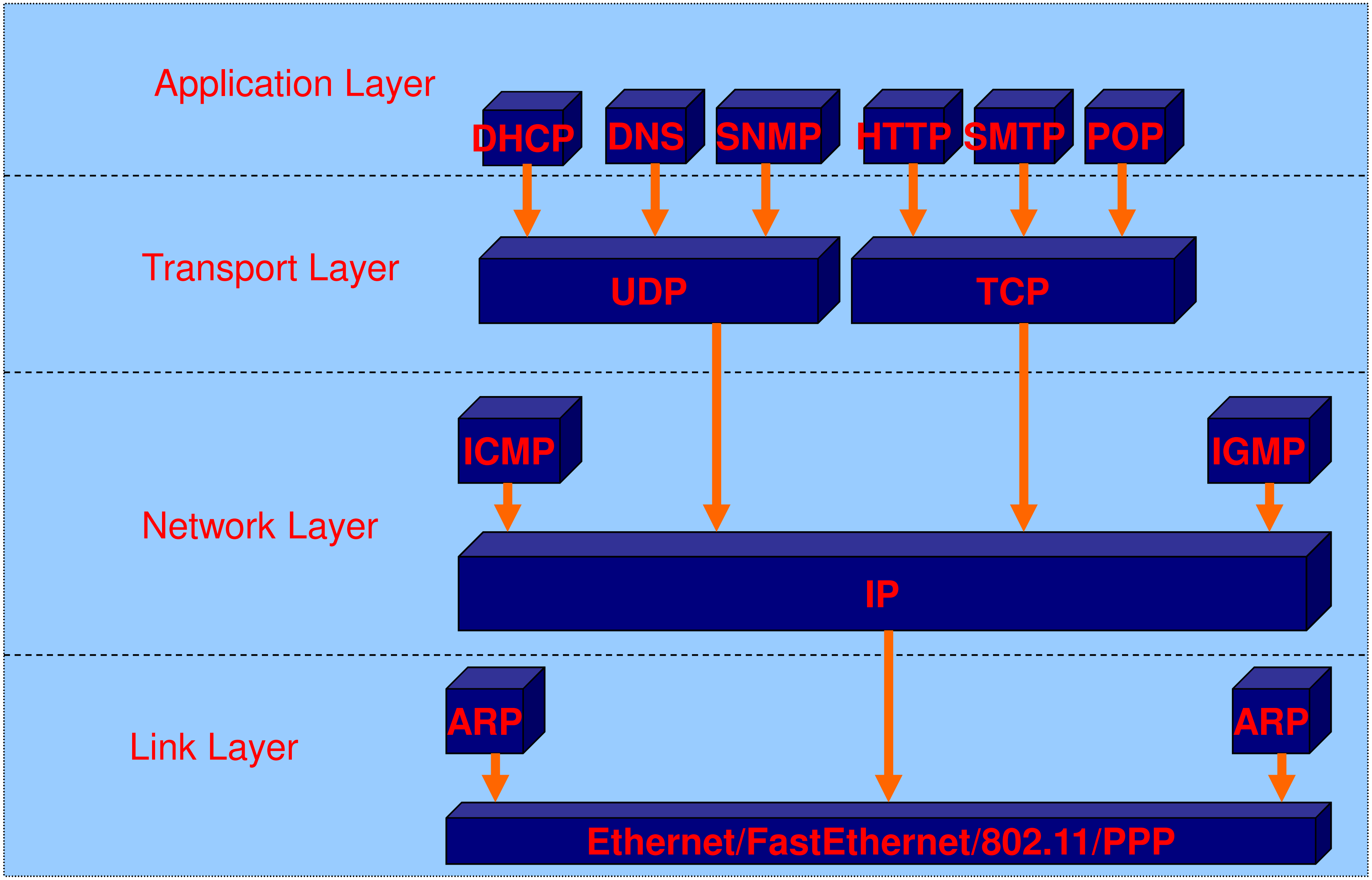
Domain Name System (DNS)

RFC 1034
RFC 1035
http://www.ietf.org

TCP/IP Protocol Suite



DNS: Domain Name System

People: many identifiers:

- □ SSN, name, Passport #

Internet hosts, routers:

- □ IP address (32 bit) - used for addressing datagrams
- □ "name", e.g., gaia.cs.umass.edu - used by humans

Q: map between IP addresses and name ?

