Internet Technology Review

Some Terminology

internet: collection of packet switching networks interconnected by routers

(the) Internet: "public" interconnection of networks

end system = host: computer that is attached to the network ↔ router; usually *one* network interface

router = gateway = intermediate system: routes packets, several interfaces

subnetwork: part of an internet (e.g., single Ethernet)

firewall: router placed between an organization's internal internet and a connection to the external Internet, restricting packet flows to provide security.

Internet WAN Physical Layers

	Gb/s	remarks
Giga Ethernet	1.25	fiber
T-3	0.045	fiber, TP or coax
OC-3c	0.155	fiber
OC-12	0.622	fiber
OC-48	2.4	fiber
OC-192	10	fiber

Dense Wavelength Division Multiplexing

- multiple optical λ in single fiber
- 1.6 to 2 Tb/s per fiber
- interfaces typically 622 Mb/s to 10 Gb/s

Link-Layer Mechanisms Used

Roughly in order of popularity:

- ATM
- IP over SONET (synchronous optical network)
- frame relay
- gigabit Ethernet (with range extenders)
- T1, T3

Asynchronous Transfer Mode (ATM)

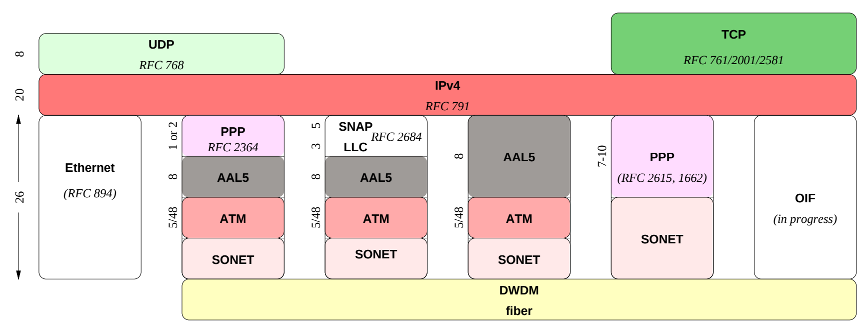
- 48-byte cells plus 5-byte header
- routing by label swapping
- virtual circuits (VCs) and paths (VPs)
- in-order delivery, but cells can be lost
- *adaptation layers*:

AAL1	continuous bit rate (CBR); “circuit emulation”
AAL2	multiplexed low-delay voice
AAL3/4	data (rarely used)
AAL5	IP packet in several cells

Frame Relay

- variable-length packets
- permanent or switched virtual circuits (PVC, SVC)
- typically, lower bandwidth ($\leq$ 45 Mb/s)
- popular as access mechanism, corporate networks

Internet Link Layers



Wireless Access

- Industrial, Scientific, Medical (ISM) bands (unlicensed): 902–928 MHz (US only), 2.4 GHz, 5.8 GHz
- analog cellular: 800 MHz
- PCS: 1.9 GHz

Wireless Ethernet:

- 900 MHz, **2.4 GHz**, or 5 GHz
- 1 or 2 Mb/s, soon 5.5 Mb/s or 11 Mb/s
- collision-based, with reservation (RTS/CTS)
- IEEE 802.11 = FH or DS

