Internet Technology

http://en.wikipedia.org/wiki/Internet_Protocol_Suite

https://www.coursera.org/course/insidetheinternet

Coursera

---

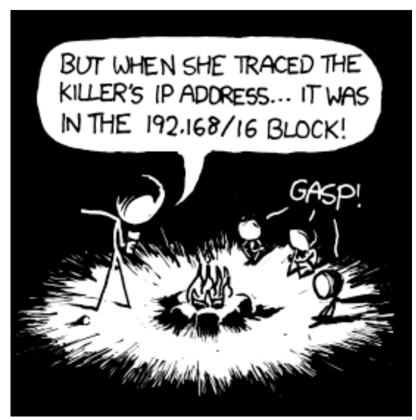
open.michigan

University of Michigan

Copyright Thanks

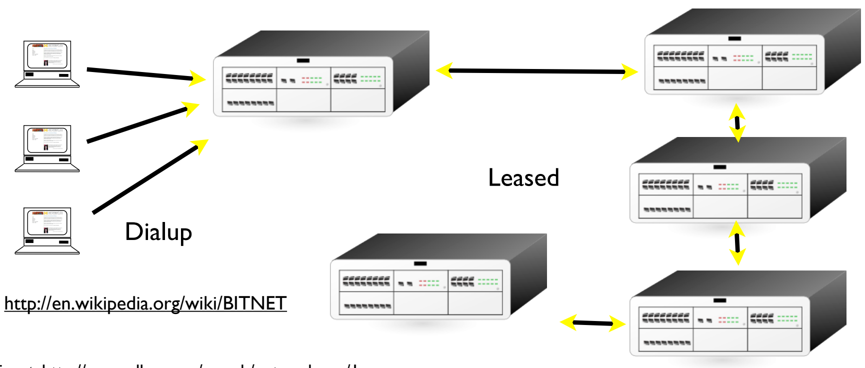
- Thanks to IEEE Computer for permisison to use IEEE Computer magazine articles associated with the videos
- Thanks to Dave Malicke and Open Michigan (open.umich.edu) for help with copyright review of these materials

---



http://xkcd.com/742/

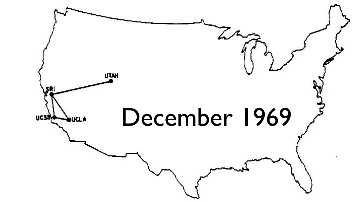
Store and Forward Networking



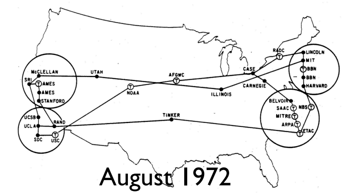
http://en.wikipedia.org/wiki/BITNET

Clipart: http://www.clker.com/search/networksym/1

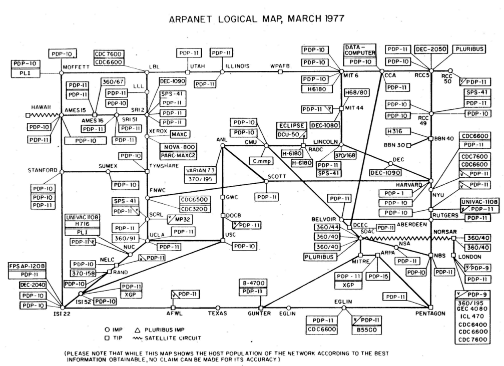
Research Networks 1960-1980's



- How can we avoid having a direct connection between all pairs of computers?
- How to transport messages efficiently?
- How can we dynamically handle outages?



http://som.csudh.edu/fac/lpress/history/arpamaps/



Heart, F., McKenzie, A., McQuillian, J., and Walden, D., ARPANET Completion Report, Bolt, Beranek and Newman, Burlington, MA, January 4, 1978.

http://som.csudh.edu/fac/lpress/history/arpamaps/arpanetmar77.jpg

Efficient Message Transmission: Packet Switching

- Challenge: in a simple approach, like store-and-forward, large messages block small ones
- Break each message into packets
- Can allow the packets from a single message to travel over different paths, dynamically adjusting for use
- Use special-purpose computers, called routers, for the traffic control

Packet Switching - Postcards

Hello there, have a nice day.

Hello ther (1, csev, daphne)

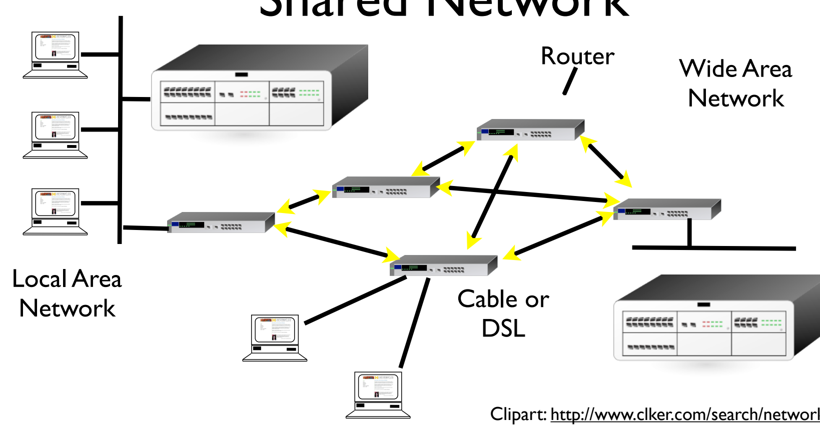
e, have a (2, csev, daphne)

nice day. (3, csev, daphne)

Packet Switching - Postcards

Hello there, have a nice day.

Shared Network

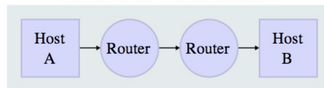


Clipart: http://www.clker.com/search/networksym/1

Shared Networks

- In order to keep cost low and the connections short geographically - data would be forwarded through several routers.
- Getting across the country usually takes about 10 “hops”
- Network designers continually add and remove links to “tune” their networks

Network Connections



Source: http://en.wikipedia.org/wiki/Internet_Protocol_Suite

Layered Network Model

- A layered approach allows the problem of designing a network to be broken into more manageable sub problems
- Best-known model: TCP/IP—the “Internet Protocol Suite”
- There was also a 7 layer OSI: Open System Interconnection Model

Application Layer
Web, E-Mail, File Transfer

Transport Layer (TCP)
Reliable Connections

Internetwork Layer (IP)
Simple, Unreliable

Link Layer (Ethernet, WiFi)
Physical Connections

Internet Standards

- The standards for all of the Internet protocols (inner workings) are developed by an organization
- Internet Engineering Task Force (IETF)
- www.ietf.org
- Standards are called “RFCs” - “Request for Comments”

INTERNET PROTOCOL

DARPA INTERNET PROGRAM

PROTOCOL SPECIFICATION

September 1981

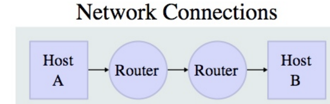
The internet protocol treats each internet datagram as an independent entity unrelated to any other internet datagram. There are no connections or logical circuits (virtual or otherwise).

The internet protocol uses four key mechanisms in providing its service: Type of Service, Time to Live, Options, and Header Checksum.