Domain Name System:

- ■ *distributed database* implemented in hierarchy of many *name servers*
- ■ *application-layer protocol* host, routers, name servers to communicate to *resolve* names (address/name translation)
  - □ note: core Internet function implemented as application-layer protocol
  - □ complexity at network's "edge"

DNS name servers

Why not centralize DNS?

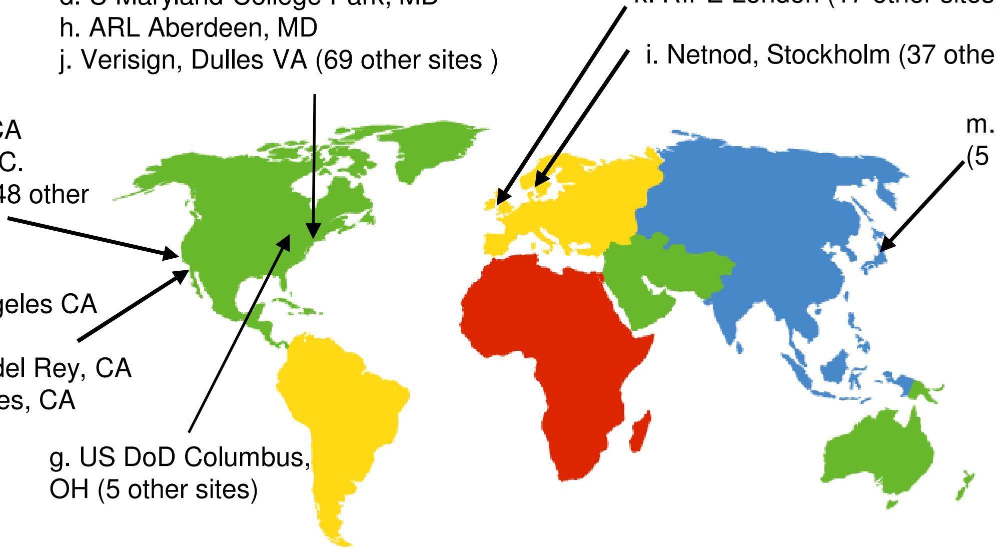
- single point of failure
- traffic volume
- distant centralized database
- maintenance

- doesn’t *scale!*

- no server has all name-to-IP address mappings
- local name servers:
  - each ISP, company has *local (default) name server*
  - host DNS query first goes to local name server
- authoritative name server:
  - for a host: stores that host’s IP address, name
  - can perform name/address translation for that host’s name

DNS: root name servers

- contacted by local name server that can not resolve name
- root name server:
  - contacts authoritative name server if name mapping not known
  - gets mapping
  - returns mapping to local name server



c. Cogent, Herndon, VA (5 other sites)
d. U Maryland College Park, MD
h. ARL Aberdeen, MD
j. Verisign, Dulles VA (69 other sites)

k. RIPE London (17 other sites)

i. Netnod, Stockholm (37 other sites)

e. NASA Mt View, CA
f. Internet Software C.
Palo Alto, CA (and 48 other sites)

m. WIDE Tokyo
(5 other sites)