Cellular Digital Packet Data (CDPD): • pauses in AMPS voice traffic

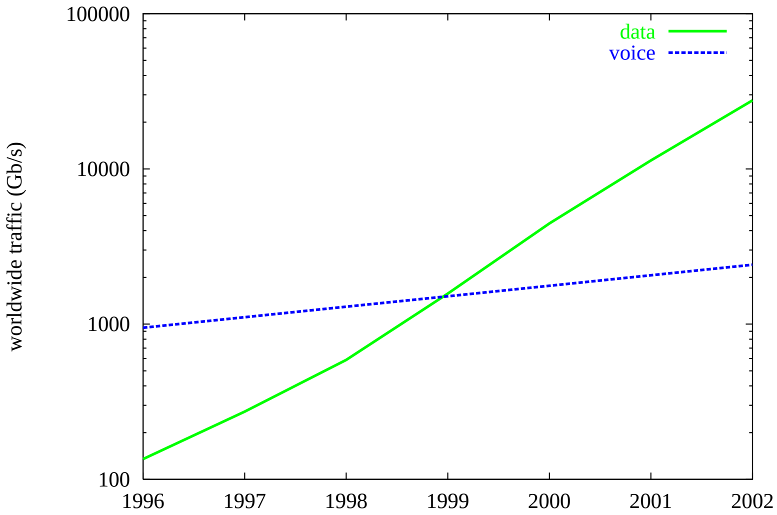
Wireless access

Technology	band	mod.	rate	open range (m)
RAM			8.0 k/bs	
GSM data	1.9 GHz	TDMA	9.6 kb/s	
CDPD			19.2 kb/s	km
Metricom Ricochet	902-928 MHz	FH	28.8 kb/s	300-450
Bluetooth	2.4 GHz	FH	432 kb/s	10
802.11	2.4 GHz	DS	1 Mb/s	540
			2 Mb/s	400
			4 Mb/s	195
			5.5 Mb/s	120

Internet Traffic

- 5,000-8,000 TB/month or 15.4–24.7 Gb/s
- long-distance calls: 525 GDEM or 64 Gb/s
- all the world's telephones: 600 Gb/s
- almost all (90%?) of the traffic is TCP

Voice vs. Data Traffic



Voice vs. Data Traffic

- local vs. LANs vs. private networks
- capacity vs. traffic
- hop length of data traffic < voice
- link utilization (higher for voice)
- revenue

Protocol Contributions

proto	src	dest	pkts	bytes
TCP	http		35%	66.4%
TCP		http	33%	7%
TCP		nntp	1.8%	3.8%
TCP	ftp		1.4%	3.2%
TCP		smtp	1.8%	1.9%
TCP	nntp		1.3%	1.5%
UDP	dns	dns	3.1%	1.0%

April 1997, NLANR

Internet Names and Addresses

Names, addresses, routes

Shoch (1979):

Name identifies what you want,

Address identifies where it is,

Route identifies a way to get there.

Saltzer (1982):

Service and users: time of day, routing, ...

Nodes: end systems and routers

Network attachment point: ≥ 1 per node ➠ multihomed host vs. router

Paths: traversal of nodes and links

binding = (temporary) equivalence of two names

Internet names and addresses

	example	organization
MAC address	8:0:20:72:93:18	flat, permanent
IP address	132.151.1.35	topological (mostly)
Host name	www.ietf.org	hierarchical

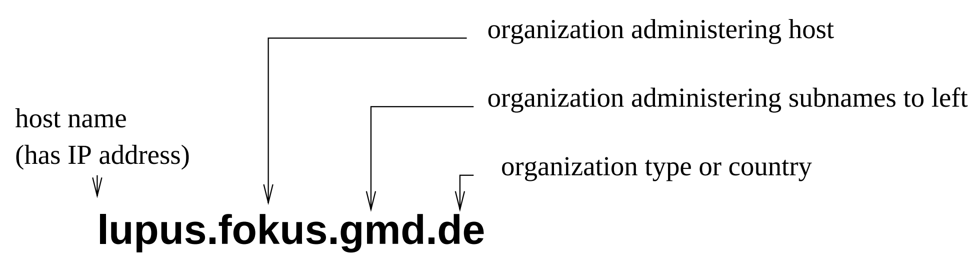
host name $\stackrel{\text{DNS,many-to-many}}{\longrightarrow}$ IP address $\stackrel{\text{ARP,1-to-1}}{\longrightarrow}$ MAC address

Mappings in the Internet

whois	domain name	owner description
LDAP	key (name)	address, other info
YP	name	data item
DNS	host name	IP addresses
	IP address	host name
atmarp	IP address	ATM NSAP
ARP	IP address	Ethernet address
RARP	MAC address	IP address

The Internet Domain Name System

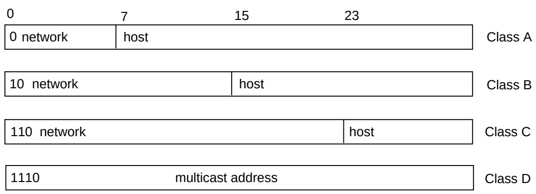
We'll talk about *name resolution* later...



