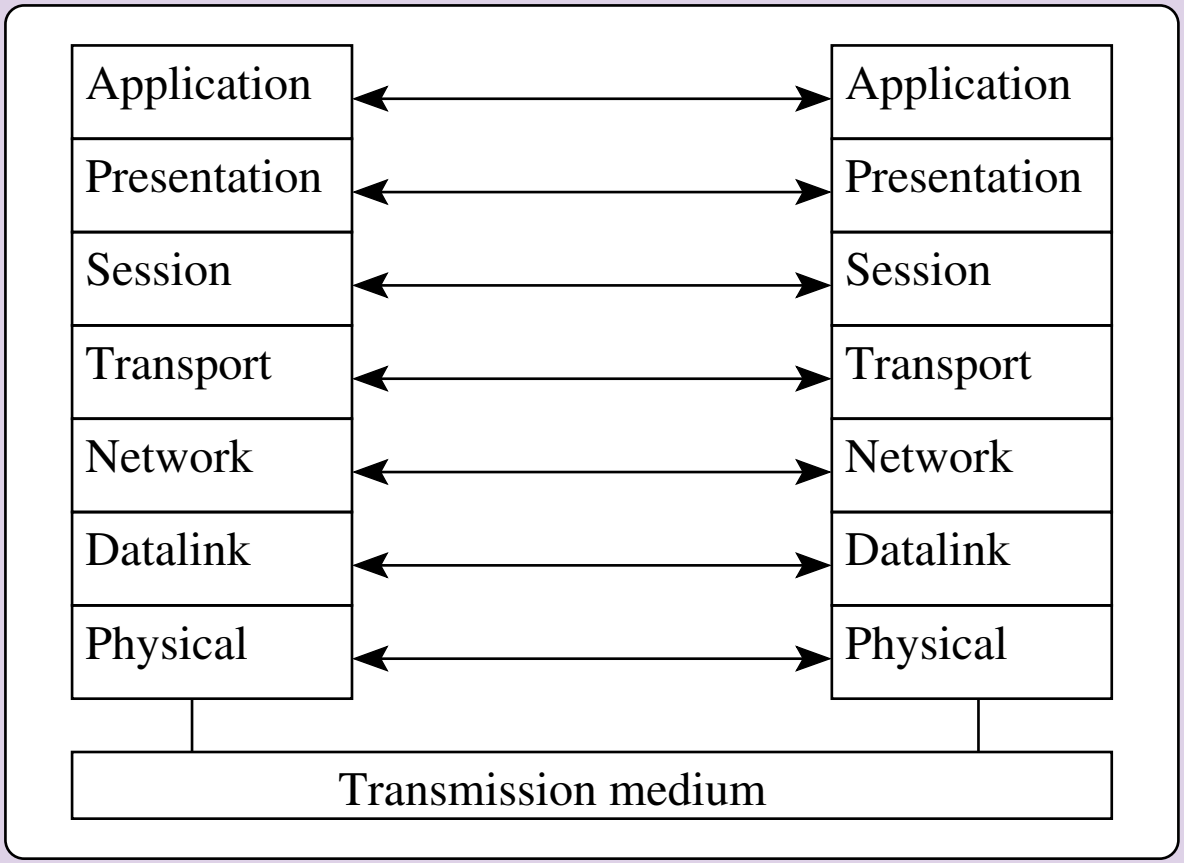
Internet traffic sent in packets...



Lots of opportunity to modify routing information, spoof etc

Looking at computer communication

The OSI reference model: levels of interest...



Looking at computer communication

The OSI reference model: levels of interest...

This model provides a framework to hang your understanding of computer networking on.

The protocol specifies how communication is performed with the layer at the same level in the other computer. This does not for instance mean that the network layer directly communicates with the network layer in the other computer, instead each layer uses the services provided by the layer below it, and provides services to the layer above it.
The definition of services between adjacent layers is called an interface.

What are the advantages of the layered structure?

1. We can change layers as needed, the interface remains constant.
2. The networking software is easier to write, as it has been modularized
3. The client software is simpler, as it only needs to know the interface.

Note that a *protocol* specifies the interaction between peer layers. An *interface* specifies the interactions between adjacent layers.

Looking at computer communication

The OSI reference model: levels/layers of interest...

Application layer: The application layer provides the user interface to the network wide services provided. For example mail.

Presentation layer: Preserve meaning: endianness of numbers.

Session layer: manage a continuous connection.

Transport layer: ensures a network independent interface for the session layer. Five "Classes of service".

Network layer: Network routing and addressing.

Datalink layer: constructs frames, checks/retransmits on error.

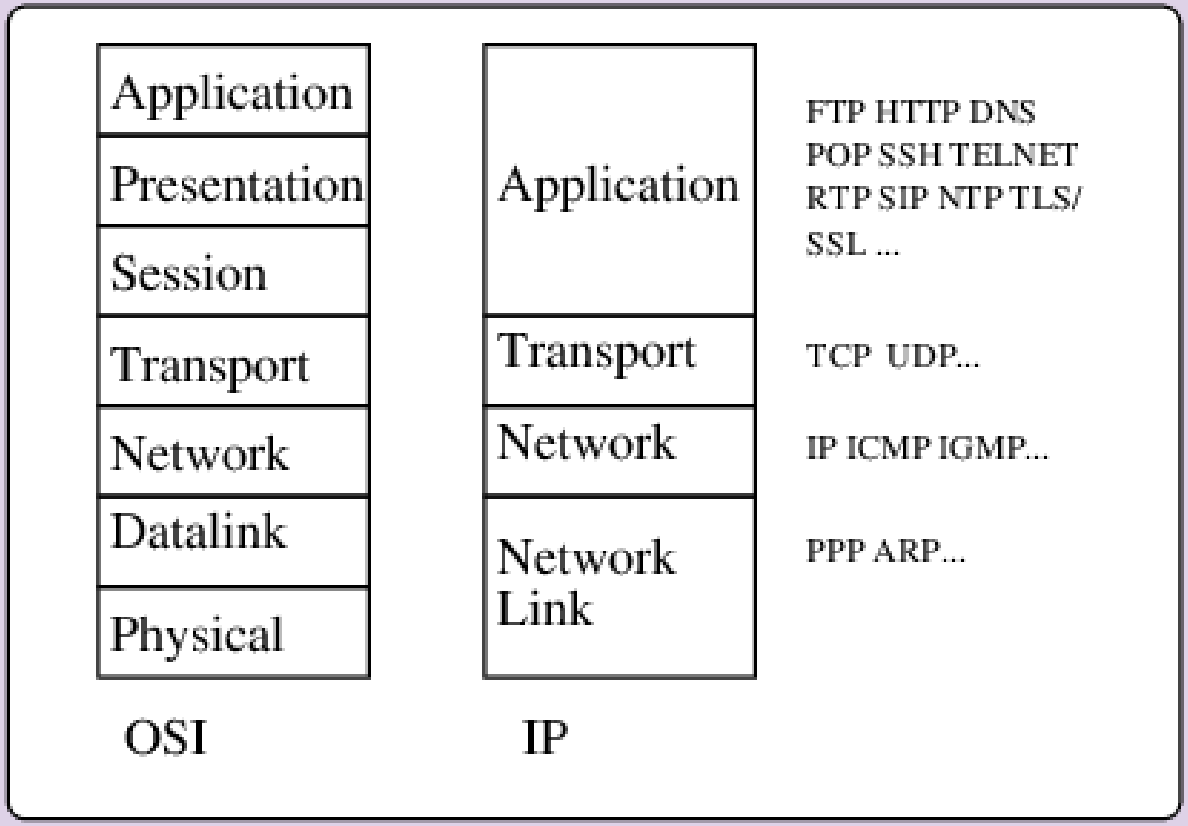
Physical layer: connectors, cables, voltage levels etc.

The Internet does not use the layered model. It only has four layers:

1. IP *application* layer: the presentation, session and application layers.
2. IP *transport* layer: the transport and session layers.
3. IP *network* layer: the same role as in the OSIRM.
4. IP *network-access* or *link* layer: the datalink and physical layers.

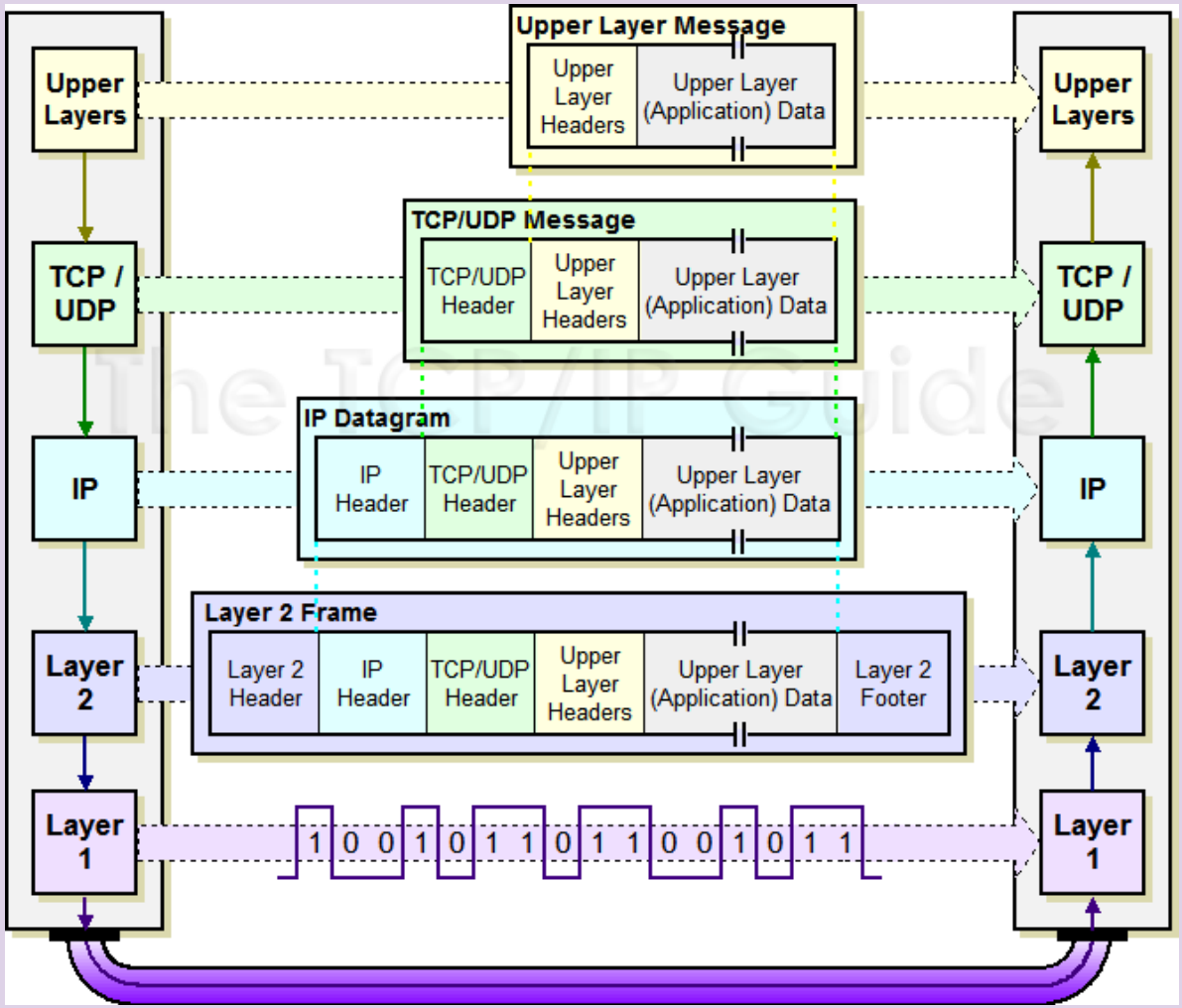
Looking at computer communication

The OSI reference model and the IP reference model...