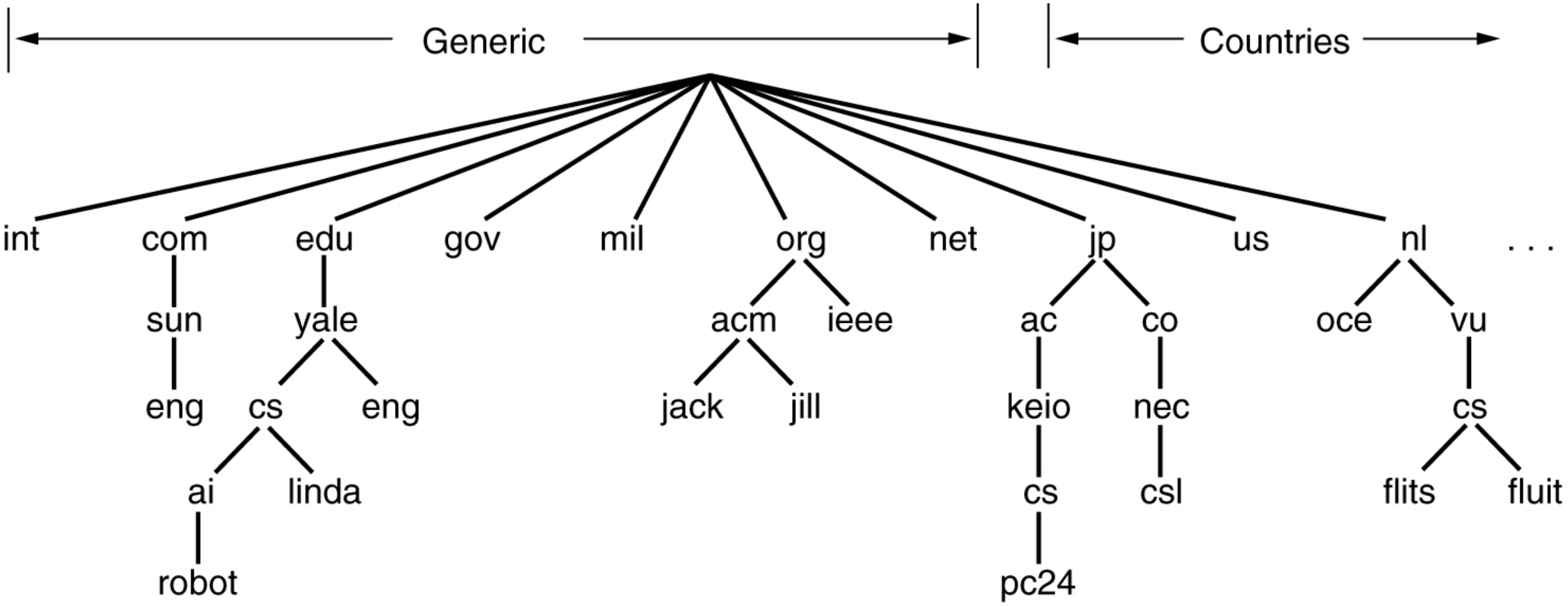
a. Verisign, Los Angeles CA
(5 other sites)
b. USC-ISI Marina del Rey, CA
l. ICANN Los Angeles, CA
(41 other sites)

g. US DoD Columbus, OH (5 other sites)