Anywhere from two to ∞ parts

Internet (IP) Addresses

Each Internet host has one or more globally unique 32-bit IP addresses, traditionally consisting of a network number and a host number:



- originally, two-level hierarch $\longrightarrow$ n-level, changing
- an IP address identifies an *interface*, not a host!
- a host may have two or more addresses. Why?

Internet addresses

- (almost) every *interface* has one
- but may
  - change (dial-in)
  - have lots (WWW servers)
  - have none (some routers)
  - not be globally unique
- old: class-{A,B,C} ➠ 2-level addressing: network,host
- new: classless interdomain routing (CIDR) ➠ aggregation, route on prefix and mask

IP addresses

- dotted decimal notation: 4 decimal integers, each specifying one byte of IP address:

host name	lupus.fokus.gmd.de
32-bit address	1100 0000 0010 0011 1001 0101 0011 0100
dotted decimal	192.35.149.52

- loopback: 127.0.0.1 (packets never appear on network)
- own network (broadcast): hostid = 0; own host: netid = 0
- directed broadcast: hostid = all ones
- local broadcast: 255.255.255.255

CIDR: Classless Interdomain Routing

- problem: too many networks ➠ routing table explosion
- problem: class C too small, class B too big (and scarce)
- discard class boundaries $\longrightarrow$ supernetting
- ISP assigns a contiguous group of 2^n class C blocks
- "longest match routing" on masked address; e.g. 192.175.132.0/22

address/mask	next hop
192.175.132.0/22	1
192.175.133.0/23	2
192.175.128.0/17	3

- e.g.,: all sites in Europe common prefix ➠ only single entry in most U.S. routers

Example: ifconfig

```
ifconfig -a
le0:  flags=863<UP,BROADCAST,NOTRAILERS,RUNNING>
        inet 192.35.149.117 netmask ffffff00
        broadcast 192.35.149.0
fa0:  flags=863<UP,BROADCAST,NOTRAILERS,RUNNING>
        inet 194.94.246.72 netmask ffffff00
        broadcast 194.94.246.0
qaa0: flags=61<UP,NOTRAILERS,RUNNING>
        inet 193.175.134.117 netmask ffffff00
qaa1: flags=61<UP,NOTRAILERS,RUNNING>
        inet 129.26.216.231 netmask ffff0000
qaa2: flags=60<NOTRAILERS,RUNNING>
qaa3: flags=60<NOTRAILERS,RUNNING>
lo0:  flags=849<UP,LOOPBACK,RUNNING>
        inet 127.0.0.1 netmask ff000000
```


IP address exhaustion

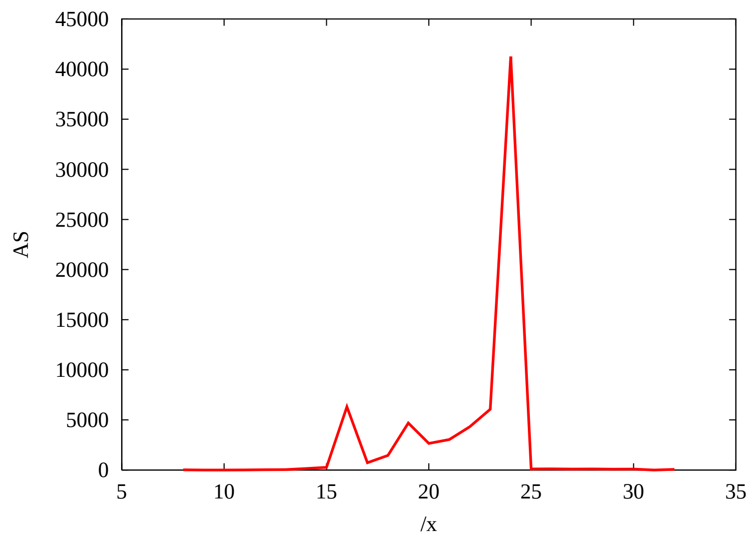
As of February 2000,

- 61.1% of available address space allocated
- 49.4% of allocated address space announced
- 30.2% of available address space announced