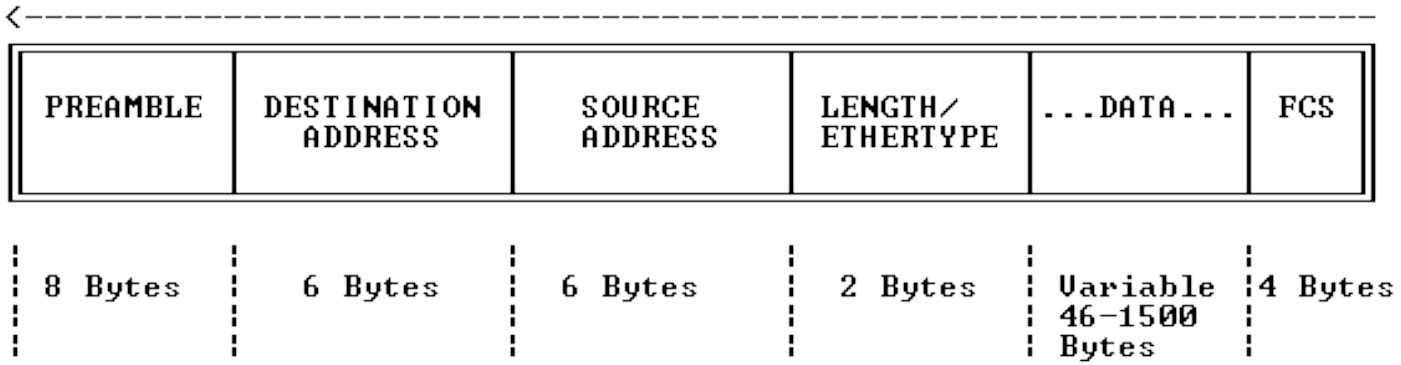
Packet (detail) on the Internet

Each layer encapsulates the previous one...



The ethernet frame

Layer 2 header and footer (detail)...



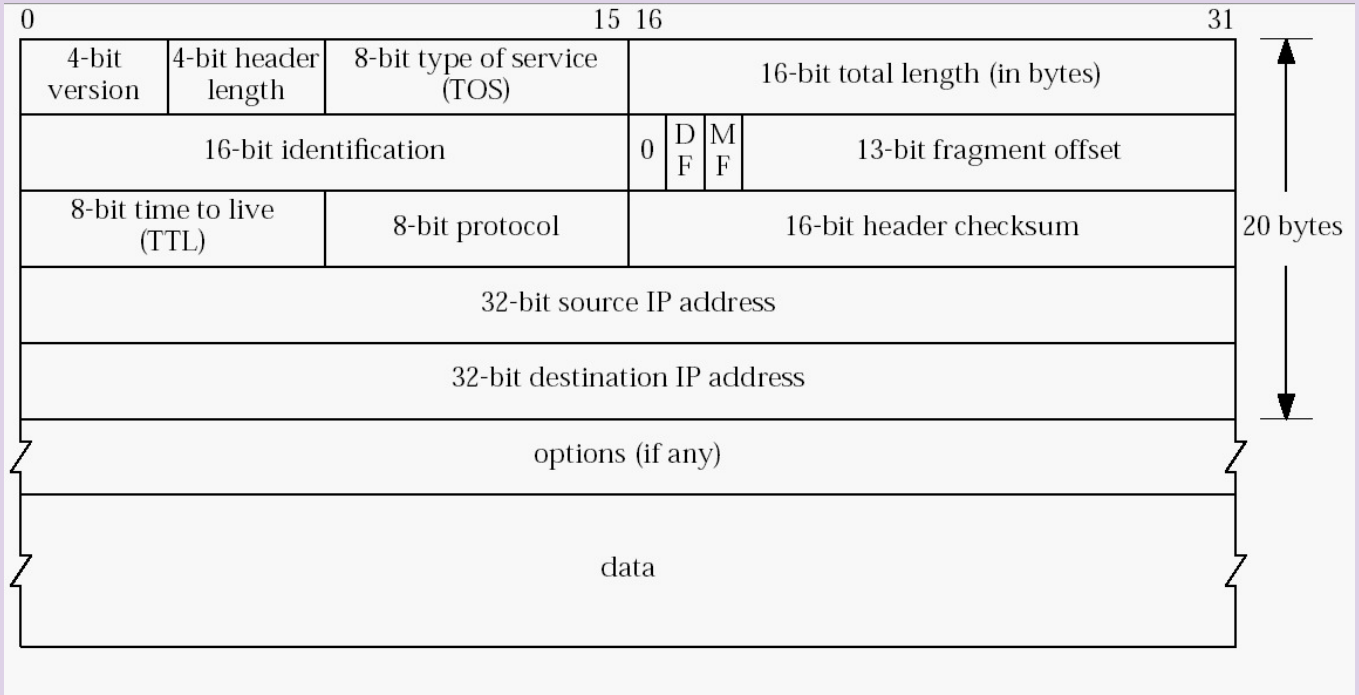
The preamble is needed so that receivers can synchronize.

The addresses are needed so machines on a shared network can respond to their own address.

The length is needed so we know how much data to expect. Note that the FCS at the end is a CRC!

Looking at a packet

The IP header (detail)...

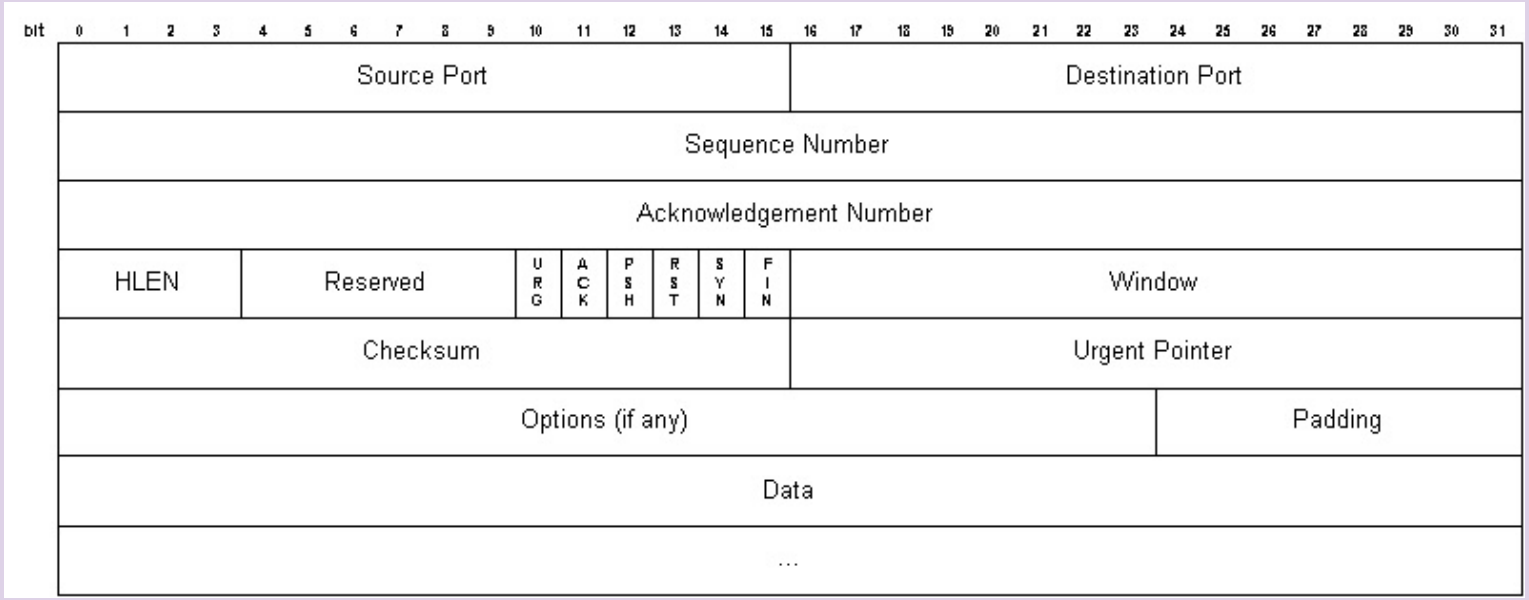


The header has its own length, and the length of the whole packet.

Other fields are type identifiers, fragment ID of a fragmented packet, time-to-live, protocol, checksum, and source and destination “IP” (32 bit) addresses.

Looking at a packet

The TCP layer (detail)...



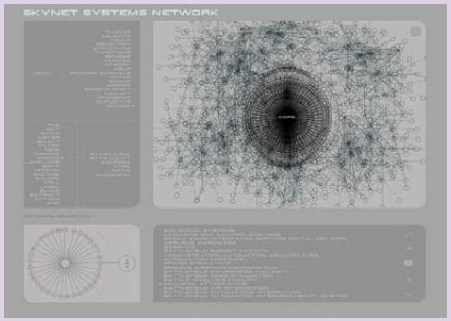
In the TCP layer, each packet has port addresses, and sequence numbers, various type identifiers, and a header (ones-complement) checksum

Summary

It just got so much worse...

Communication between computers comes in many flavours of *protocol*, and at different levels.
Not only multiple machines that could cause problems, but also multiple protocols at multiple levels.

No administrative centre, just general agreement to use the RFCs, not much in the way of police!

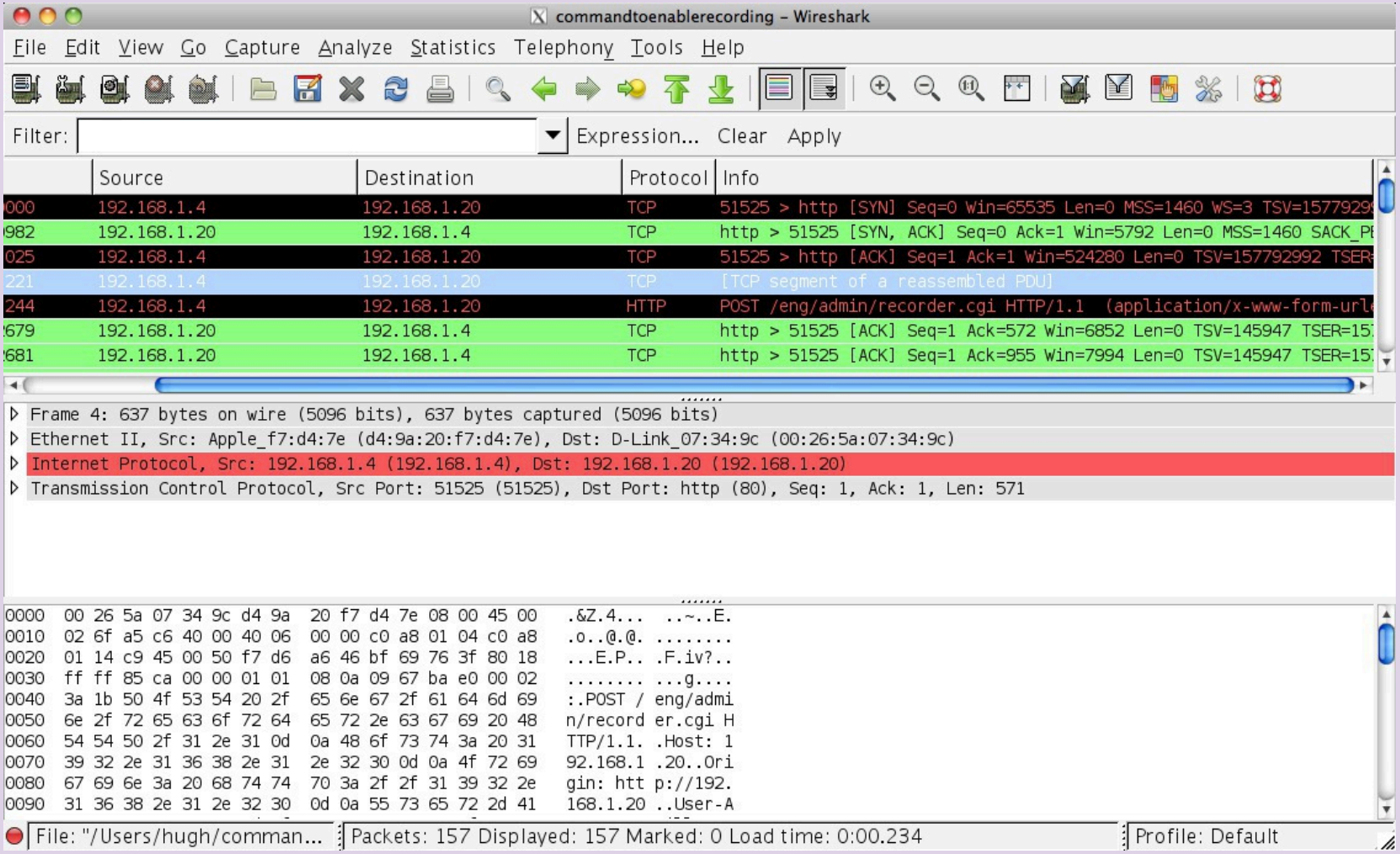


It seems like we cannot trust the protocols or the machines!