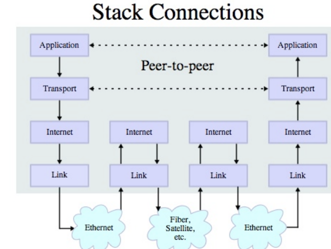
Source: http://tools.ietf.org/html/rfc791

Layered Architecture

Network Connections



Stack Connections

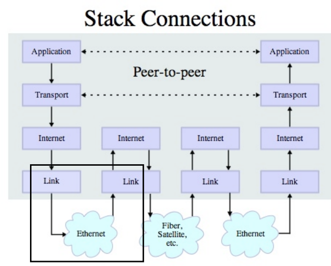


- The Physical and Internet Layers are like trucks and trains - they haul stuff and get it to the right loading dock - it takes multiple steps
- The Transport layer checks to see if the trucks made it and send the stuff again if necessary

Source: http://en.wikipedia.org/wiki/Internet_Protocol_Suite

Link Layer (aka Physical Layer)

- As your data crosses the country may use a different physical medium for each "hop"
- Wire, Wireless, Fiber Optic, etc.
- The link is "one hop" - Is it up or down? Connected or not?
- Very narrow focus - no view at all of the "whole Internet"



Source: http://en.wikipedia.org/wiki/Internet_Protocol_Suite

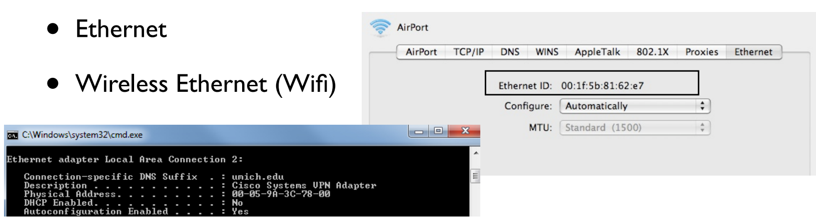
Problems solved by the Link Layer

- **How does data get pushed onto a link?**
- **How is the link shared?**

- **Common Link Technologies**
  - **Ethernet**
  - **WiFi**
  - **Cable modem**
  - **DSL**
  - **Satellite**
  - **Optical**

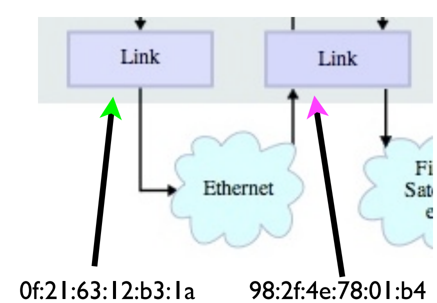
Link Layer Addresses

- Many physical layer devices have addresses built in to them by the manufacturer
  - Ethernet
  - Wireless Ethernet (Wifi)



Link Layer

- Physical addresses are to allow systems to identify themselves on the ends of a single link
- Physical addresses go no farther than one link
- Sometimes links like Wifi and Wired Ethernet are shared with multiple computers



Source: http://en.wikipedia.org/wiki/Internet_Protocol_Suite

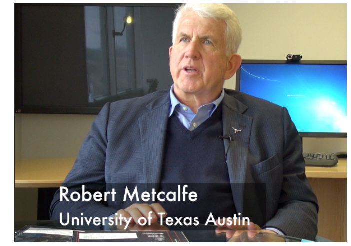
Sharing Nicely - Avoiding Chaos

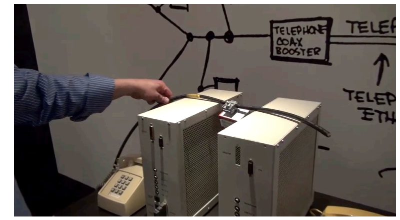
- **CSMA/CD Carrier Sense Multiple Access with Collision Detection**
- **To avoid garbled messages, systems must observe “rules” (Protocols)**
- **Ethernet rules are simple**

- Wait for silence
- Begin Transmitting data
- Listen for your own data
- If you cannot hear your own data clearly, assume a collision, stop and wait before trying again
- Each system waits a different amount of time to avoid “too much politeness”

Ethernet

- Invented at PARC (Xerox)
- The first Local-Area-Network
- Connected PC's to laser printers
- Inspired by an earlier wireless network called Aloha from the University of Hawaii





Internetwork Layer (IP)

http://en.wikipedia.org/wiki/Internet_Protocol

http://en.wikipedia.org/wiki/Traceroute

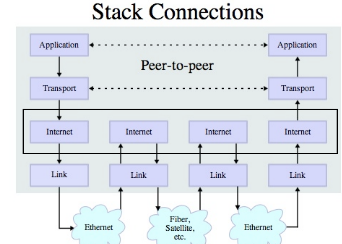
http://en.wikipedia.org/wiki/Ping

Application Layer Web, E-Mail, File Transfer
Transport Layer (TCP) Reliable Connections
Internetwork Layer (IP) Simple, Unreliable
Link Layer (Ethernet, WiFi) Physical Connections

Internet Protocol Layer

- Goal: Gets your data from this computer to the other computer half way across the world
- Each router knows about nearby routers
- IP Is best effort - it is OK to drop data if things go bad...

Stack Connections



Source: http://en.wikipedia.org/wiki/Internet_Protocol_Suite

IP Addresses

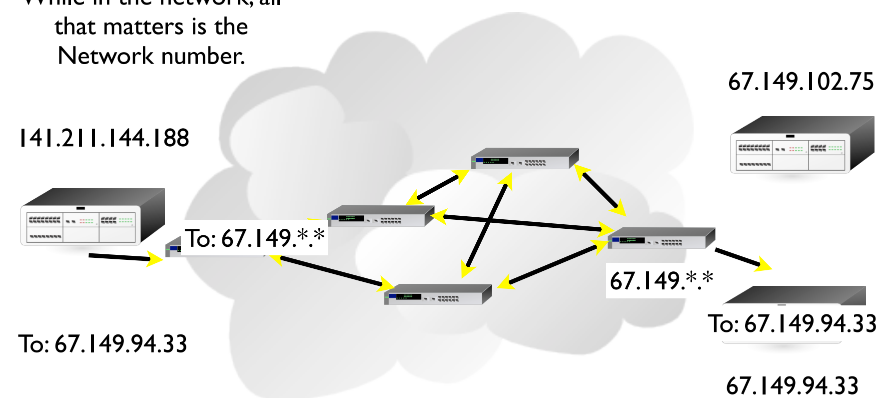
- The IP address is the worldwide number which is associated with one particular workstation or server
- Every system which will send packets directly out across the Internet must have a unique IP address
- IP addresses are based on where station is connected
- IP addresses are not controlled by a single organization - address ranges are assigned
- They are like phone numbers – they get reorganized once in a great while

IP Address Format

- **Four numbers with dots - each number 1-255 (32 bits)**
- **Kind of like phone numbers with an "area code"**
- **The prefix of the address is "which network"**
- **While the data is traversing the Internet - all that matters is the network number**

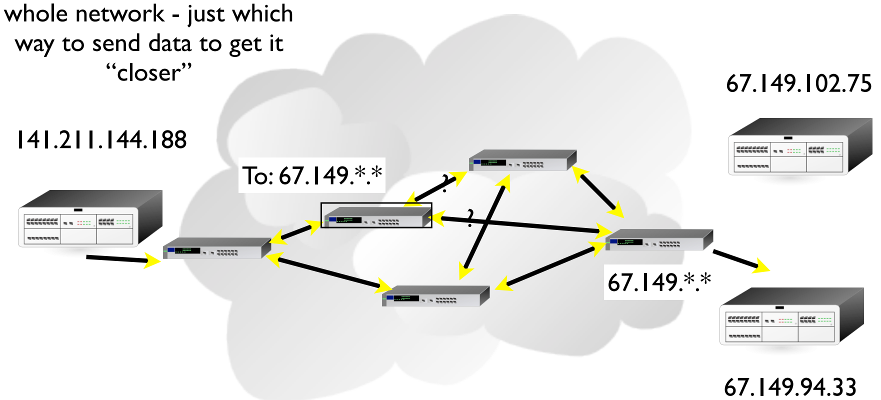
(734) 764 1855		141.211.144.188
Area code	Network Number	141.211.*.*

While in the network, all that matters is the Network number.