Routing table:

- 71,717 “autonomous system” (AS) entries
- 41,256 of which are /24

Routing Table Entries



Network Address (and Port) Translation (NA(P)T)

- most corporations use private address space, also residential
- 10/8, 172.16/12, 192.168/16
- NAT translates internal $\longleftrightarrow$ external as needed
- works for outgoing TCP connections: POP, HTTP, SMTP, Telnet
- need application layer gateway (ALG) for out-of-band protocols (ftp, SIP, RTSP, H.323, ...)
- problems:
  - controlled connections (ftp, Internet telephony, media-on-demand)
  - UDP services (streaming media)
  - security – rewriting breaks IPsec
- suggestion: Realm-Specific IP (RSIP) makes host aware of mapping

Problems with IP Addresses

- if a host moves from one network to another, its IP address changes
- currently, mostly assigned without regards to topology $\rightarrow$ too many networks ➠ CIDR
- limited space ➠ IPv6
- class thresholds: class C net grows beyond 254 hosts
- hard to change: hidden in lots of places
- multihomed host: path taken to host depends on destination address

Multihoming

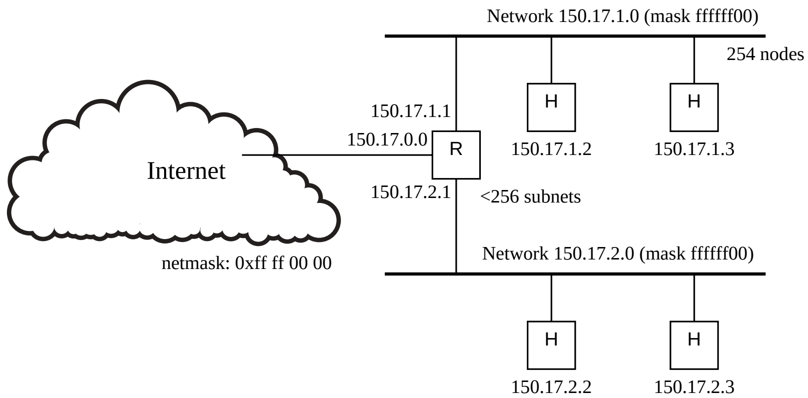
- = one "stub" network, multiple providers
- options:
  1. global prefix ➠ aggregation $\downarrow$
  2. divide network ➠ no redundancy
  3. multiple addressess ➠ applications need to try several, address space use $\uparrow$

Mobility and Renumbering

- renumber if immediate or up-stream provider changes
- mobility: change network attachment point
- mobility = renumbering: network "location" changes
- IP address as location ➠ keep address, break aggregation
- renumbering is hard: configuration files, transition
- IP address as identifier ➠ break connections

Subnetting

- large organizations: multiple LANs with single IP network address
- subdivide "host" part of network address ➠ subnetting

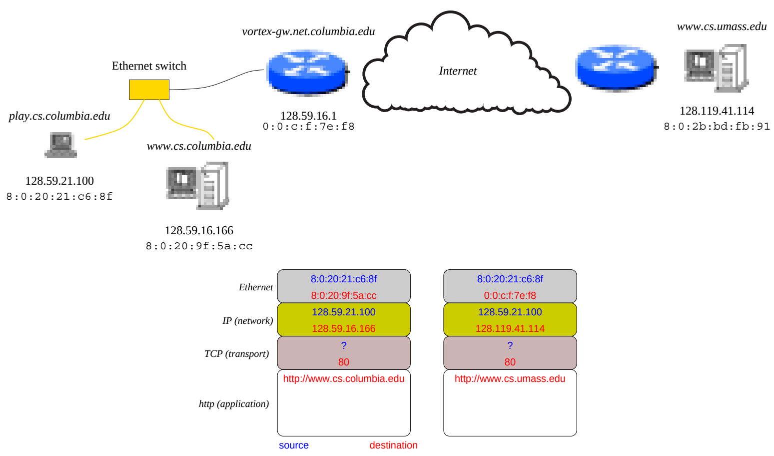


How does a packet get to the server?