Outline

Looking at a packet

Wireshark displaying an ethernet frame...



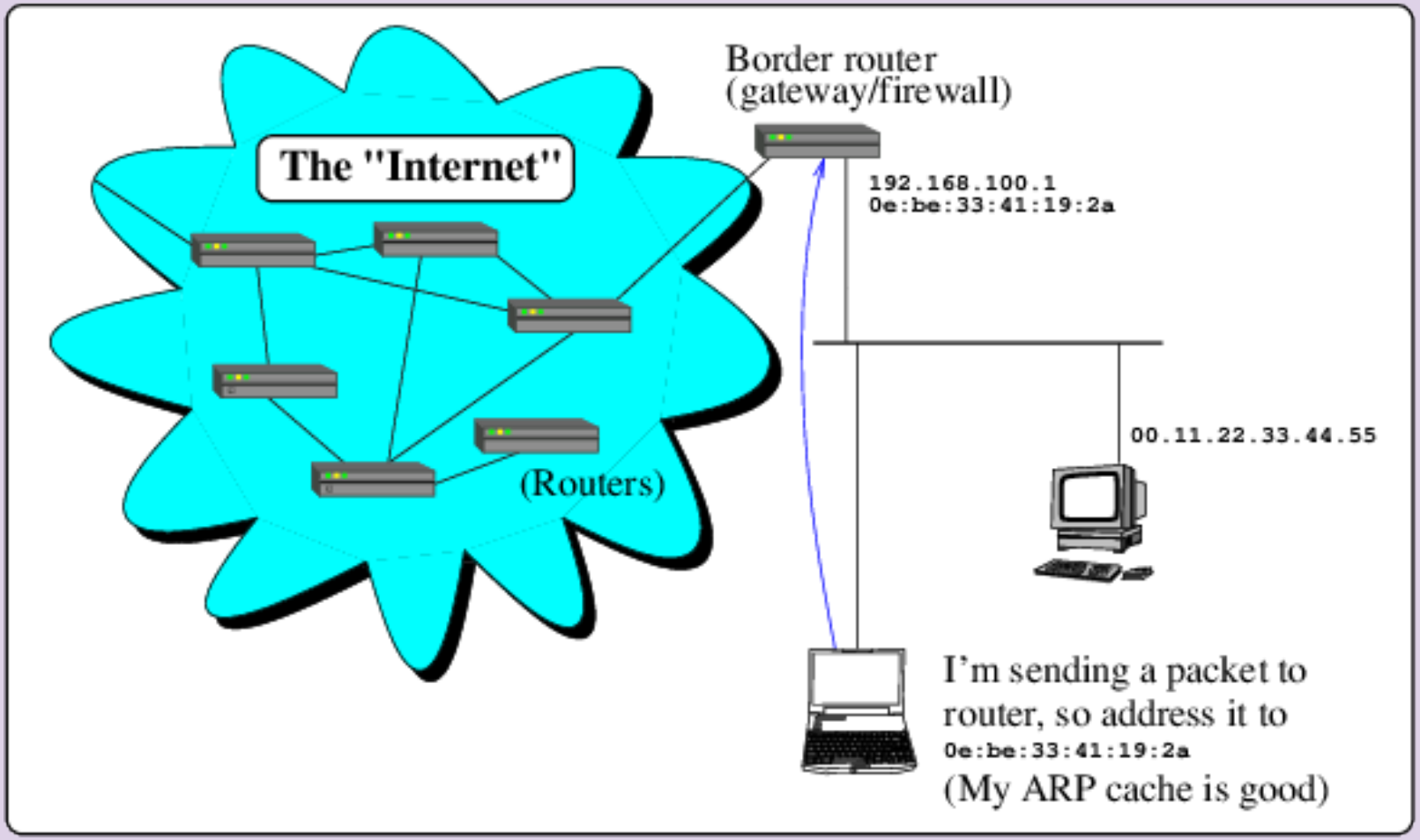
“Trust” when using our computer

Not an exhaustive list by any stretch...

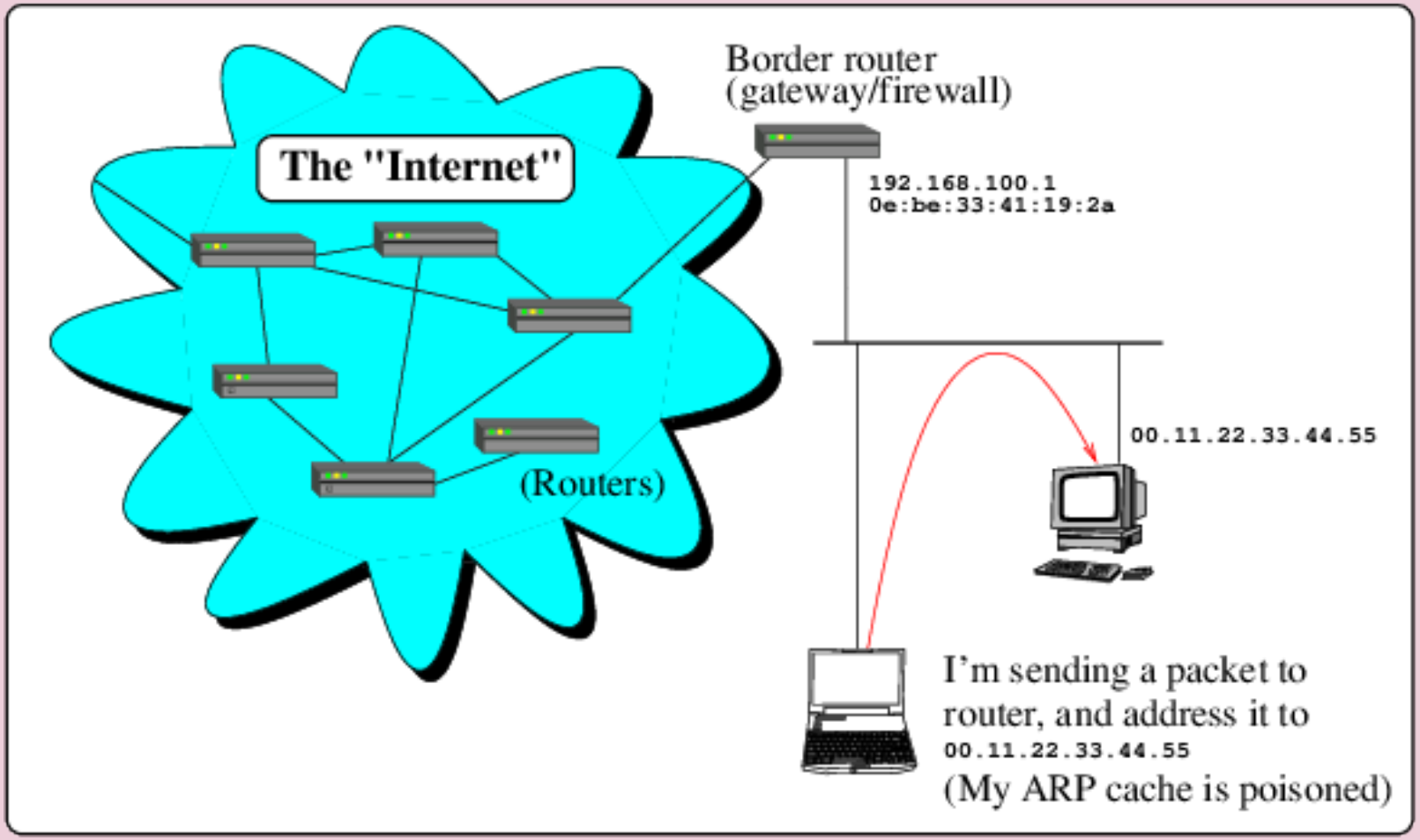
1. Open the laptop: Computer uses various remote services to configure itself to work (DHCP, DNS, ARP, NTP), possibly via routers
2. Start a browser with a particular URL: Computer uses various remote services (DNS, ARP) possibly via routers
3. Request a web page: Computer uses various remote services (HTTP, ARP), and sends info via routers
4. Automatic updates: Computer uses various remote services (DNS, ARP, HTTPS) via routers
5. PKI certificates: If they are signed they must be good right?

In general we trust each one of these remote services and routers to do what we want.

MAC address for point-to-point communication...



MAC address for point-to-point communication...



ARP poisoning

Mechanism can be used to subvert any traffic...

ARP is a local network number to MAC address mapping service, with caches for efficiency.

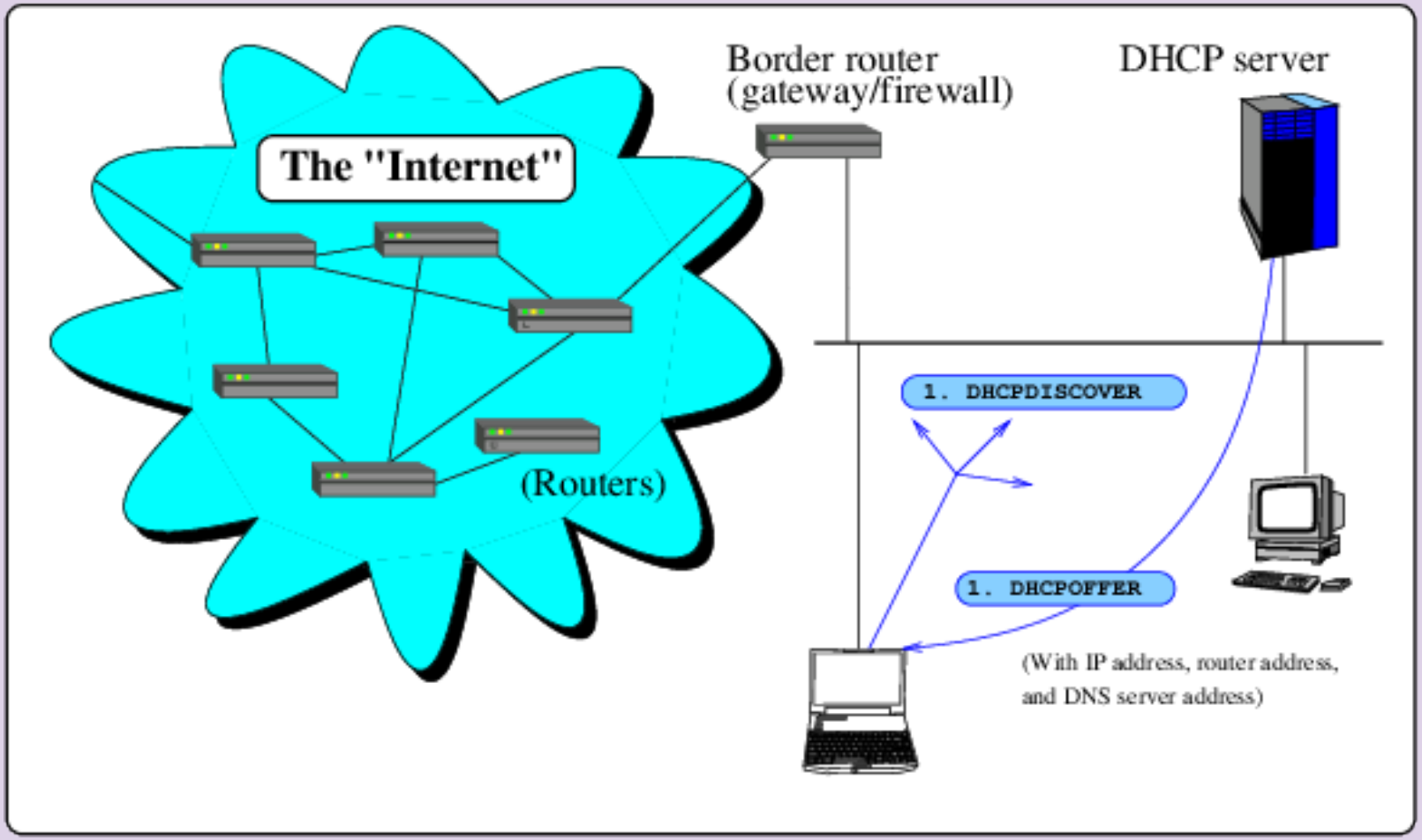
If an ARP cache is poisoned, the machine using that cache may send information to the wrong machine

Example: your ARP cache has been poisoned convincing you that your local router 192.168.100.1 is at MAC address `f8:1e:df:e2:b4:63`. This happens to be a hacking machine on your network. You send your packets addressed to that machine (which responds to convince you it is 192.168.100.1).