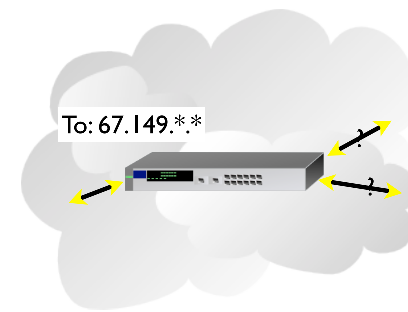
Clipart: http://www.clker.com/search/networksym/1

No single router knows the whole network - just which way to send data to get it "closer"



Clipart: http://www.clker.com/search/networksym/1

Router Tables

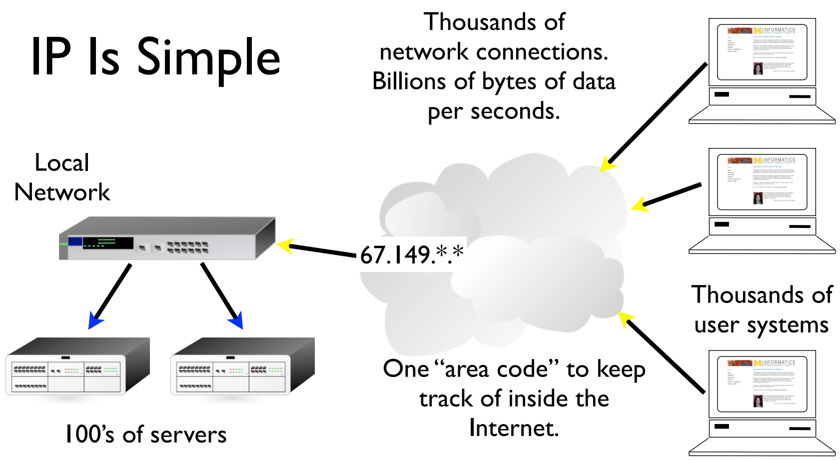


Lists of where to send packets, based on
- destination network address;
- bandwidth on adjacent links;
- traffic on adjacent links;
- state of neighbor nodes (up or not);
- . . .

Updated dynamically
Routers "ask each other" for information

Clipart: http://www.clker.com/search/networksym/1

IP Is Simple



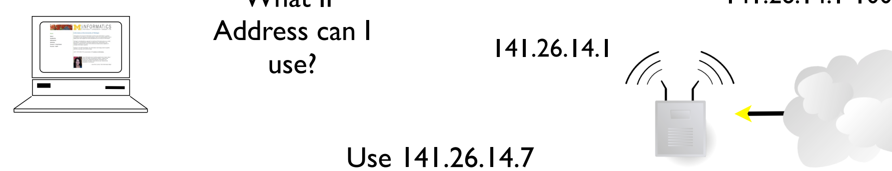
Clipart: http://www.clker.com/search/networksym/1

DHCP = Dynamic Host Configuration Protocol

Hello?

Here I am

What IP Address can I use?

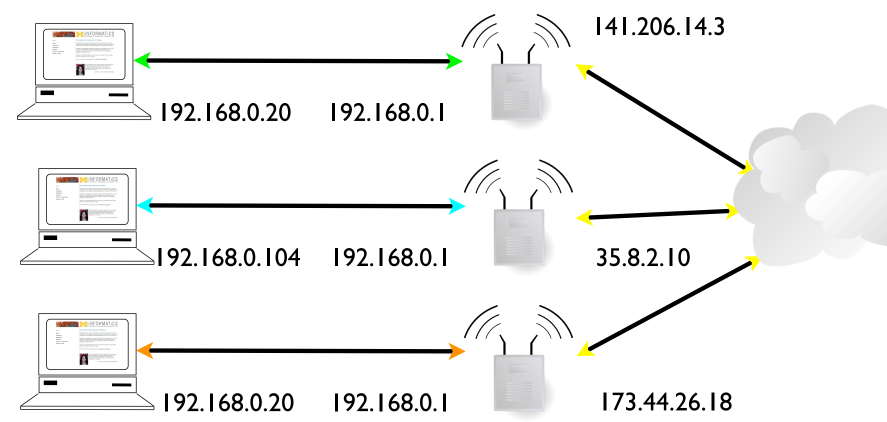


Use 141.26.14.7

Non-Routable Addresses

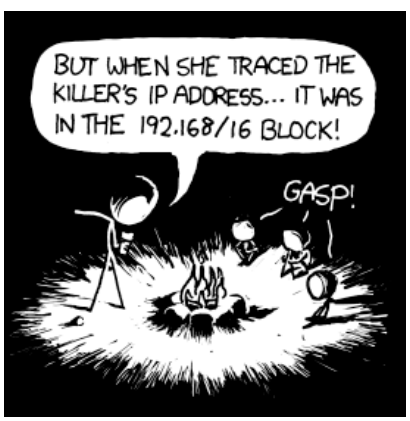
- A typical home router does Network Address Translation (NAT)
- Your ISP gives your home router a real global routable address
- Your router gives out local addresses in a special range (192.168.*.*)
- The router maps remote addresses for each connection you make from within your home network

http://en.wikipedia.org/wiki/Network_address_translation



NAT = Network Address Translation

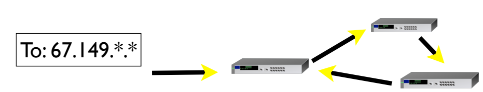
Clipart: http://www.clker.com/search/networksym/1



http://xkcd.com/742/

Peering into the Internet

- Most systems have a command that will reveal the route taken across the internet (traceroute on Mac and tracert on Windows)
- Each IP packet has a field called "Time to Live" - TTL
- The TTL is used to deal with loops in the network - normally if routers got confused and ended up with a loop - the network would clog up rapidly.



Clipart: http://www.clker.com/search/networksym/1

How Traceroute Works

- Normal packets are sent with a Time to Live (TTL) of 255 hops
- Trace route sends a packet with TTL=1, TTL=2, ...
- So each packet gets part-way there and then gets dropped and traceroute gets a notification of where the drop happens
- This builds a map of the nodes that a packet visits when crossing the Internet.

Traceroute

```
$ traceroute www.stanford.edu
traceroute to www5.stanford.edu (171.67.20.37), 64 hops max, 40 byte packets
 1  141.211.203.252 (141.211.203.252)  1.390 ms  0.534 ms  0.490 ms
 2  v-bin-seb.r-bin-seb.umnet.umich.edu (192.122.183.61)  0.591 ms  0.558 ms  0.570 ms
 3  v-bin-seb-i2-aa.merit-aa2.umnet.umich.edu (192.12.80.33)  6.610 ms  6.545 ms  6.654 ms
 4  192.122.183.30 (192.122.183.30)  7.919 ms  7.209 ms  7.122 ms
 5  so-4-3-0.0.rtr.kans.net.internet2.edu (64.57.28.36)  17.672 ms  17.836 ms  17.673 ms
 6  so-0-1-0.0.rtr.hous.net.internet2.edu (64.57.28.57)  31.800 ms  41.967 ms  31.787 ms
 7  so-3-0-0.0.rtr.losa.net.internet2.edu (64.57.28.44)  63.478 ms  63.704 ms  63.710 ms
 8  hpr-lax-hpr--i2-newnet.cenic.net (137.164.26.132)  63.093 ms  63.026 ms  63.384 ms
 9  svl-hpr--lax-hpr-10ge.cenic.net (137.164.25.13)  71.242 ms  71.542 ms  76.282 ms
10  oak-hpr--svl-hpr-10ge.cenic.net (137.164.25.9)  72.744 ms  72.243 ms  72.556 ms
11  hpr-stan-ge--oak-hpr.cenic.net (137.164.27.158)  73.763 ms  73.396 ms  73.665 ms
12  bbra-rtr.Stanford.EDU (171.64.1.134)  73.577 ms  73.682 ms  73.492 ms
13  * * *
14  www5.Stanford.EDU (171.67.20.37)  77.317 ms  77.128 ms  77.648 ms
```