E.g., web page from `http://www.cs.umass.edu`:

- get host name `www.columbia.edu` from URL;
- DNS: translate to IP address 128.59.35.60
- is it on local network? no ➠ find local router
- local router sends to Internet
- Internet routes to Columbia network router (128.59.?.?)
- Columbia router routes to web server

Peeking inside a packet

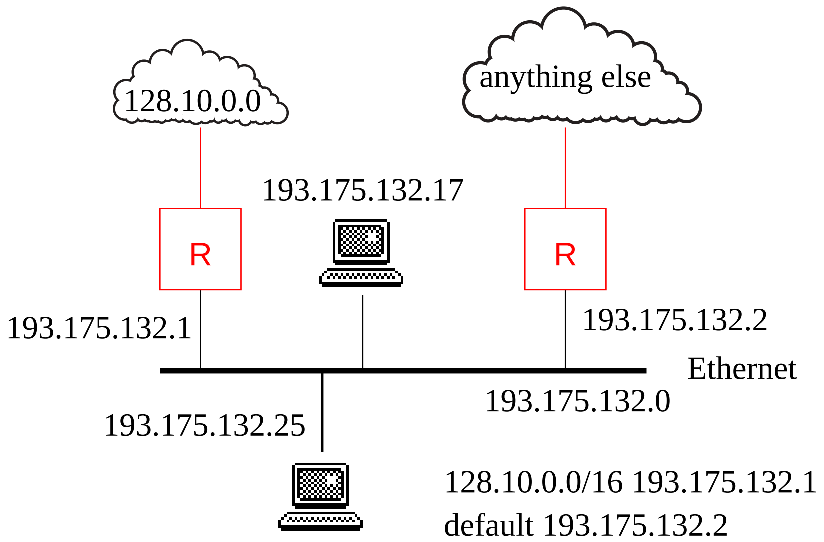


IP Forwarding

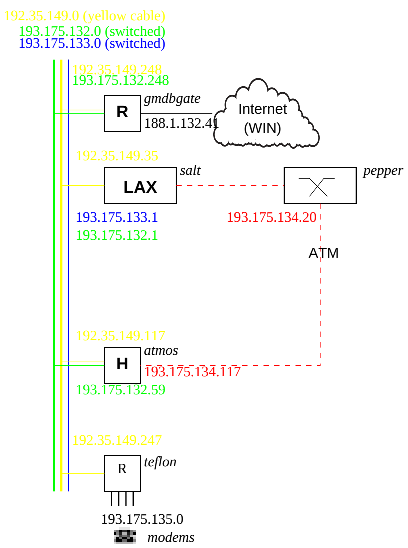
```
get destination IP address D
if network(D) == directly attached network {
  ARP: D -> MAC address
  put in link layer frame
  forward
else
  foreach entry in routing table {
    if (D & subnet mask) == network(entry) {
      get next hop address N
      ARP: N -> MAC address
      put in link layer frame
      forward
    }
  }
}
```

➠ IP source/destination remains same, MAC changes

IP Forwarding



GMD Fokus Network



Network Layer: IPv4 and IPv6

- unreliable datagram ➠ misorder, loose, duplicate
- 32-bit (IPv6: 128 bit) globally unique addresses
- no checksum on payload
- allow *fragmentation* of large packets into MTU-sized frames
- 20 (IPv6: 40) byte header
- IP multicast: receiver group with anonymous membership