Example: your router's ARP cache has been poisoned to convince it to reply to the hacking machine instead of your PC.

Open the laptop: DHCP

Before you do anything...



Open the laptop: DHCP

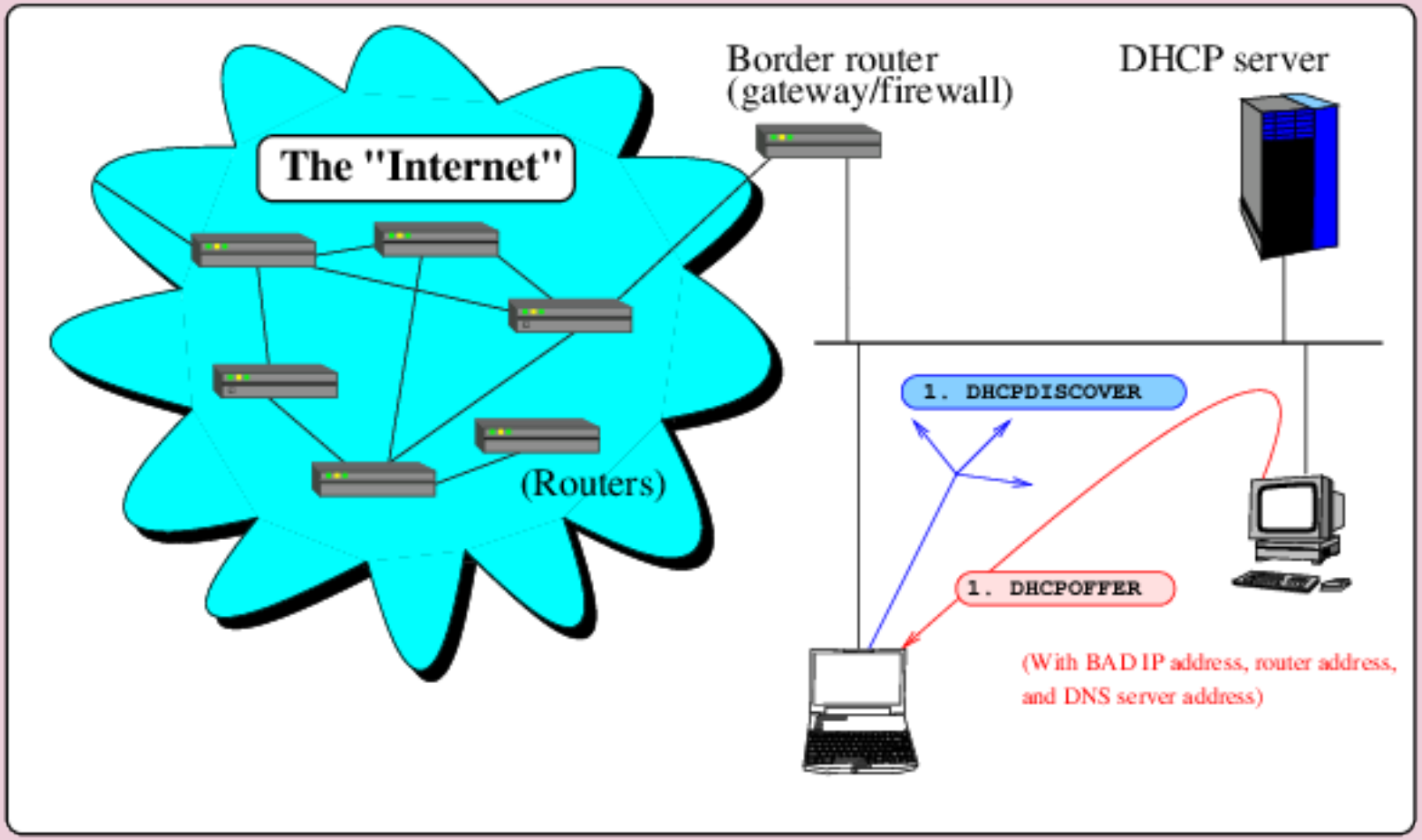
Before you do anything...

Your computer, when it wakes up, may try to configure itself by using DHCP (Dynamic Host Configuration Protocol). DHCP on your computer expects to find a DHCP server somewhere. Your computer does not know its own network (IP) address, and does not know the server's IP address , so it...

1. broadcasts a DHCPDISCOVER packet to IP broadcast address 255.255.255.255.
2. The (local) DHCP server(s) respond(s) with DHCPOFFER (offers of information) such as your new IP address, network mask, and local router and a DNS (Domain name server) your machine can use.
3. Your computer accepts one of the offers with a DHCPREQUEST
4. The server acknowledges your acceptance DHCPACK

Open the laptop: DHCP

Sounds easy, what can go wrong?



Open the laptop: DHCP

Sounds easy, what can go wrong?

Well - there is no security at all in the process, so maybe some other malicious DHCP server can respond with bad information.

Example: malicious DHCP server tells you a bad router: from then on you send all your network traffic via this router, and a "man-in-the-middle" reads/processes/modifies all your Internet traffic.

Example: malicious DHCP tells you a bad DNS server: from then on you send all your DNS requests to this server, and our mapping for dbs.com.sg is given as 91.207.5.138 (in Russia) instead of 203.127.2.6 (in Singapore).

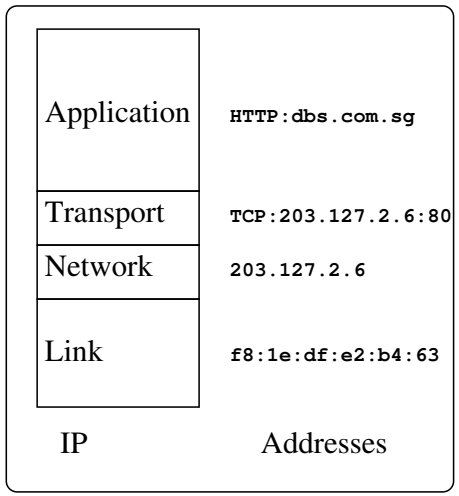
A surprising amount of trust

Lets start with the assumption that our personal computer is safe, no virus, worm, bad software on it.

When we use our computers, we (or our computers) trust all sorts of things we interact with to act honestly.

Each thing we trust may be something that could be used to attack us. There are multiple machines we rely on, and also multiple protocols. With more complex systems, there are more avenues for attacks/insecurity

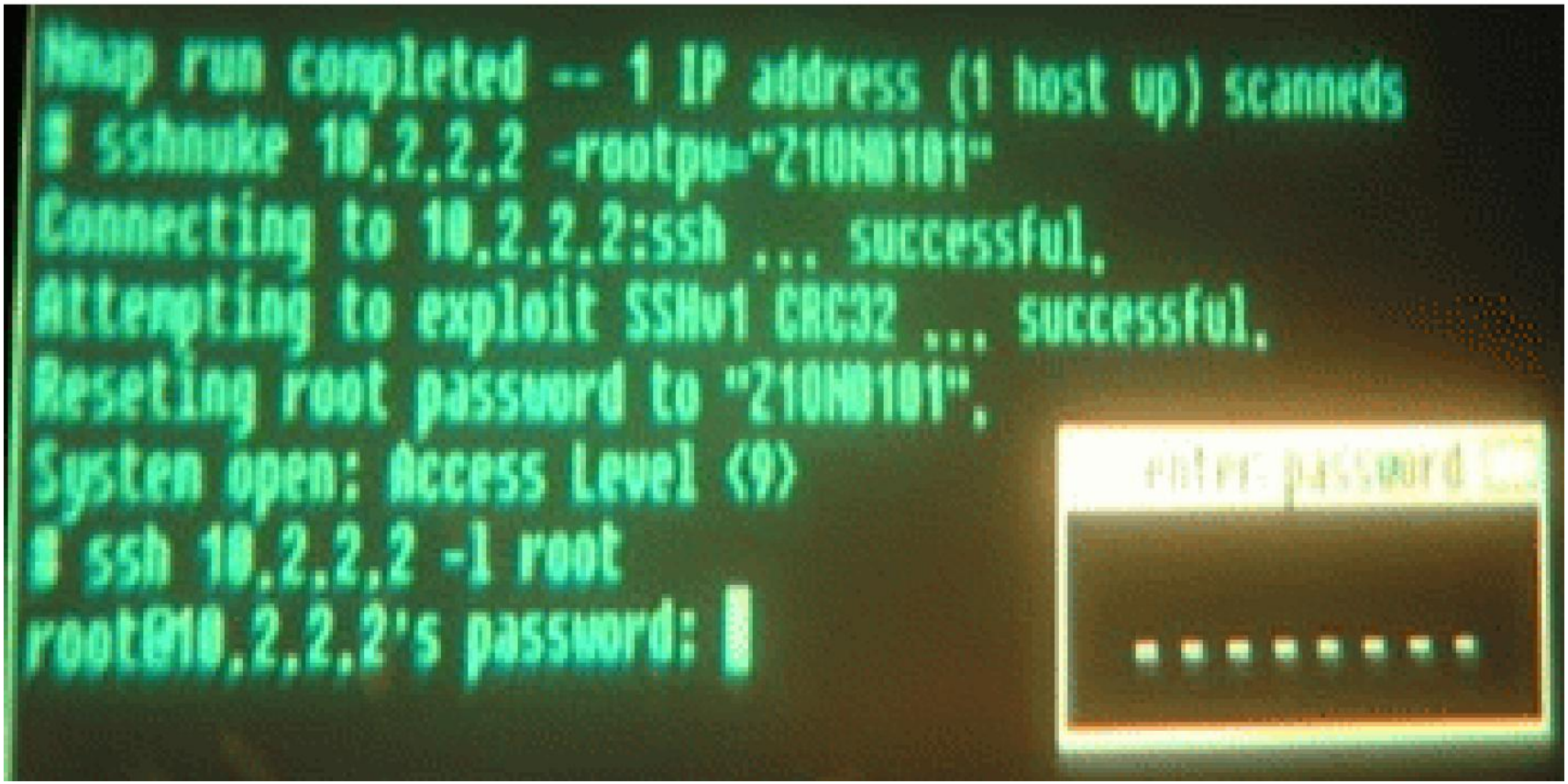
Messin' with addressin'



- Application layer: Various (higher level) protocols, and names (dbs.com.sg) rather than IP addresses
- Transport layer: your PC will use specific ports, with specific (transport-layer) protocols such as TCP or UDP. (TCP,IP,PORT).
- Network layer: your PC will aquire an IP address(es) (203.127.2.6) to identify your machine to others.
- Link layer: network hardware in your PC comes with a factory built-in world-wide media access control (MAC) address. For ethernet `f8:1e:df:e2:b4:63`. It is easy to forge MAC addresses.

In summary - each layer has its own mode of addressing, and we have automatic translations from MAC to network addresses (ARP), and from names to IP addresses (DNS).

Which ports? What services?