Traceroute

```
$ traceroute www.msu.edu
traceroute to www.msu.edu (35.8.10.30), 64 hops max, 40 byte packets
 1  141.211.203.252 (141.211.203.252)  2.644 ms  0.973 ms  14.162 ms
 2  v-bin-seb.r-bin-seb.umnet.umich.edu (192.122.183.61)  1.847 ms  0.561 ms  0.496 ms
 3  v-bin-seb-i2-aa.merit-aa2.umnet.umich.edu (192.12.80.33)  6.490 ms  6.499 ms  6.529 ms
 4  lt-0-3-0x1.eq-chi2.mich.net (198.108.23.121)  8.096 ms  8.113 ms  8.103 ms
 5  xe-0-0-0x23.msu6.mich.net (198.108.23.213)  7.831 ms  7.962 ms  7.965 ms
 6  192.122.183.227 (192.122.183.227)  12.953 ms  12.339 ms  10.322 ms
 7  cc-t1-ge1-23.net.msu.edu (35.9.101.209)  9.522 ms  9.406 ms  9.817 ms
 8  * * *
```

Traceroute

```
$ traceroute www.pku.edu.cn
traceroute: Warning: www.pku.edu.cn has multiple addresses; using 162.105.129.104
traceroute to www.pku.edu.cn (162.105.129.104), 64 hops max, 40 byte packets
1  141.211.203.252 (141.211.203.252)  1.228 ms  0.584 ms  0.592 ms
2  v-bin-seb.r-bin-seb.umnet.umich.edu (192.122.183.61)  0.604 ms  0.565 ms  0.466 ms
3  v-bin-seb-i2-aa.merit-aa2.umnet.umich.edu (192.12.80.33)  7.511 ms  6.641 ms  6.588 ms
4  192.122.183.30 (192.122.183.30)  12.078 ms  6.989 ms  7.619 ms
5  192.31.99.133 (192.31.99.133)  7.666 ms  8.953 ms  17.861 ms
6  192.31.99.170 (192.31.99.170)  59.275 ms  59.273 ms  59.108 ms
7  134.75.108.209 (134.75.108.209)  173.614 ms  173.552 ms  173.333 ms
8  134.75.107.10 (134.75.107.10)  256.760 ms 134.75.107.18 (134.75.107.18)  256.574 ms  256.53
9  202.112.53.17 (202.112.53.17)  256.761 ms  256.801 ms  256.688 ms
10  202.112.61.157 (202.112.61.157)  257.416 ms  257.960 ms  257.747 ms
11  202.112.53.194 (202.112.53.194)  256.827 ms  257.068 ms  256.962 ms
12  202.112.41.202 (202.112.41.202)  256.800 ms  257.053 ms  256.933 ms
```

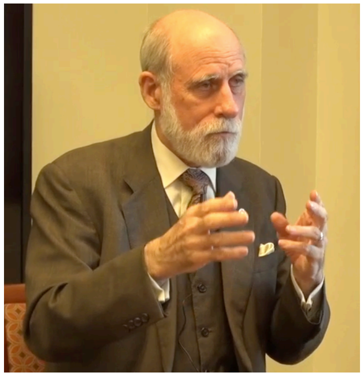
Michigan
Tennessee
Seoul
Beijing

The perfect is the enemy of the good

Le mieux est l'ennemi du bien. –Voltaire

- IP Does: Best effort to get data across bunch of hops from one network to another network
- IP Does Not: Guarantee delivery - if things go bad - the data can vanish
- Best effort to keep track of the good and bad paths for traffic - tries to pick better paths when possible
- This makes it fast and scalable to very large networks - and ultimately "reliable" because it does not try to do too much

Vint Cerf: A Brief History of Packets



- Instrumental in the design and development of the ARPANET
- Vint was a graduate student as the notions of packet-switching were emerging across academia

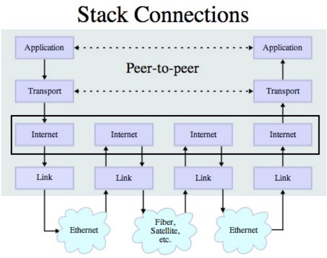
Domain Name System

The Domain Name System convert user-friendly names, like

www.umich.edu

to network-friendly IP addresses, like

141.211.32.166



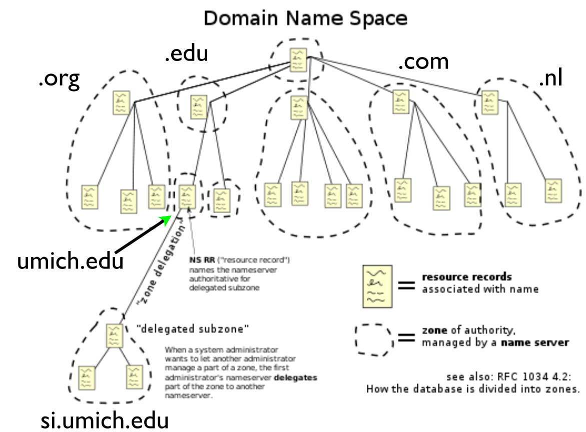
Source: http://en.wikipedia.org/wiki/Internet_Protocol_Suite

Domain Name System

- Numeric addresses like 141.211.63.45 are great for Internet routers but lousy for people
- Each campus ends up with a lot of networks (141.211.*.*, 65.43.21.*)
- Sometimes (rarely) the IP address numbers get reorganized
- When servers physically move they need new IP addresses

DNS: Internet Address Book

- The Domain Name System is a big fast distributed database of Internet names to Internet "phone numbers"
- IP Addresses reflect technical "geography"
  - 141.211.63.44 - read left to right like a phone number
- Domain names reflect organizational structure
  - www.si.umich.edu - read right to left like postal address
  - 2455 North Quad, Ann Arbor, MI, USA, Earth



http://en.wikipedia.org/wiki/File:Domain_name_space.svg

Internetwork Layer (IP)

http://en.wikipedia.org/wiki/Internet_Protocol

http://en.wikipedia.org/wiki/Traceroute

http://en.wikipedia.org/wiki/Ping

Application Layer Web, E-Mail, File Transfer
Transport Layer (TCP) Reliable Connections
Internetwork Layer (IP) Simple, Scalable, Unreliable
Link Layer (Ethernet, WiFi) Physical Connections

Transport Layer

http://en.wikipedia.org/wiki/Transmission_Control_Protocol

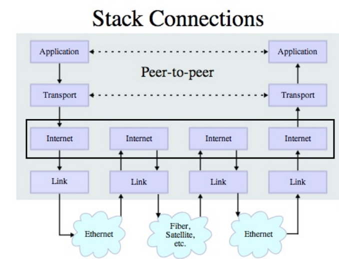
Application Layer Web, E-Mail, File Transfer
Transport Layer (TCP) Reliable Connections
Internetwork Layer (IP) Simple, Scalable, Unreliable
Link Layer (Ethernet, WiFi) Physical Connections

Review: The Magic of IP

- What it does - Tries to get one packet across a 5-20 of hops from one network to another network
- Keeps track of the good and bad paths for traffic - tries to pick better paths when possible
- But no guarantee of delivery - if things go bad - the data vanishes
- This makes it fast and scalable - and ultimately "reliable" because it does not try to do too "everything"