IPv4

IPv4 Service Model

datagram: each packet is independent of all others

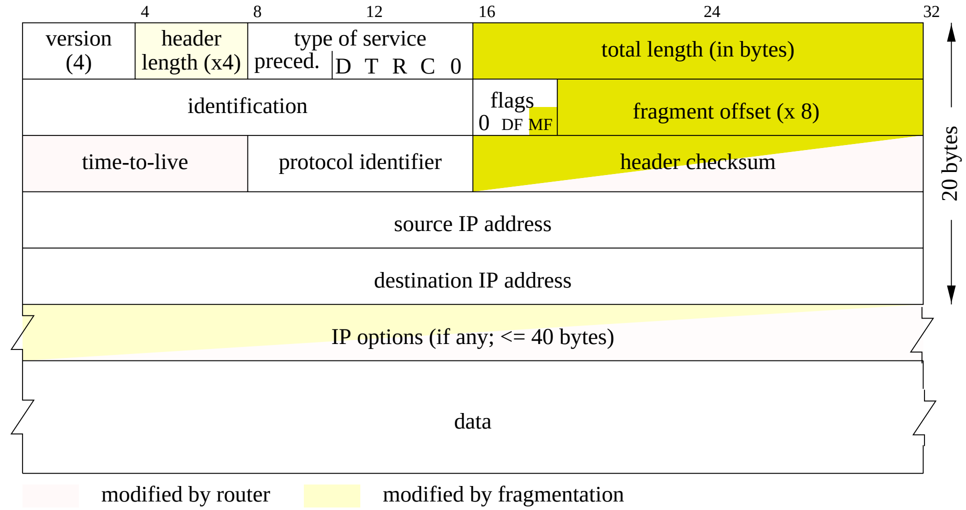
best effort: packet may arrive *or not* after some time

IPv4

- independent packets
- unreliable
- might be reordered (rare), delayed, duplicated, . . .
- but: minimal service on top of *anything* (see RFC 1149)
- only *header* checksum

IPv4 Header

RFC 791



IPv4

version: always 4

TOS (type of service): precedence (3 bits) and “minimize delay”, “maximize throughput”, “maximize reliability”, “minimize cost” bits ➠ rarely used

identifier: identifier, different for each packet from host

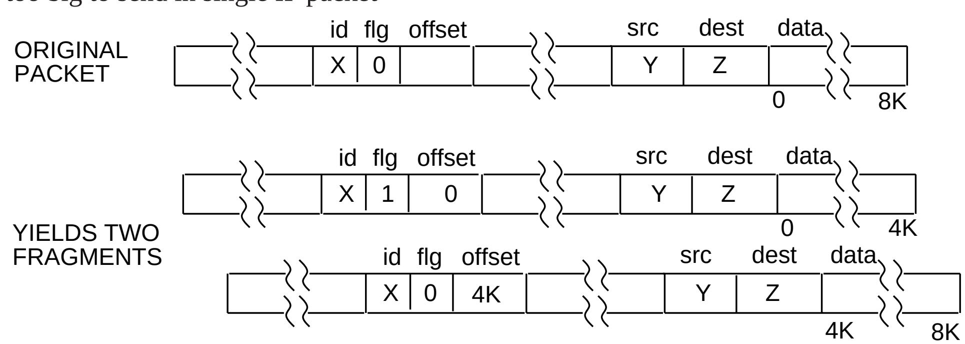
TTL: time to live field; initialized to 64; decremented at each router ➠ drop if TTL = 0 (prevent loops!)

protocol: next higher protocol (TCP: 6, UDP: 17)

header checksum: add together 16-bit words using one’s complement ➠ optimized for software

IP Fragmentation and Reassembly

data link protocol may limit packets $< 65,536$ bytes ➠ transport layer packet may be too big to send in single IP packet



IP Fragmentation and Reassembly

➠ split TPDU into *fragments*

- each fragment becomes its own IP packet (routers don't care)
- each fragment has same identifier, source, destination address
- fragment offset field gives offset of data from start of original packet
- *more fragments* (MF) flag of 0 if last (or only) fragment of packet
- fragments reassembled only at final destination
- routers must handle at least 576 bytes
- *do not fragment* bit prevents fragmentation ➠ drop + error message
- avoid multiple fragmentation (1500 $\rightarrow$ 620) ➠ MTU discovery

IP Options

Extend functionality of IP without carrying useless information:

- security and handling restrictions for military
- determine route (source route)
- record route
- record route and timestamps

(rarely used $\leftrightarrow$ not all routers support them)