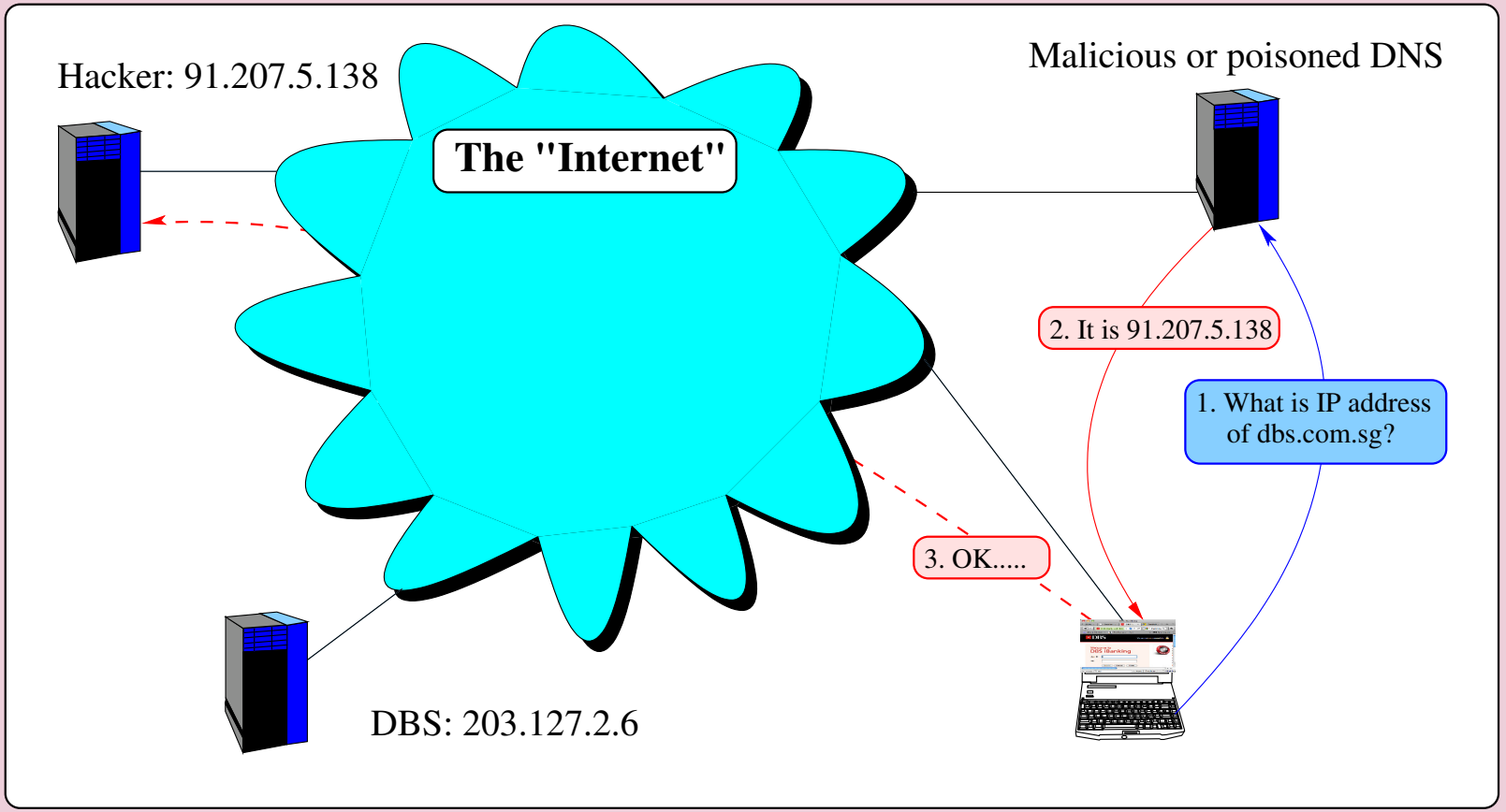


It is just a movie, but you should be paranoid...

Systems are dangerously easy to subvert. And look! nmap!

Start a browser with URL: DNS

What can go wrong?



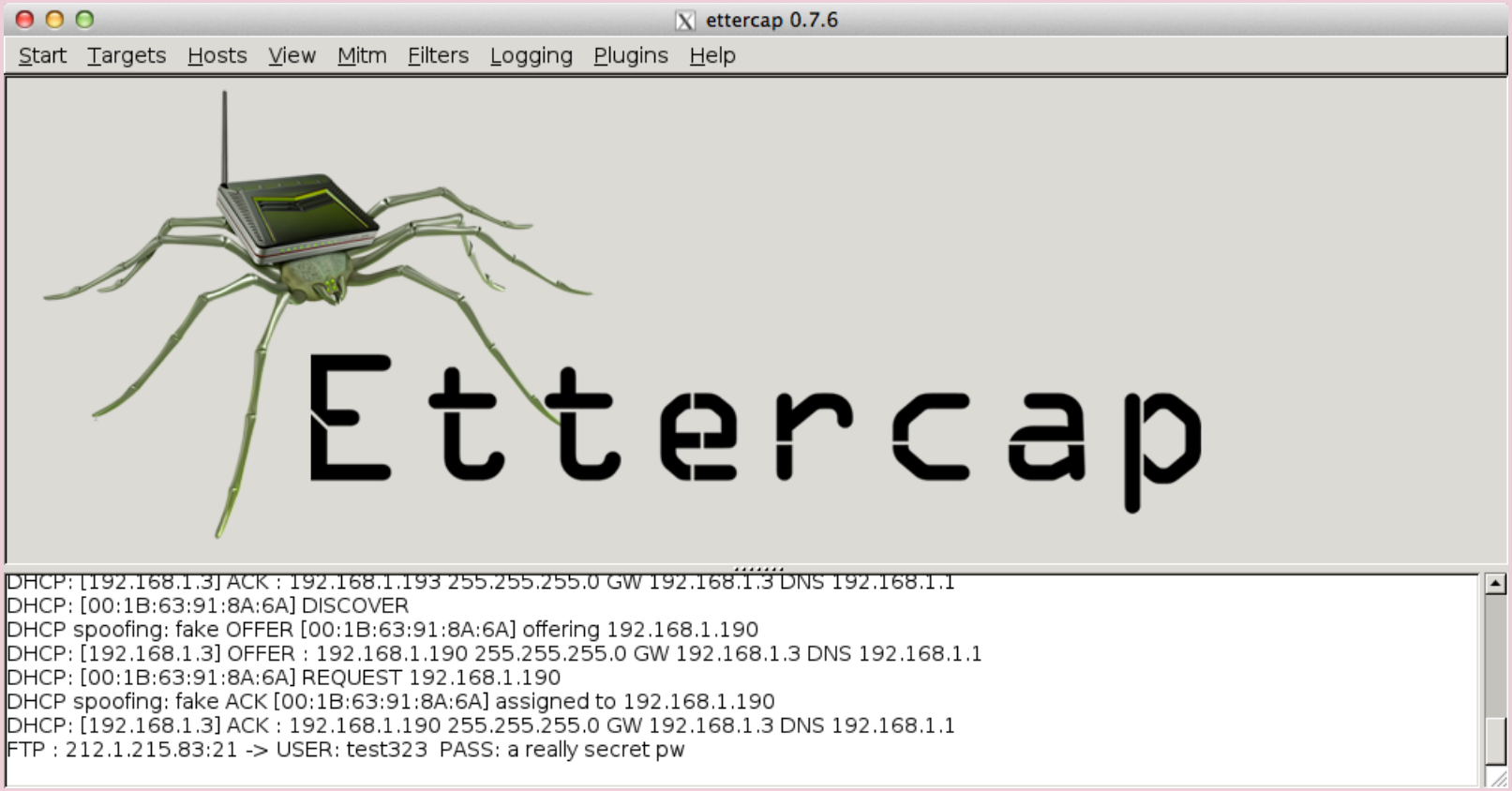
Start a browser with URL: DNS

Sounds easy, what can go wrong?

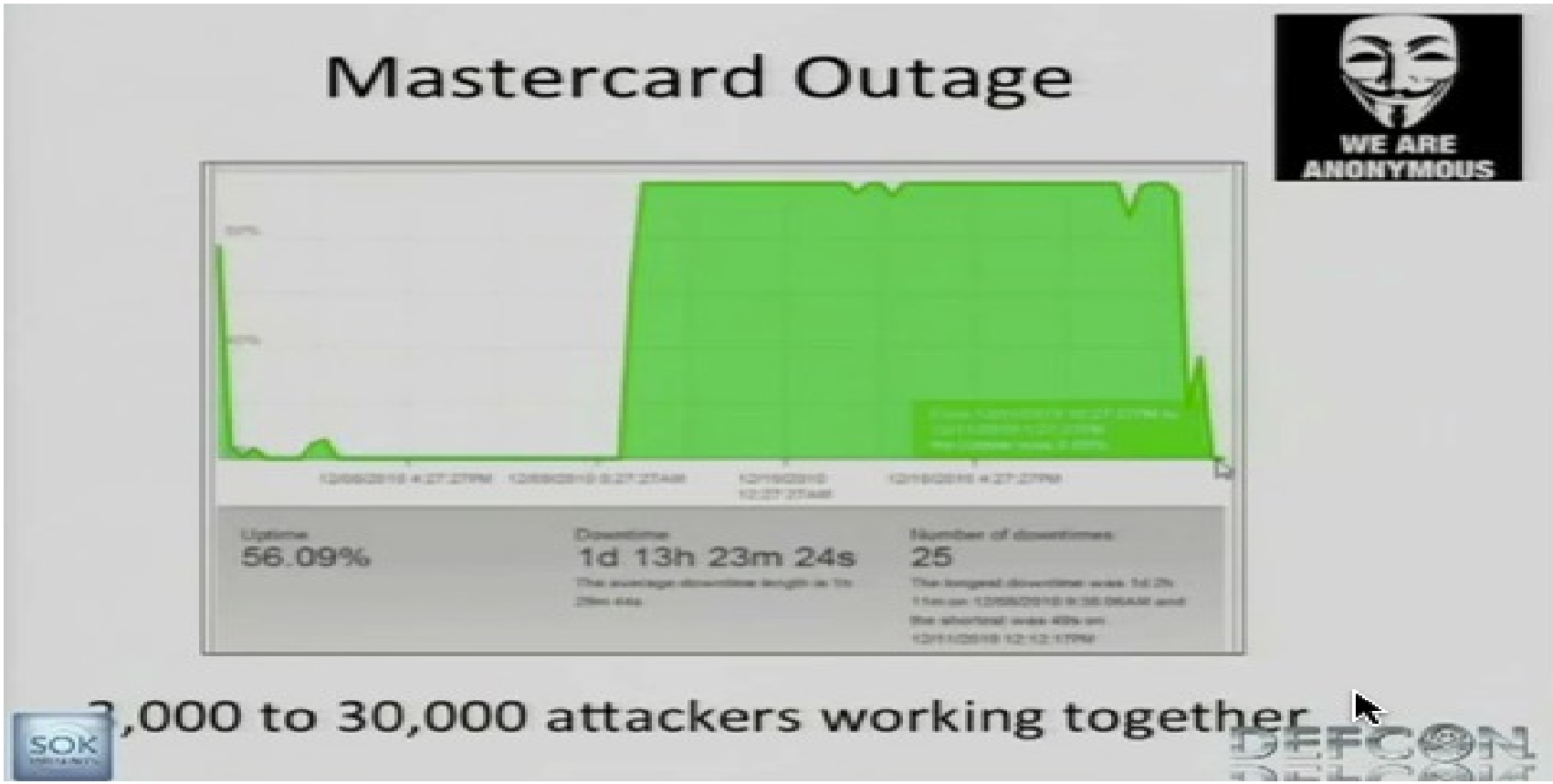
- Well - there is little security in the process, so maybe
  - Your own cache could have been poisoned
  - Your local server's cache could have been poisoned
  - A server upstream of your DNS could be poisoned
  - An ARP cache is poisoned somewhere (next slide)
- Example: a machine on your local network sees your DNS request for dbs.com.sg and responds quickly with corrupt information (91.207.5.138):
  - from then on your computer send all your network traffic to the corrupt address; your cache is poisoned.
- Example: a machine on an upstream network sees your DNS server's request for dbs.com.sg and responds quickly with corrupt information (91.207.5.138):
  - from then on your computer send all your network traffic to the corrupt address; DNS server's cache is poisoned.