Internet Protocol

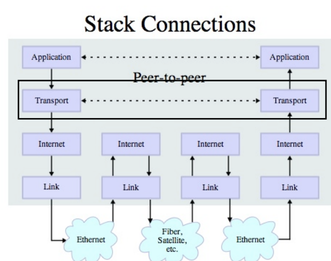
- So many links / hops
- So many routes
- Thinks can change dynamically and IP has to react (links up/ down)
- IP can drop packets



Source: http://en.wikipedia.org/wiki/Internet_Protocol_Suite

Tramsmission Protocol (TCP)

- Built on top of IP
- Assumes IP might lose some data
- In case data gets lost - we keep a copy of the data a we send until we get an acknowledgement
- If it takes "too long" - just send it again



Source: http://en.wikipedia.org/wiki/Internet_Protocol_Suite

Sender

100

200

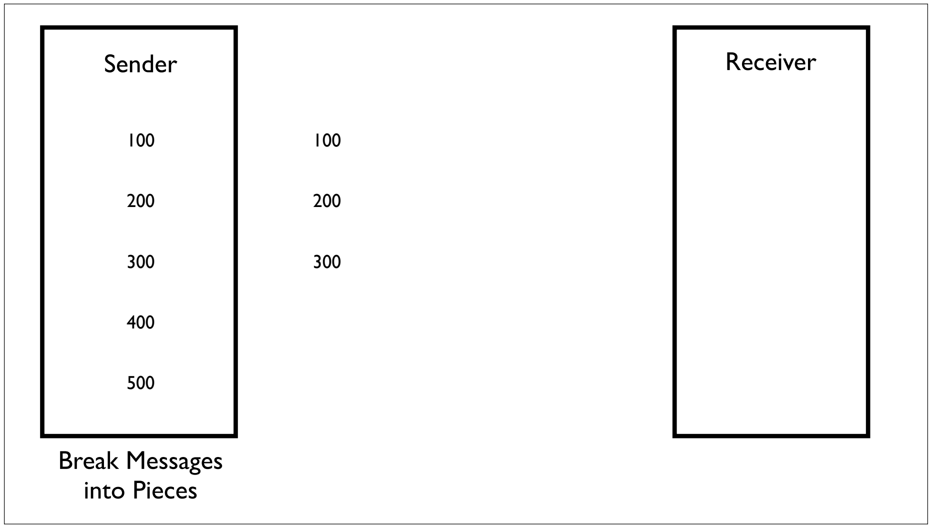
300

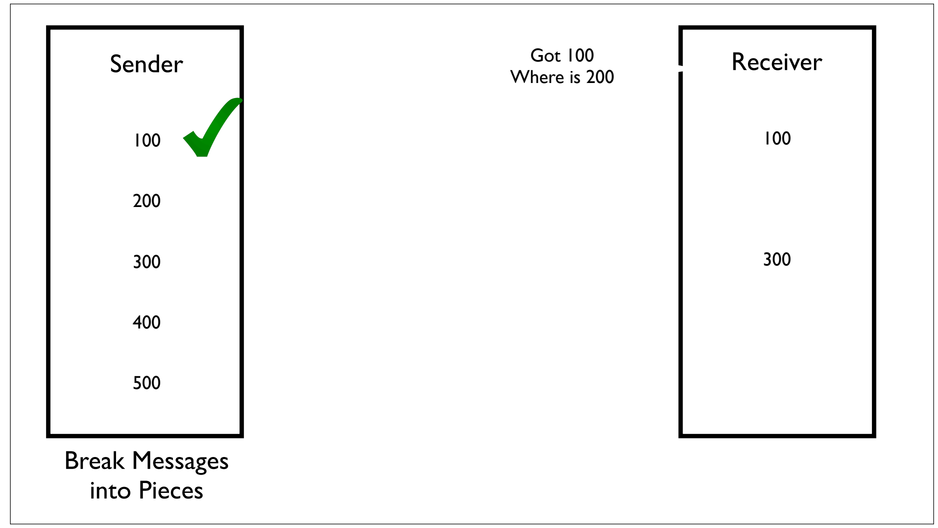
400

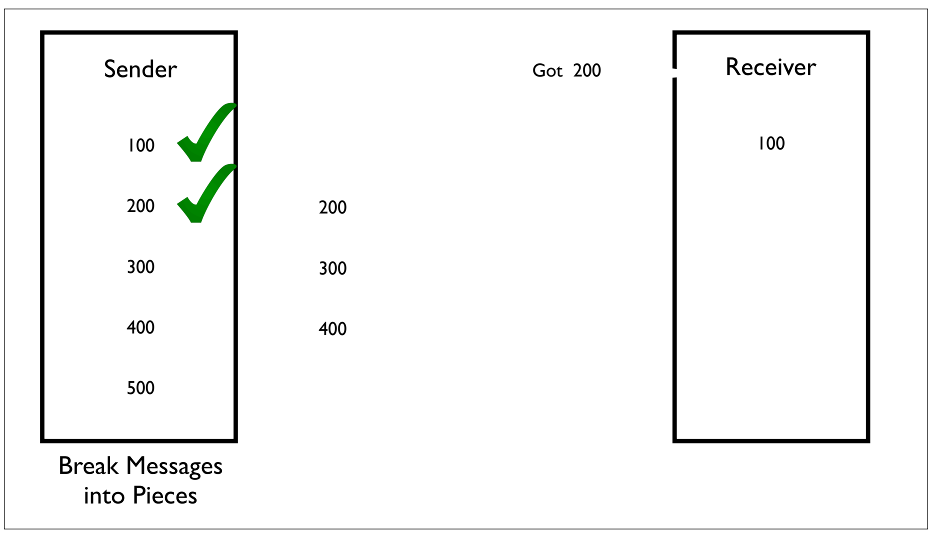
500

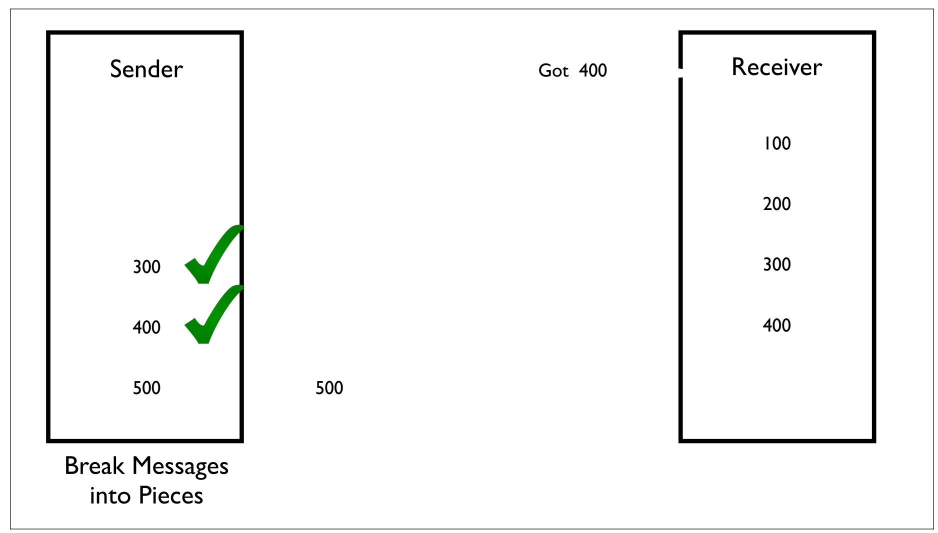
Break Messages into Pieces

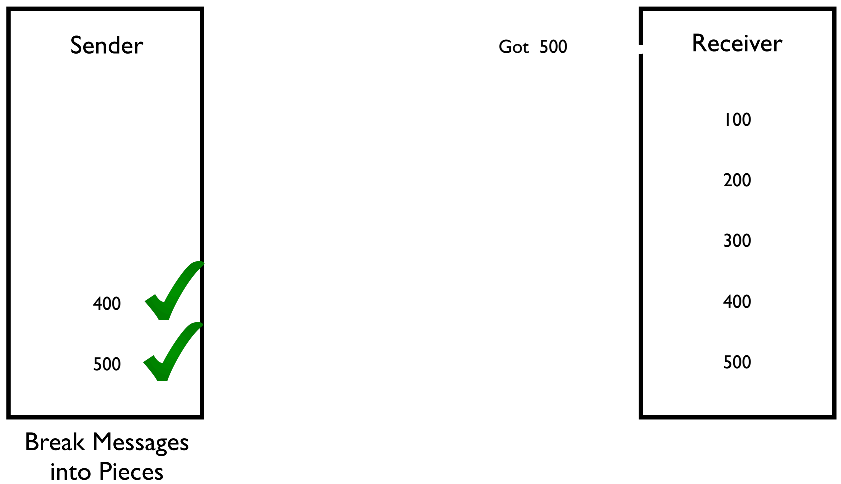
Receiver

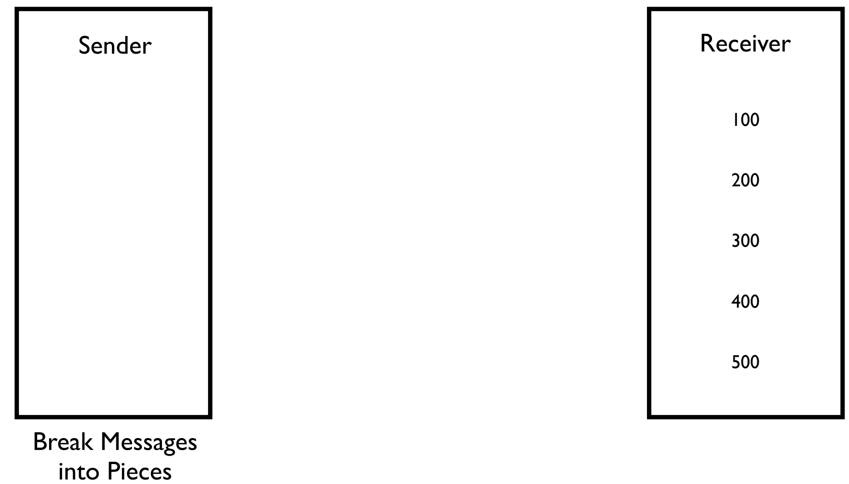

Sender
100
200
300
400
500
Break Messages into Pieces
100
200
300
Receiver

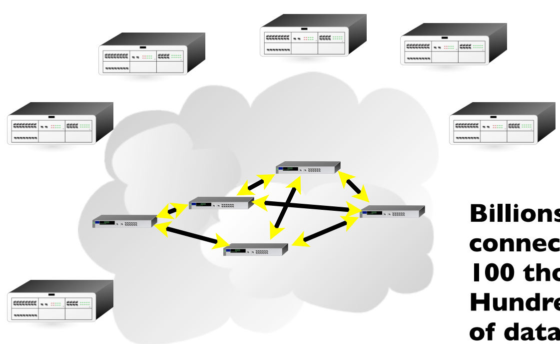

Sender
100
200
300
400
500
Break Messages into Pieces
Got 100
Where is 200
Receiver
100
300


Sender
100
200
300
400
500
Break Messages into Pieces
200
300
400
Got 200
Receiver
100


Sender
300
400
500
Break Messages into Pieces
500
Got 400
Receiver
100
200
300
400







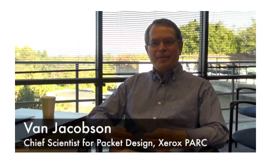
Billions of computers connected to the internet; 100 thousands of routers. Hundreds of billions bytes of data enroute at any moment.

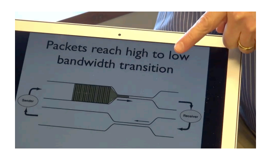
Storage of enroute data done at the edges only!

Clipart: http://www.clker.com/search/networksym/1

One (of many) Scary Problem(s)

- In 1987 as local campuses with 10 MBit networks were connected together using 56Kbit leased lines, things kind of fell apart
- At some point, when there was a little too much traffic, it all fell apart...

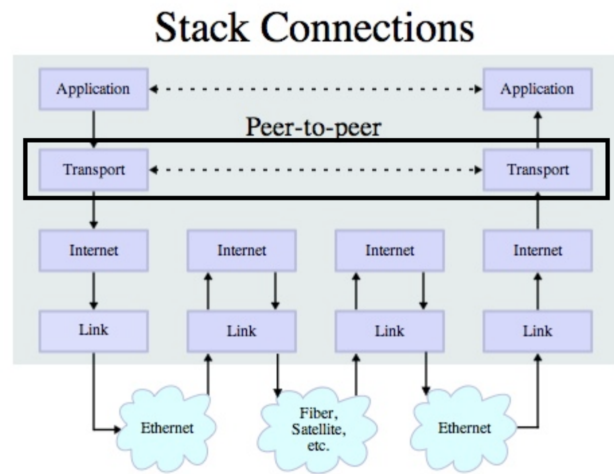




http://www.youtube.com/watch?v=1Vg1MeRYmWI
http://en.wikipedia.org/wiki/Van_Jacobson