IP Record Route Option

- source creates empty list of ≤ 9 IP addresses
- option: length, pointer, list of IP addresses
- routers note outgoing interface in list
- . . . and bump pointer

IP Source Route Option

- source determines path taken by packet (≤ 9 hops)
- *loose*: any number of hops in between
- *strict*: every hop; if not directly connected, discard
- same format as record route option
- router overwrites with address of outgoing interface
- must be copied to fragments
- destination should reverse route for return packets
- not too popular ➠ router performance $\downarrow$

ICMP

- used to communicate network-level error conditions and info to IP/TCP/UDP entities or user processes
- often considered part of the IP layer, but
  - IP demultiplexes up to ICMP using IP protocol field
  - ICMP messages sent within IP datagram
- ICMP contents always contain IP header and first 8 bytes of IP contents that caused ICMP error message to be generated

20−byte standard IP header	8 bit ICMP type	8 bit ICMP code	16−bit checksum	contents of ICMP msg

type	code	description
0	0	echo reply (to a ping)
3	0	destination network unreachable
3	1	destination host unreachable
3	2	destination protocol unreachable
3	3	destination port unreachable
3	4	fragmentation needed and DF set
3	6	destination network unknown
3	7	destination host unknown
3	...	other reasons
4	0	source quench (slow down)
5	1	redirect message to host
8	0	echo request (ping)
9	0	IS-ES router advertisement (new)
10	0	ES-IS router discovery (new)
11	0	time exceeded = TTL zero
12	0	IP header bad
17	0	address (subnet) mask request
18	0	address (subnet) mask reply

ping

- checks if host is reachable, alive
- uses ICMP echo request/reply
- copy packet data request → reply

```
ping -s gaia.cs.umass.edu
PING gaia.cs.umass.edu: 56 data bytes
64 bytes from gaia.cs.umass.edu (128.119.40.186): icmp_seq=0 time=276 ms
64 bytes from gaia.cs.umass.edu (128.119.40.186): icmp_seq=1 time=281 ms
64 bytes from gaia.cs.umass.edu (128.119.40.186): icmp_seq=2 time=276 ms
^C
----gaia.cs.umass.edu PING Statistics----
4 packets transmitted, 3 packets received, 25% packet loss
round-trip (ms)  min/avg/max = 276/277/281
```


traceroute

- allows to follow path taken by packet
- send UDP to unlikely port; ’time exceeded’ and ’port unreachable’ ICMP replies
- can use source route (-g), but often doesn’t work

```
$ traceroute gaia.cs.umass.edu
 1  gmdbgate (192.35.149.248)  6 ms  2 ms  2 ms
 2  188.1.132.142 (188.1.132.142)  263 ms  178 ms  188 ms
 3  gmdisgate.gmd.de (192.54.35.68)  153 ms  187 ms  151 ms
 4  icm-bonn-1.gmd.de (192.76.246.17)  226 ms  207 ms  242 ms
 5  icm-dc-1-S2/6-512k.icp.net (192.157.65.209)  320 ms  315 ms  393 ms
 6  icm-mae-e-H1/0-T3.icp.net (198.67.131.9)  372 ms  297 ms  354 ms
 7  mae-east (192.41.177.180)  456 ms  537 ms  401 ms
 8  borderx2-hssi2-0.Washington.mci.net (204.70.74.117)  529 ms  385 ms  340 ms
 9  core-fddi-1.Washington.mci.net (204.70.3.1)  437 ms  554 ms  581 ms
10  core-hssi-3.NewYork.mci.net (204.70.1.6)  418 ms  547 ms  492 ms
11  core-hssi-3.Boston.mci.net (204.70.1.2)  453 ms  595 ms  724 ms
12  border1-fddi-0.Boston.mci.net (204.70.2.34)  789 ms  404 ms  354 ms
13  nearnet.Boston.mci.net (204.70.20.6)  393 ms  323 ms  346 ms
14  mit3-gw.near.net (192.233.33.10)  340 ms  465 ms  399 ms
15  umass1-gw.near.net (199.94.201.66)  557 ms  316 ms  369 ms
16  lgrc-gw.gw.umass.edu (192.80.83.1)  396 ms  309 ms  389 ms
17  cs-gw.cs.umass.edu (128.119.44.1)  276 ms  490 ms  307 ms
18  gaia.cs.umass.edu (128.119.40.186)  335 ms  317 ms  350 ms
```