Ettercap - spoofing and logging

In this case, DHCP

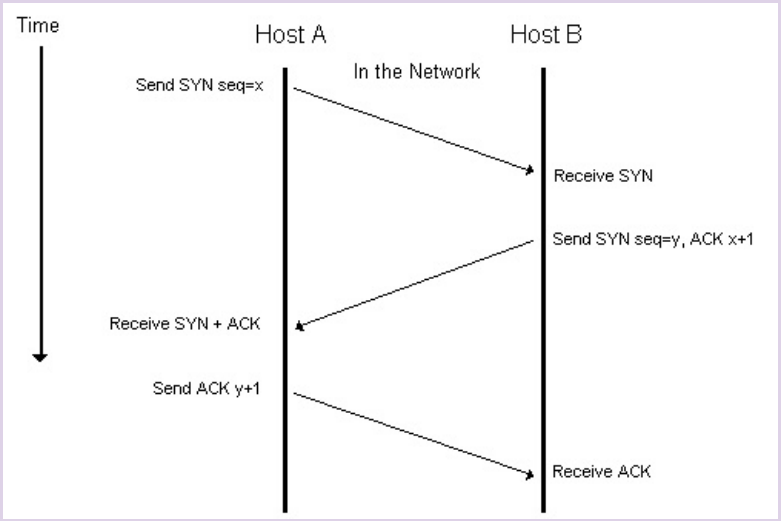


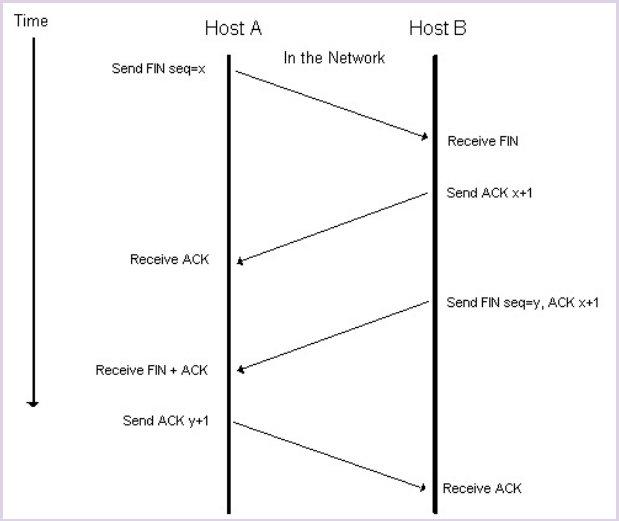
Layer 4 attacks (From Sam Browne)



Looking at connections

Starting and terminating a TCP connection...

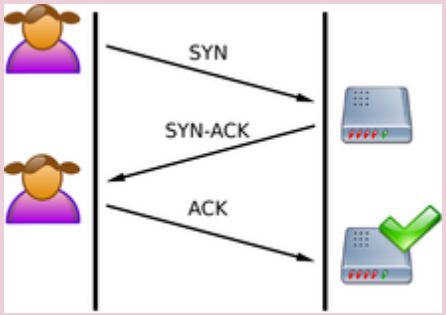


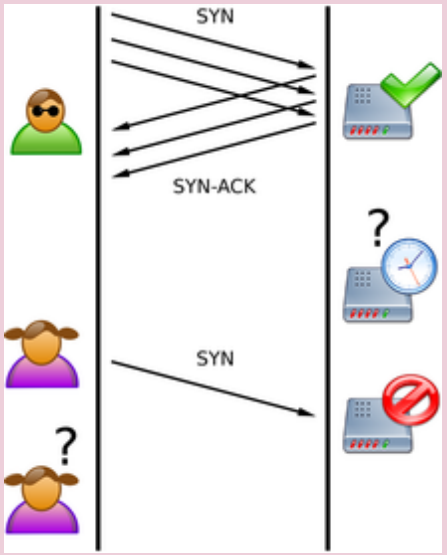


To start a connection there is a 3-way handshake. During this time, initial sequence numbers are exchanged - a different number used in each direction. To end a connection, there is just a 2-way handshake.

SYN flood attack

Attack the connection setup





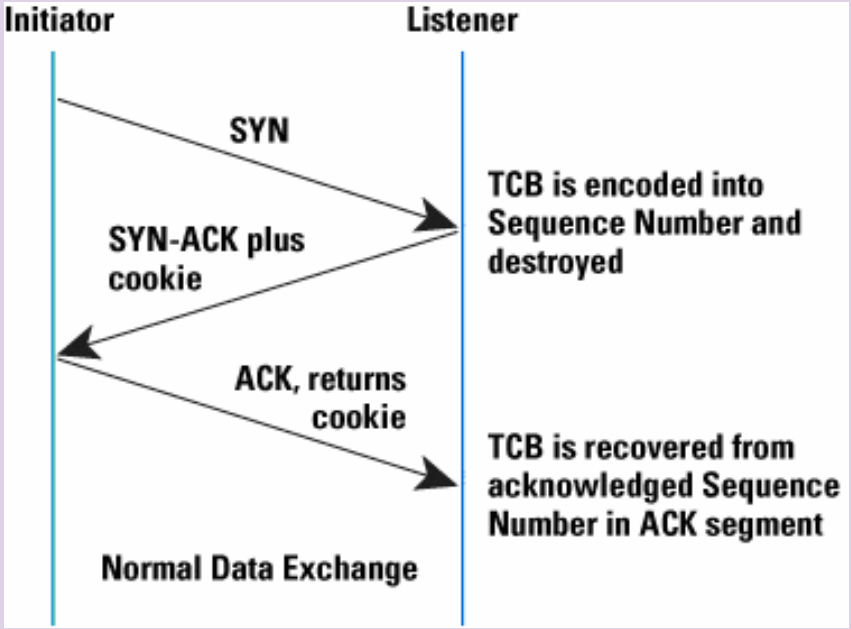
Attacker sends lots of requests, and does not complete the connection.

The receiver allocates resources for each connection (Ports, tables etc), until no more resources are left.

When Alice tries to connect, she is unable to.

SYN flood attack

Protecting the server

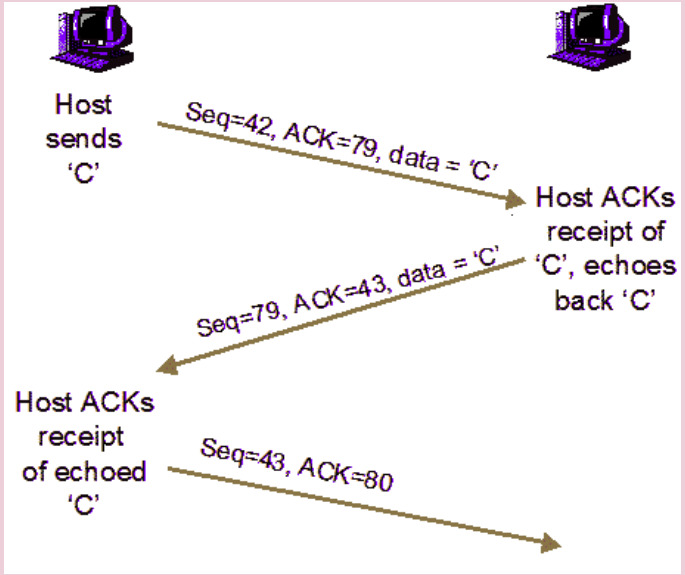


Server returns a specially constructed sequence number, which encodes the resources for the connection (TCB).

TCB only allocated when ACK is returned.

TCP sequence prediction attack

Attack/takeover an existing connection...

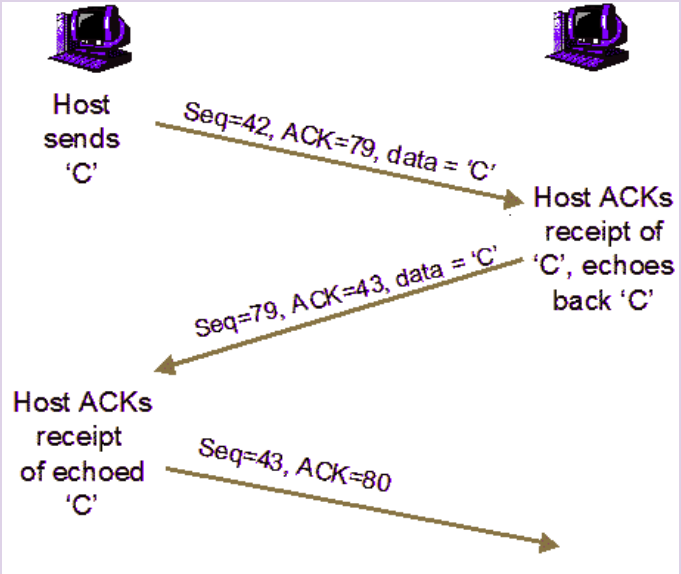


Sequence numbers allow re-ordering of packets.

If an attacker can predict the correct sequence number for a response, and get in before the server sends its response (perhaps by SYN-flooding the server), attacker can start pretending to be the server. (Even without being able to see the packets).

TCP sequence prediction attack

Check other layers, disallow prediction...



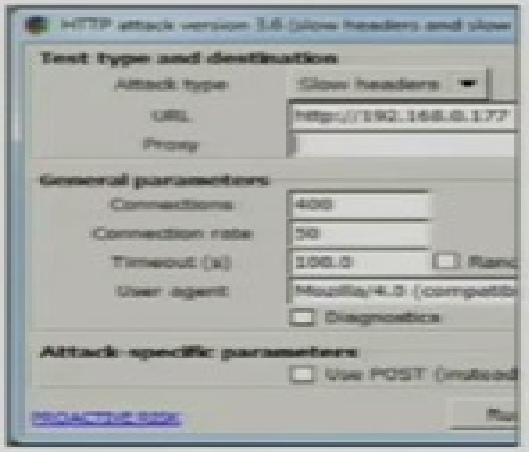
Use unpredictable sequence numbers.

Checking other layers for evidence of spoofing (such as source IP address) can help minimize success of this attack.

Layer 7 attacks (From Sam Browne)

SlowLoris

- Send incomplete GET requests
- Freezes Apache with one packet per second



Another vector: the routers to (re) direct traffic

And they are computers...

Since the Internet topology/connectivity can change, the routers continually exchange “routing information”.
This is in line with the original goals of a system which can survive nuclear attacks on parts of the network.
Routing information is communicated using routing protocols, for example:

RIP: (Routing Information Protocol) an older protocol for exchanging routing information. Your PC understands RIP, but most routers ignore it these days. Not all.

IGRP: (Interior Gateway Routing Protocol) is a proprietary (CISCO) protocol, created to overcome limitations of RIP for large networks.

BGP: (Border Gateway Protocol) makes most routing decisions on the Internet.

These protocols update tables/caches in each connected router to allow it to make decisions where to send IP traffic.

We trust routers to (re) direct traffic

Sounds easy, what can go wrong?

Two general attack vectors:

1. Hack directly into specific routers and re-configure them to behave differently.

   Note that many routers allow FTP/telnet/HTTP access to control them - we have already seen how FTP passwords can be observed. Routers are just computers - typical home routers often run some cut down variant of linux.

2. Corrupt routers caches/routing tables without hacking

   Various attacks have been done using the BGP, IGRP and RIP protocols.
   On April the 8th, 2010, a significant amount of American Internet traffic was re-routed through China for 18 minutes after 40,000 peculiar routes were injected into the core Internet routers by a Chinese ISP. This was most likely an accident, but interesting nevertheless.

Summary

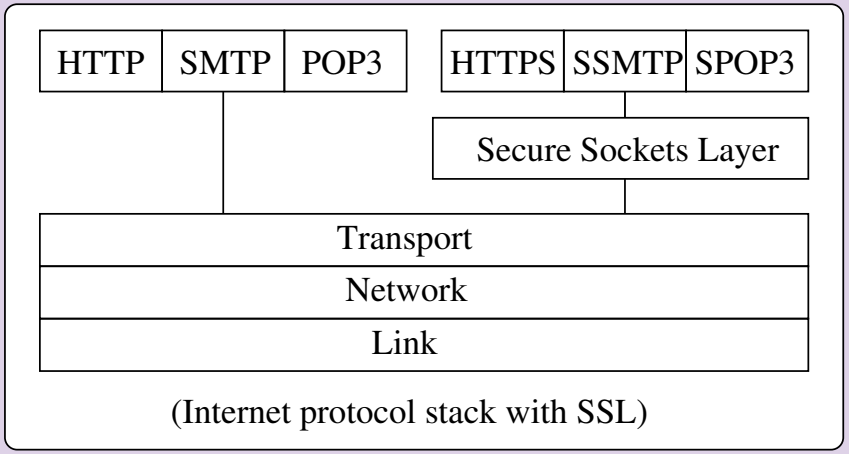
Many opportunities for manipulation:

1. Your configuration information could be altered (your PC gets bad IP addresses, DNS servers, routers)
2. Your PC can be convinced to send your traffic via a hacker's computer for man-in-the-middle attacks
3. Your ability to map names to IP addresses, and IP addresses to hardware MAC addresses can be manipulated (DNS and ARP poisoning)
4. Resources on a server can be exhausted to do DoS.
5. Ongoing sessions you have initiated (for example to get a web page) can be hijacked midway through.