http://en.wikipedia.org/wiki/TCP_congestion_avoidance_algorithm

Tramsmission Protocol (TCP)

- The responsibility of the transport layer is to present a reliable end-to-end pipe to the application
- Data either arrives in the proper order or the connection is closed
- TCP keeps buffers in the sending and destination system to keep data which has arrived out of order or to retransmit if necessary
- TCP provides individual connections between applications

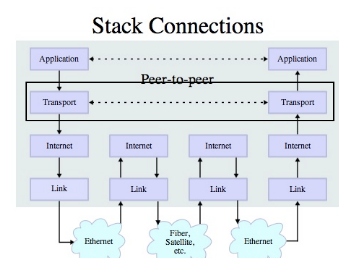


Application Layer

Application Layer Web, E-Mail, File Transfer
Transport Layer (TCP) Reliable Connections
Internetwork Layer (IP) Simple, Unreliable
Link Layer (Ethernet, WiFi) Physical Connections

Quick Review

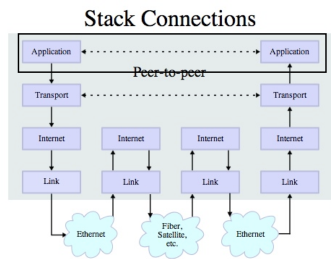
- Link layer: gets the data onto the link, and manages collisions on a single hop
- Internet layer: moves the data over one hop, trying to get it "closer" to its destination
- Transport layer: Assumes that the internet layer may lose data, so request retransmission when needed—provides a nice reliable pipe from source to destination



Source: http://en.wikipedia.org/wiki/Internet_Protocol_Suite

Application Protocol

- Since TCP gives us a reliable pipe, what to we want to do with the pipe? What problem do we want to solve?
  - Mail
  - World Wide Web
  - Stream kitty videos



Source: http://en.wikipedia.org/wiki/Internet_Protocol_Suite

Two Questions for the Application Layer

- Which application gets the data?
  - Ports
- What are the rules for talking with that application?
  - Protocols

http://en.wikipedia.org/wiki/TCP_and_UDP_port
http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers

Ports

- Like extensions in a phone number
- The IP address network number (the area code) gets to the LAN
- The IP address host number (the telephone number) gets you to the destination machine
- The port (the extension) gets you to a specific application