ARP: IP address → MAC address

- for broadcast networks like Ethernet, token ring, . . .
- if MAC address unknown, send ARP request and hold on to packet
- ARP request → broadcast: sender IP, MAC; target IP, MAC
- *all* machines update their cache ➠ efficiency, allow change of interface
- ARP reply → requestor: reverse source/target; fill in source MAC
- directly on Ethernet, *not* IP!
- cache ARP replies; drop after 20 minutes

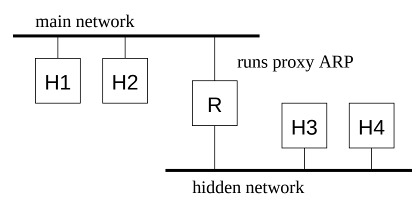
ARP example

```
arp -a
Net to Media Table
Device IP Address               Mask  Flags   Phys Addr
------ ----------- --------------- ----- ---------------
le0    hamlet      255.255.255.255       08:00:09:70:7d:16
le0    gaia        255.255.255.255       08:00:20:20:07:03
le0    pern        255.255.255.255       08:00:20:20:75:3c
le0    kite        255.255.255.255       08:00:09:92:0d:d1
le0    condor      255.255.255.255       08:00:20:1c:95:ed
```


RARP: MAC → IP address

- determine IP address at boot for diskless workstations
- remember: MAC address is unique and permanent
- host broadcasts RARP request (with its own MAC address)
- RARP server responds with reply
- allows third-party queries
- want several servers for reliability

Proxy ARP



- extend network: router fronts for H3, H4
- router answers ARP requests for H3, H4 from H1, H2 with its *own* hardware address
- assumes trusting relationship
- only needs to be added to single router
- only works for broadcast networks

Transport Layer: UDP and TCP

- UDP service = IP service + checksum + *ports*
- TCP service = UDP service + flow control + congestion control + sequenced, reliable byte stream
- ~~TCP~~ for multimedia:
  - loss recovery delay (RTT + ϵ)
  - windowed flow/congestion control ➠ variable bandwidth
  - no multicast

Internet Domain Names

The Internet Domain Name System (DNS)

- hierarchical, dot-separated names
- ➠ multi-level delegation
- by country and by type of organization
- needs to be overhauled (59% of all domains = .com!)

Global top-level domains (gTLDs):

2 letters: countries

3 letters: independent of geography (except edu, gov, mil)

domain	usage	example	domains (9/01)
com	business (global)	research.att.com	22,373,097
edu	U.S. 4 yr colleges	cs.columbia.edu	6,587
net	network provider	nis.nsf.net	4,244,092
mil	U.S. military	arpa.mil	
gov	U.S. non-military gov't	whitehouse.gov	1,217
org	non-profit orgs (global)	www.ietf.org	2,688,657
us	U.S. geographical	ietf.cnri.reston.va.us	56
uk	United Kingdom	cs.ucl.ac.uk	
de	Germany	fokus.gmd.de	

Example

```
server 128.9.0.107
Default Server:  b.root-servers.net
Address:  128.9.0.107

> erlang.cs.columbia.edu
Server:  b.root-servers.net
Address:  128.9.0.107

Name:    erlang.cs.columbia.edu
Served by:
- CUNIXD.CC.COLUMBIA.edu
          128.59.35.142
          COLUMBIA.edu
- DNS2.ITD.UMICH.edu
          141.211.125.17
          COLUMBIA.edu
```


Domain Name Resolution

- hierarchy of redundant servers with time-limited cache
- each server knows the 13 root servers `a.root-servers.net`
- each root server knows gTLDs and refers queries to those
- each domain has $\geq$ 2 servers, often widely distributed
- also: mailbox translation
- *almost* a distributed database