Outline

SSL

A transport Layer security service



(Internet protocol stack with SSL)

Transport Layer security service

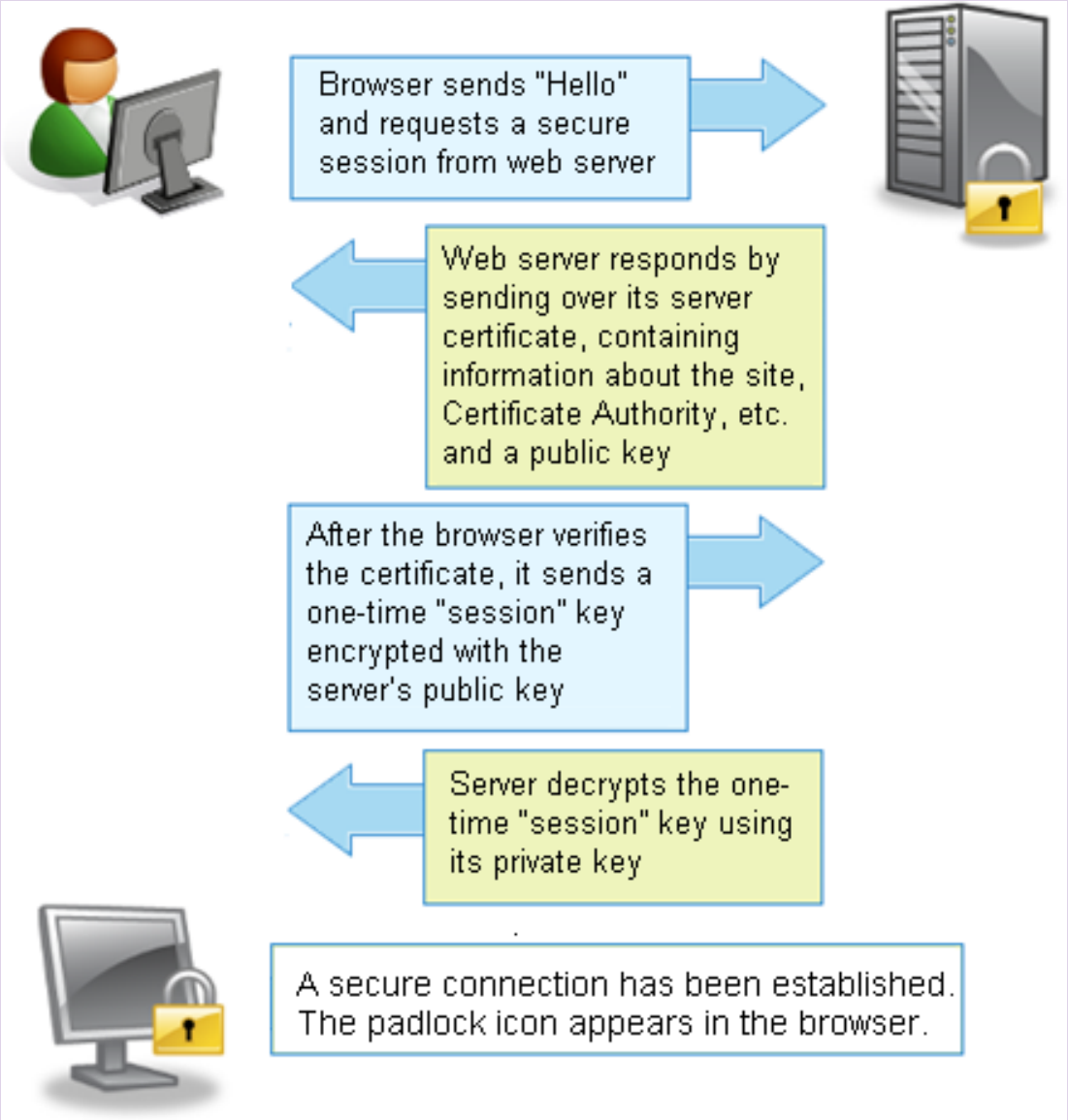
Originally, SSL was developed by Netscape in 1995. TLS is an IETF standard: `http://tools.ietf.org/html/rfc5246`

Public key cryptography is used to exchange session keys, for ordinary symmetric key communication, with new session keys for each connection. Symmetric encryption schemes used include IDEA, DES, 3DES...

There is an SSL handshake (initialization) protocol, to authenticate, and negotiate encryption and MAC algorithms, and to exchange session keys.

SSL handshake

Simplified view



SSL handshake

Simplified view

1. Browser sends the (encryption) algorithms the browser will support and a nonce (one time value)
2. Server responds with a nonce, certificate and which algorithms to use.
3. Browser verifies the certificate, and extracts Server public key. Encrypts a pre-master session key with the public key.
4. Server decrypts using it's private key (Both compute session key from pre-master and nonces)

Session switches to the correct algorithm with the calculated key. The protocol sends a few more packets, authenticating the previous messages.

Summary/key points...

Standard protocols (HTTP, SMTP ...) are "tunnelled" through the SSL layer. PKI,PK are used for authentication, and to exchange session keys. Symmetric protocols (such as 3DES) are used for encryption.

The protocol includes an idea of not just public/private keys, but public keys authenticated by a certifying authority.

SSL has problems too

Web no longer secure, civilization will soon collapse...

Unfortunately, in 2009, various attacks on SSL/TLS were discovered, from loopholes in the protocol. An attacker is able to execute any HTTP transaction, authenticated by a legitimate user

---

You can read about it here `http://www.ietf.org/mail-archive/web/tls/current/msg03928.html`.

It allows a man-in-the-middle to inject a chosen plaintext prefix into the encrypted data stream. An “authentication gap” exists during a renegotiation process during which separate SSL/TLS connections can be put together. Fixes were done in early 2010, but many servers and browsers still not fixed.

IPsec

IPsec is a set of standards:

Intended to support communication security between networked computers, particularly in the newer IPv6 (IP Next-Generation) network.
IPsec software is available in Windows, Linux, and on routers on the Internet.
`http://www.faqs.org/rfcs/rfc4301.html`
IPsec may be used in a range of ways.

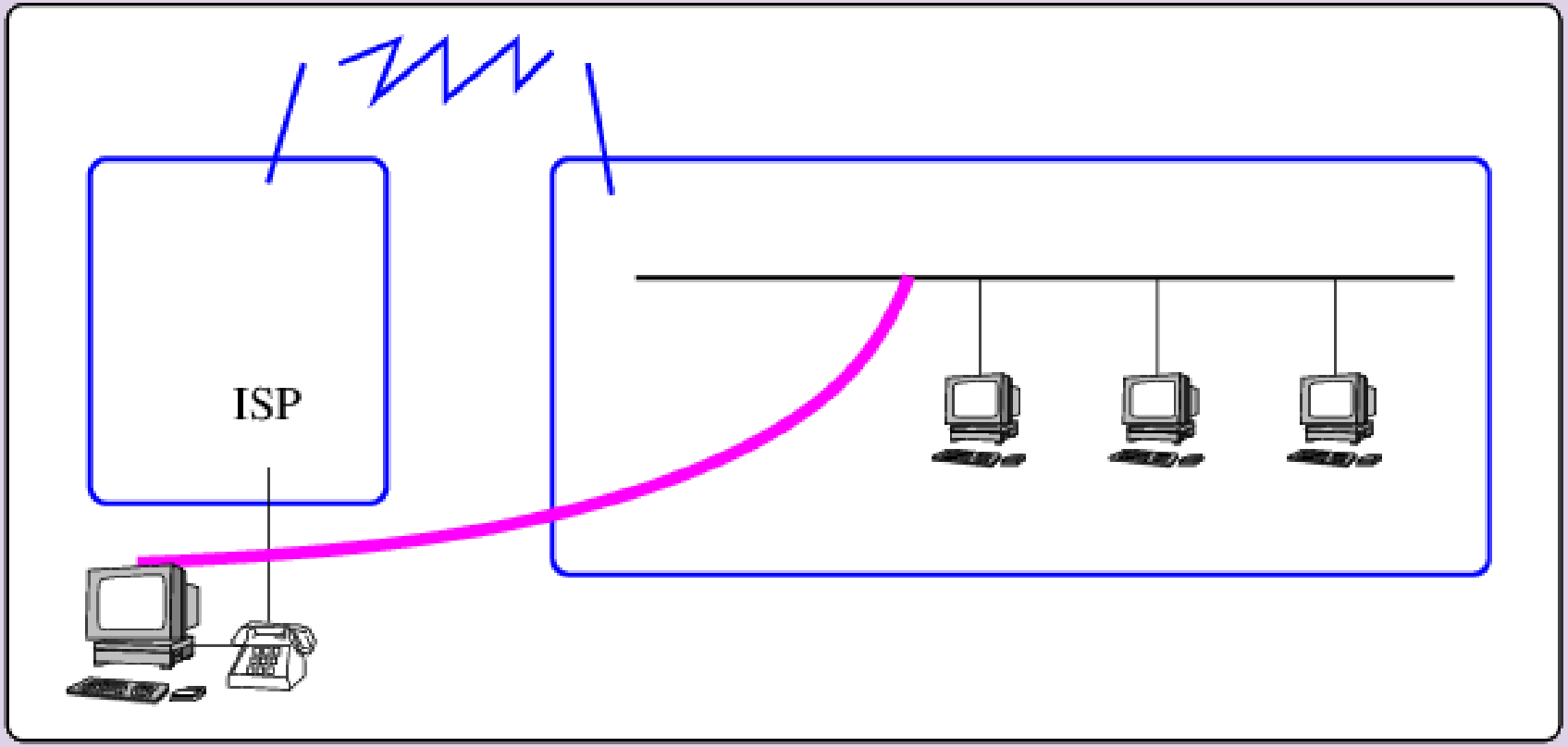
Why VPN when we already have SSL and SSH...

- With SSL/SSH we set up individual encrypted links for each application
- We may want end-to-end encryption, where all traffic is encrypted (not just the https traffic)
- The model is that a tunnel is set up between a host and a network and all traffic through that tunnel is encrypted.
- With VPN, we take care and use (high end) cryptography to set up the tunnel

Note that again there can be problems...

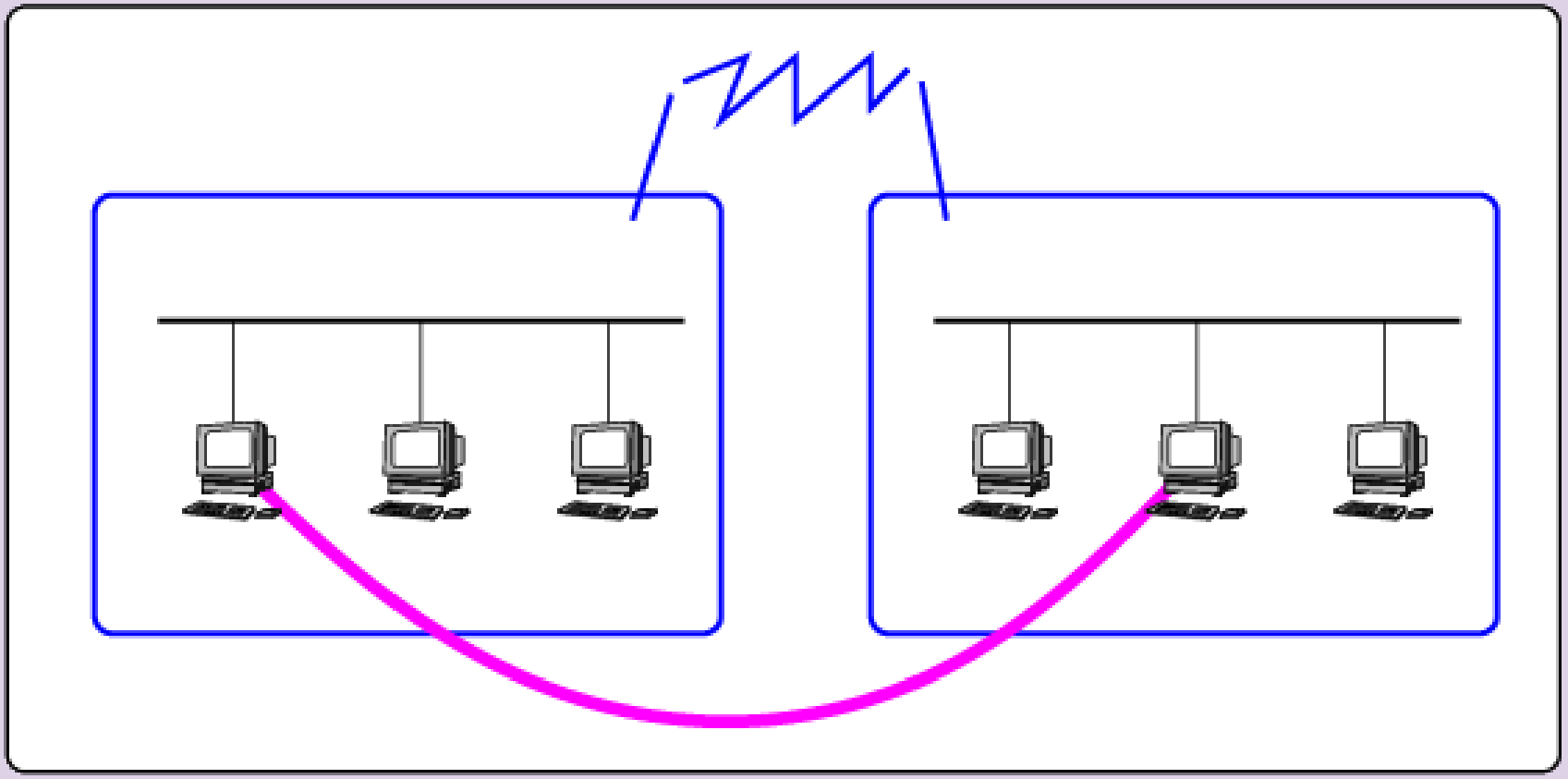
IPsec VPN (Virtual Private Network)