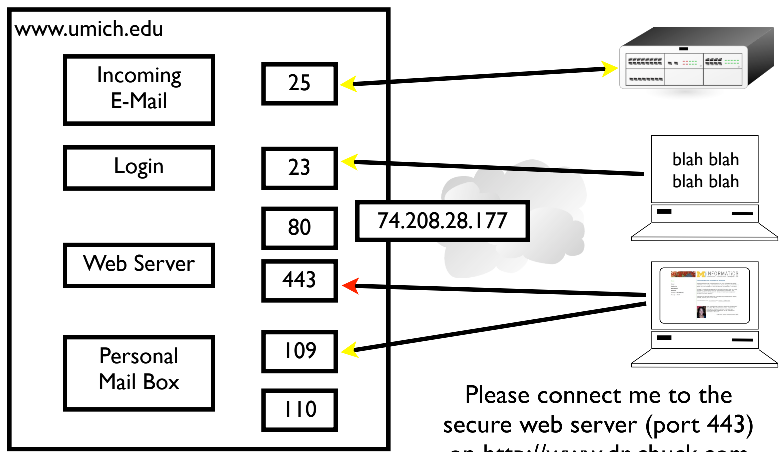
(734) 764 1855 ext. 27

141.211.144.188
Port 25

TCP, Ports, and Connections

http://en.wikipedia.org/wiki/TCP_and_UDP_port

http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers



Please connect me to the secure web server (port 443) on http://www.dr-chuck.com

Clipart: http://www.clker.com/search/networksym/1

Common TCP Ports

- Telnet (23) - Login
- SSH (22) - Secure Login
- HTTP (80)
- HTTPS (443) - Secure
- SMTP (25) (Mail)
- IMAP (143/220/993) - Mail Retrieval
- POP (109/110) - Mail Retrieval
- DNS (53) - Domain Name
- FTP (21) - File Transfer

http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers

Application Protocols

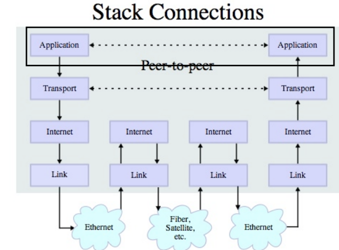
http://en.wikipedia.org/wiki/Http

http://en.wikipedia.org/wiki/Pop3

Application Protocol

- Since TCP gives us a reliable pipe, what to we want to do with the pipe? What problem do we want to solve?
  - Mail
  - World Wide Web
  - Stream kitty videos

Stack Connections



Source: http://en.wikipedia.org/wiki/Internet_Protocol_Suite

HTTP - Hypertext Transport Protocol

- The dominant Application Layer Protocol on the Internet
- Invented for the Web - to Retrieve HTML, Images, Documents etc
- Extended to be data in addition to documents - RSS, Web Services, etc..
- Basic Concept - Make a Connection - Request a document - Retrieve the Document - Close the Connection

http://en.wikipedia.org/wiki/Http

HTTP Request / Response Cycle

Web Server

HTTP Request

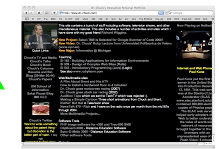
HTTP Response

Browser

Internet Explorer, FireFox, Safari, etc.

Hello there my name is Chuck

Go ahead and click on here.



http://www.oreilly.com/openbook/cgi/ch04_02.html

Source: http://www.dr-chuck.com/

HTTP Request / Response Cycle

Web Server

GET /page2.html

HTTP Request

HTTP Response

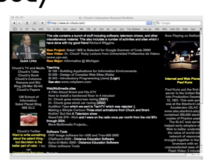
<head> .. </head>
<body>
<h1>Welcome to my application</h1>
....
</body>

Browser

Internet Explorer, FireFox, Safari, etc.

Hello there my name is Chuck

Go ahead and click on here.



http://www.oreilly.com/openbook/cgi/ch04_02.html

Source: http://www.dr-chuck.com/

Internet Standards

- The standards for all of the Internet protocols (inner workings) are developed by an organization
- Internet Engineering Task Force (IETF)
- www.ietf.org
- Standards are called "RFCs" - "Request for Comments"

```
Network Working Group                                   T. Berners-Lee
Request for Comments: 1945                                     MIT/LCS
Category: Informational                                     R. Fielding
                                                              UC Irvine
                                                             H. Frystyk
                                                                MIT/LCS
                                                               May 1996

              Hypertext Transfer Protocol -- HTTP/1.0

Status of This Memo

   This memo provides information for the Internet community.  This memo
   does not specify an Internet standard of any kind.  Distribution of
   this memo is unlimited.

IESG Note:

   The IESG has concerns about this protocol, and expects this document
   to be replaced relatively soon by a standards track document.

Abstract

   The Hypertext Transfer Protocol (HTTP) is an application-level
   protocol with the lightness and speed necessary for distributed,
   collaborative, hypermedia information systems. It is a generic,
   stateless, object-oriented protocol which can be used for many tasks,
   such as name servers and distributed object management systems,
   through extension of its request methods (commands). A feature of
   HTTP is the typing of data representation, allowing systems to be
   built independently of the data being transferred.
```

Source: http://www.ietf.org/rfc/rfc1945.txt

5.1.2 Request-URI

The Request-URI is a Uniform Resource Identifier (Section 3.2) and identifies the resource upon which to apply the request.

```
Request-URI    = absoluteURI | abs_path
```

The two options for Request-URI are dependent on the nature of the request.

The absoluteURI form is only allowed when the request is being made to a proxy. The proxy is requested to forward the request and return the response. If the request is GET or HEAD and a prior response is cached, the proxy may use the cached message if it passes any restrictions in the Expires header field. Note that the proxy may forward the request on to another proxy or directly to the server specified by the absoluteURI. In order to avoid request loops, a proxy must be able to recognize all of its server names, including any aliases, local variations, and the numeric IP address. An example Request-Line would be:

```
GET http://www.w3.org/pub/WWW/TheProject.html HTTP/1.0
```


The most common form of Request-URI is that used to identify a resource on an origin server or gateway. In this case, only the absolute path of the URI is transmitted (see Section 3.2.1, abs_path). For example, a client wishing to retrieve the resource above directly from the origin server would create a TCP connection to port 80 of the host "www.w3.org" and send the line:

```
GET /pub/WWW/TheProject.html HTTP/1.0
```

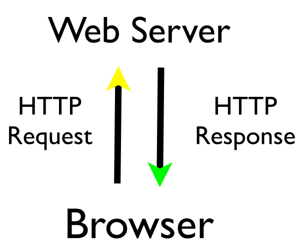
followed by the remainder of the Full-Request. Note that the absolute path cannot be empty; if none is present in the original URI, it must be given as "/" (the server root).

The Request-URI is transmitted as an encoded string, where some characters may be escaped using the "% HEX HEX" encoding defined by RFC 1738 [4]. The origin server must decode the Request-URI in order to properly interpret the request.

Source: http://www.ietf.org/rfc/rfc1945.txt

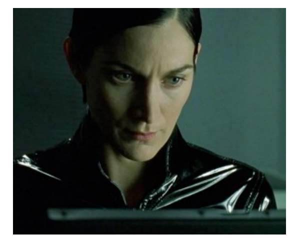
“Hacking” HTTP

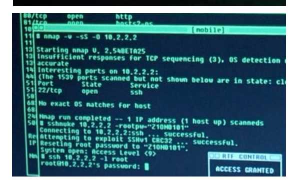
```
Last login: Wed Oct 10 04:20:19 on ttyp2
si-csev-mbp:~ csev$ telnet www.dr-chuck.com 80
Trying 74.208.28.177...
Connected to www.dr-chuck.com.
Escape character is '^]'.
GET http://www.dr-chuck.com/page1.htm
<h1>The First Page</h1>
<p>
If you like, you can switch to the
<a href="http://www.dr-chuck.com/page2.htm">
Second Page</a>.
</p>
```



Port 80 is the non-encrypted HTTP port

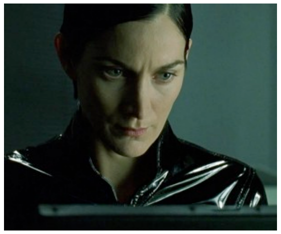
Accurate Hacking in the Movies

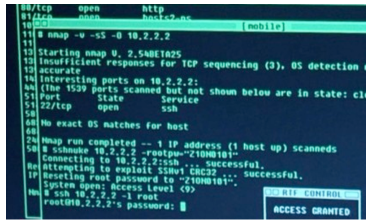




- Matrix Reloaded
- Bourne Ultimatum
- Die Hard 4
- ...

http://nmap.org/movies.html (scroll down for video)
Or search YouTube for "Trinity hacking scene"



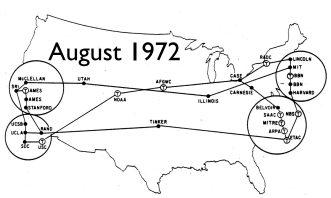


http://nmap.org/movies.html (scroll down for video)
Or search YouTube for "Trinity hacking scene"

Application Layer Summary

- We start with a "pipe" abstraction - we can send and receive data on the same "socket"
- We can optionally add a security layer to TCP using SSL - Secure Socket Layer (aka TLS - Transport Layer Security)
- We use well known "port numbers" so that applications can find a particular application *within* a server such as a mail server, web service, etc

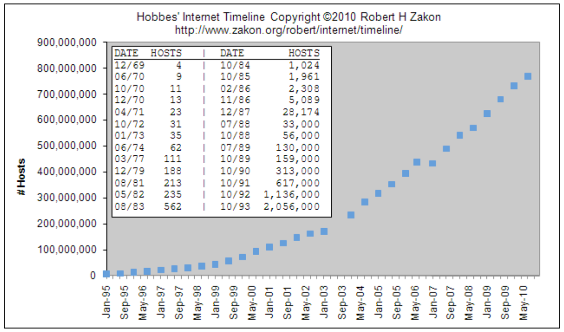
The Architecture of the Internet



Application Layer Web, E-Mail, File Transfer
Transport Layer (TCP) Reliable Connections
Internetwork Layer (IP) Simple, Unreliable
Link Layer (Ethernet, WiFi) Physical Connections

The Architecture of the Internet

Application Layer Web, E-Mail, File Transfer
Transport Layer (TCP) Reliable Connections
Internetwork Layer (IP) Simple, Unreliable
Link Layer (Ethernet, WiFi) Physical Connections



DATE	HOSTS	DATE	HOSTS
12/69	4	10/84	1,024
06/70	9	10/85	1,961
10/70	11	02/86	2,308
12/70	13	11/86	5,089
04/71	23	12/87	28,174
10/72	31	07/88	33,000
01/73	35	10/88	56,000
06/74	62	07/89	130,000
03/77	111	10/89	159,000
12/79	188	10/90	313,000
08/81	213	10/91	617,000
05/82	235	10/92	1,136,000
08/83	562	10/93	2,056,000

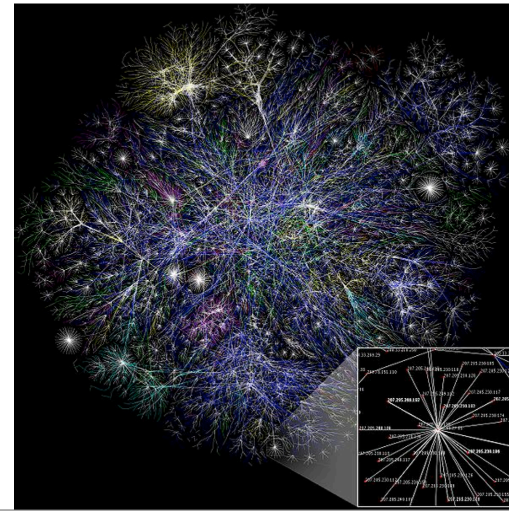
http://www.zakon.org/robert/internet/timeline/

Application Layer
Web, E-Mail, File Transfer

Transport Layer (TCP)
Reliable Connections

Internetwork Layer (IP)
Simple, Unreliable

Link Layer (Ethernet, WiFi)
Physical Connections



Application Layer
Web, E-Mail, File Transfer

Transport Layer (TCP)
Reliable Connections

Internetwork Layer (IP)
Simple, Unreliable

Link Layer (Ethernet, WiFi)
Physical Connections

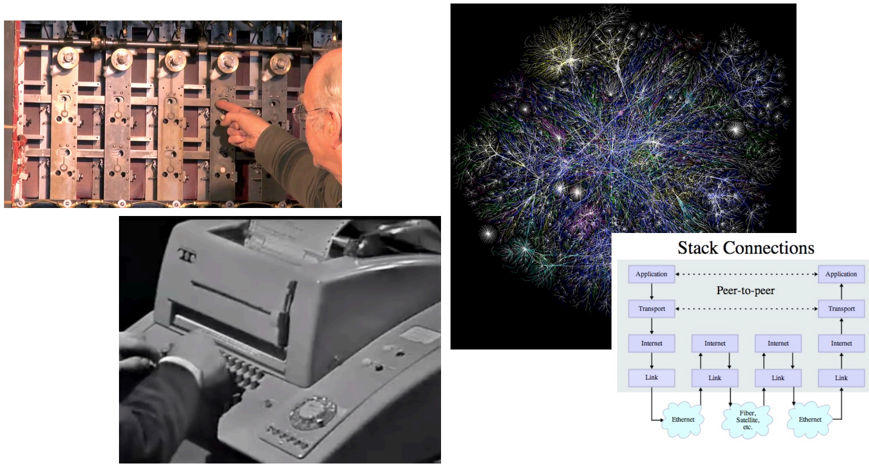
http://en.wikipedia.org/wiki/File:Internet_map_1024.jpg

The Internet: An Amazing Design

- Hundreds of millions of computers
- Thousands of routers inside the Internet
- Hundreds of millions of simultaneous connections
- Trillions of bytes of data moved per second around the world
- And it works

The Internet

- It is said that "The Internet is the largest single engineering effort ever created by mankind"
- It was created to work in an organic way - to repair itself and automatically adjust when parts fail
- No one part of the Internet knows all of the Internet (like life)
- It is never 100% up - but it seems up all the time



We are not done experimenting...

- There is still very active exploration on how network technology can be improved
- Content-Centric Networking is only one advanced idea
- Routers in the future can have *lots* of memory - lets try not to send the same piece of data more than once

Additional Source Information

- xkcd, http://xkcd.com/742/, CC: BY-NC, http://creativecommons.org/licenses/by-nc/2.5/
- Internet Protocol Suite Diagrams: Kbrose, Wikimedia Commons, http://upload.wikimedia.org/wikipedia/commons/c/c4/IP_stack_connections.svg, CC:BY-SA, http://creativecommons.org/licenses/by-sa/3.0/deed.en
- All your bases are belong to me: Karin Dalziel, Flickr, http://www.flickr.com/photos/nirak/270213335/, CC:BY, http://creativecommons.org/licenses/by/2.0/deed.en
- Internet Map: The Opte Project, Wikimedia Commons, http://upload.wikimedia.org/wikipedia/commons/d/d2/Internet_map_1024.jpg, CC:BY, http://creativecommons.org/licenses/by/2.5/deed.en

Reuse of these materials

- I intend for these materials to be reusable as open educational resources for those who would do so in a responsible manner
- Please contact me if you are interested in reusing or remixing these materials in your own teaching or educational context