Provides secure connection so machine appears on remote network



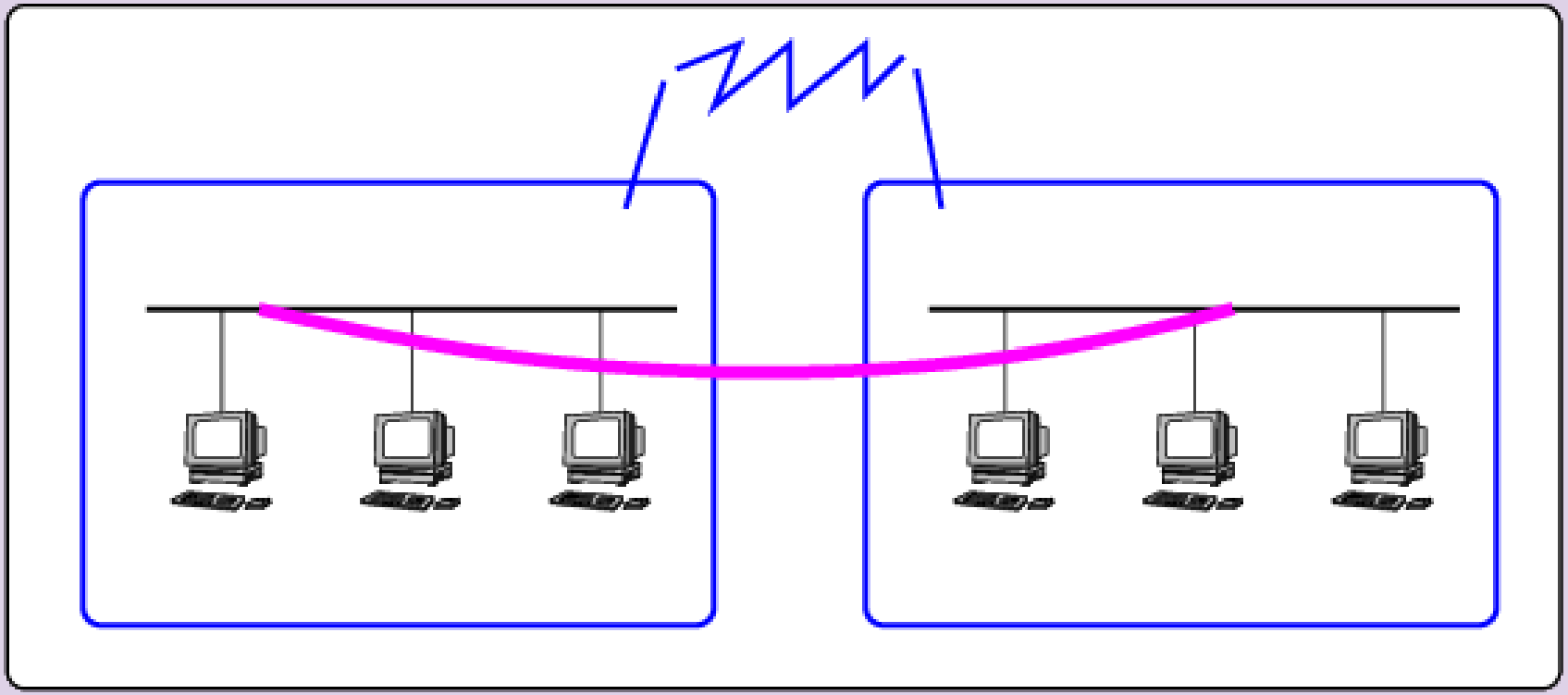
IPsec point-to-point

Or secure communication from one PC to another



IPsec network-to-network

Or from one network to another



IPsec headers

The IP headers contain the extra information needed

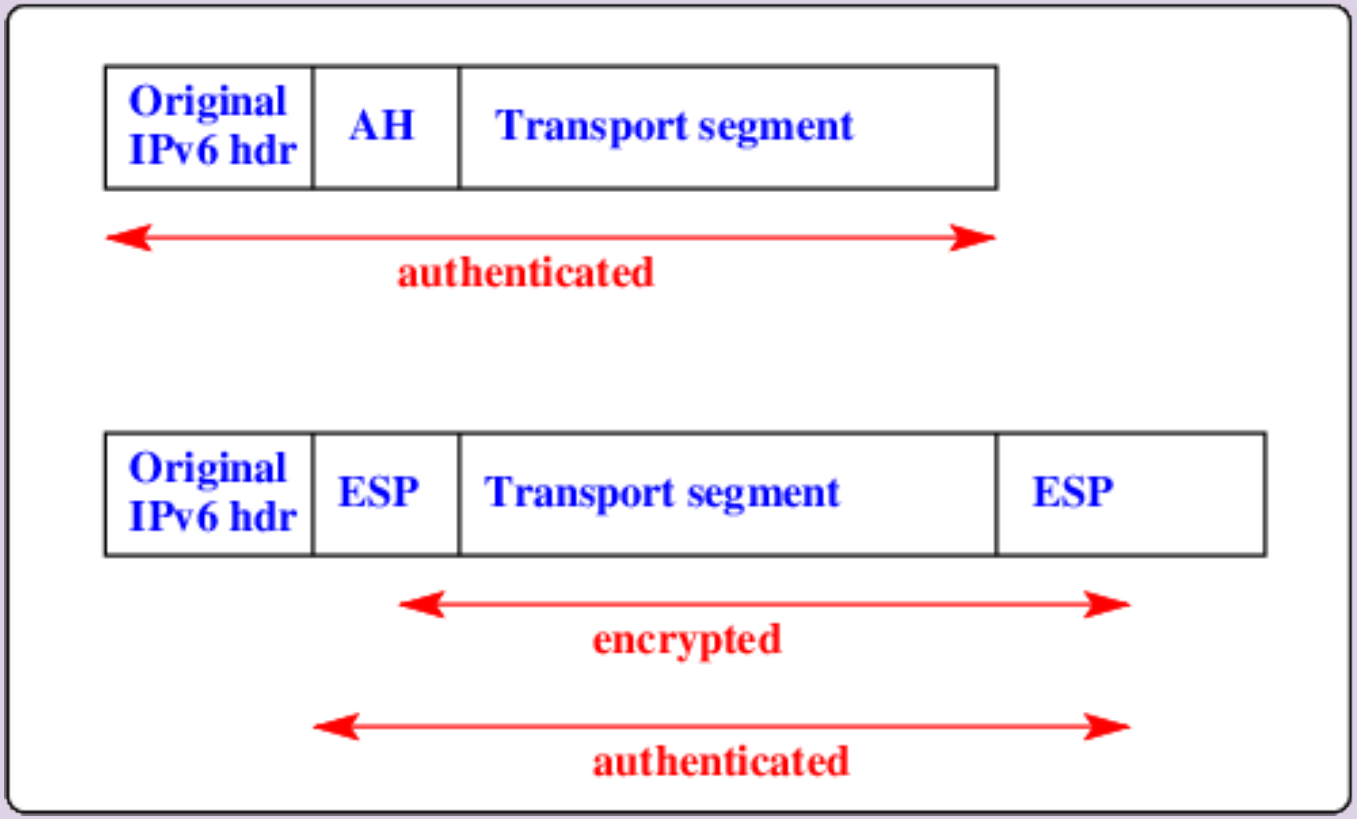
There are two types of header, one used for **authentication**, and the other used for **encryption**:

1. **AH** - the **Authentication Header** for data integrity, anti-replay and authentication
2. **ESP** - the **Encapsulating Security Payload** header, for confidentiality. ESP can also provide AH services.

Communicating parties agree on a **Security Association** (SA), one SA for each direction, and one SA for each type of communication.

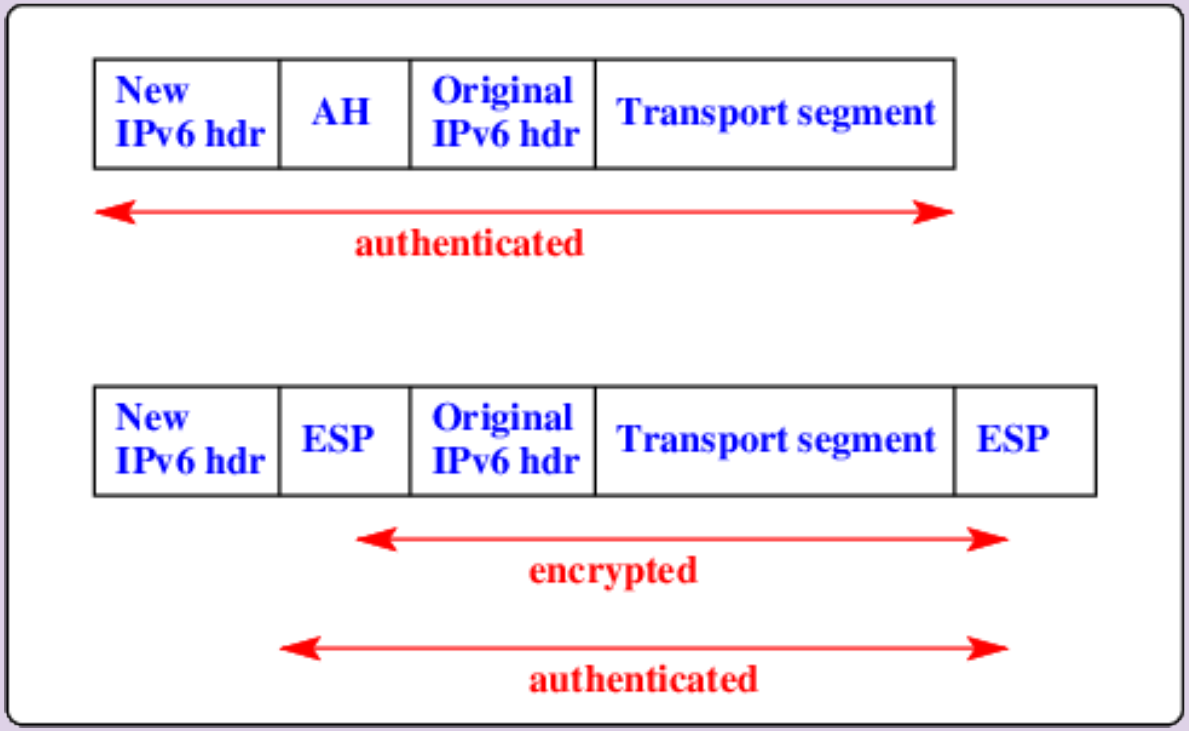
Modes of operation

An end-to-end SA - Transport mode:



Modes of operation

An SA between security gateways - Tunnel mode



SAs form a kind of distributed database.