13 root name "servers" worldwide

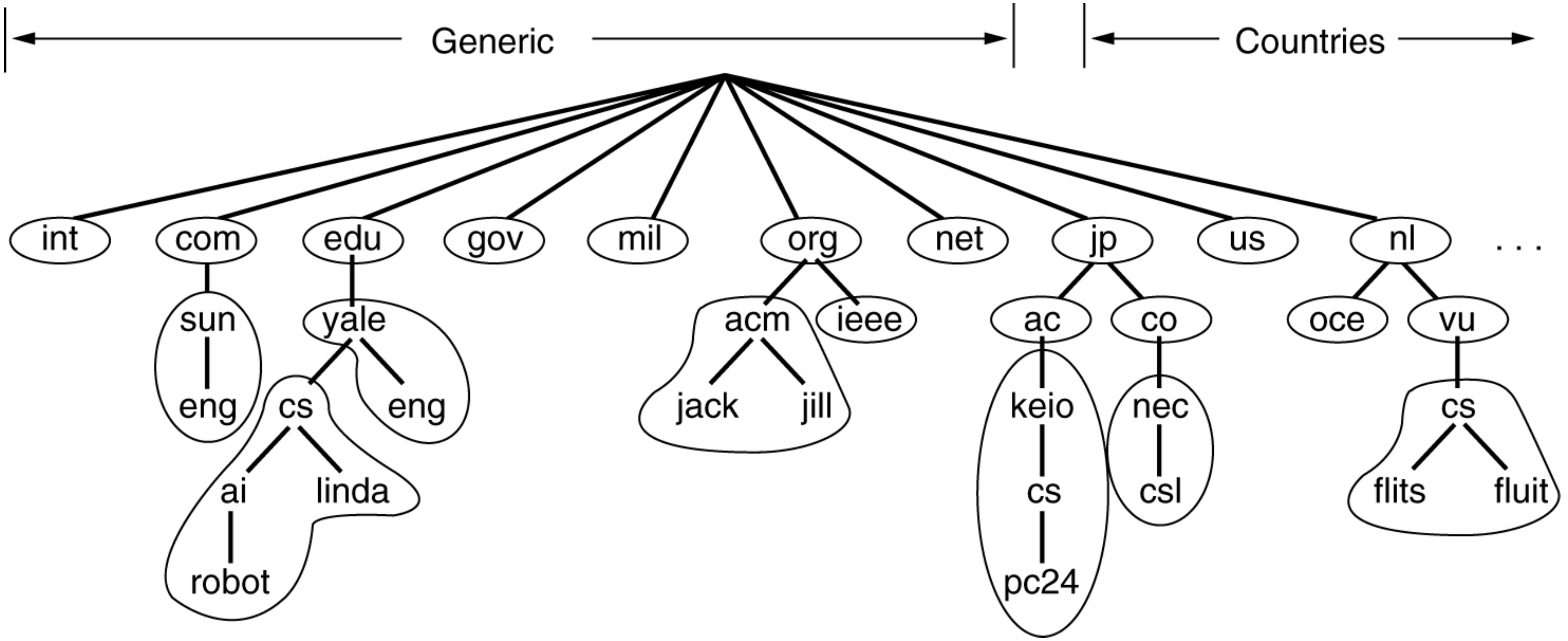
The DNS Name Space

A portion of the Internet domain name space showing some top Level Domains (TLDs).



Name Servers

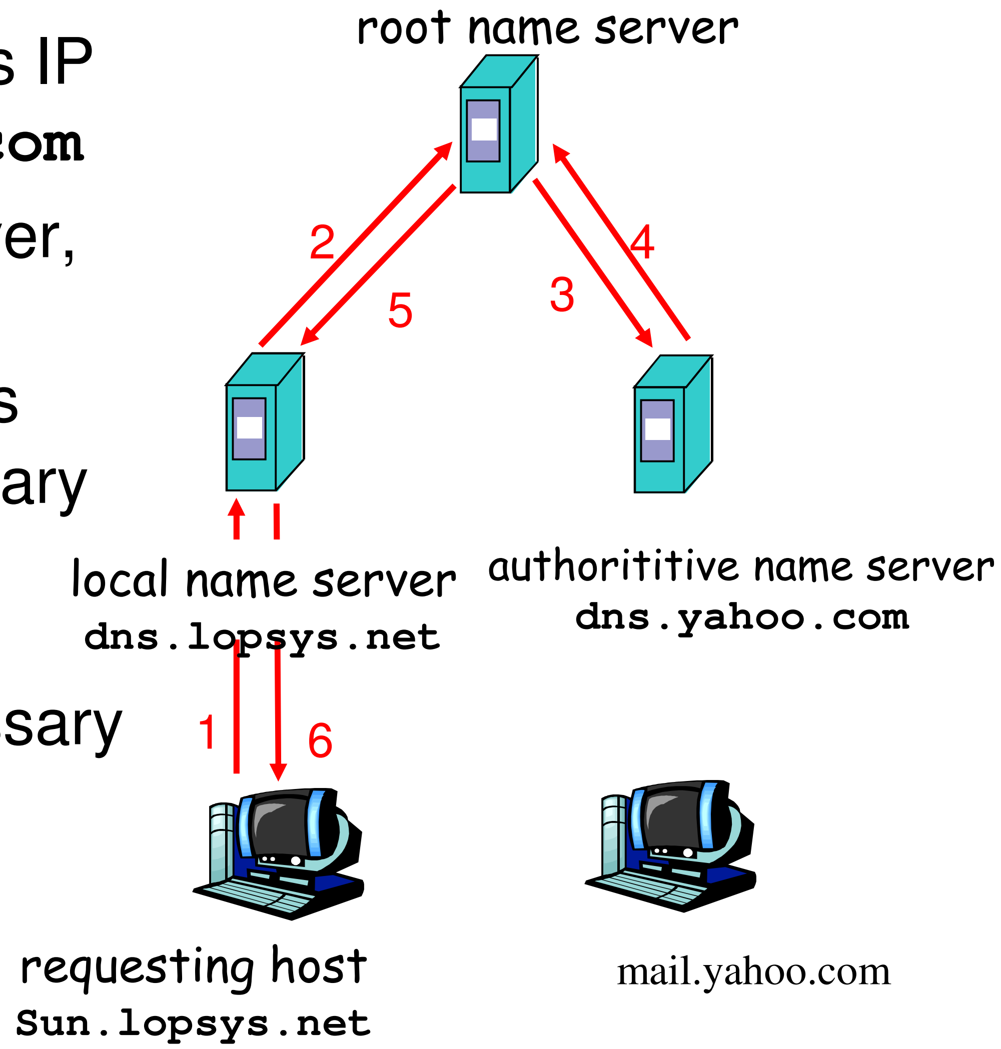
Part of the DNS name space showing the division into zones.



Simple DNS example

host **sun.lopsys.net** wants IP address of **mail.yahoo.com**

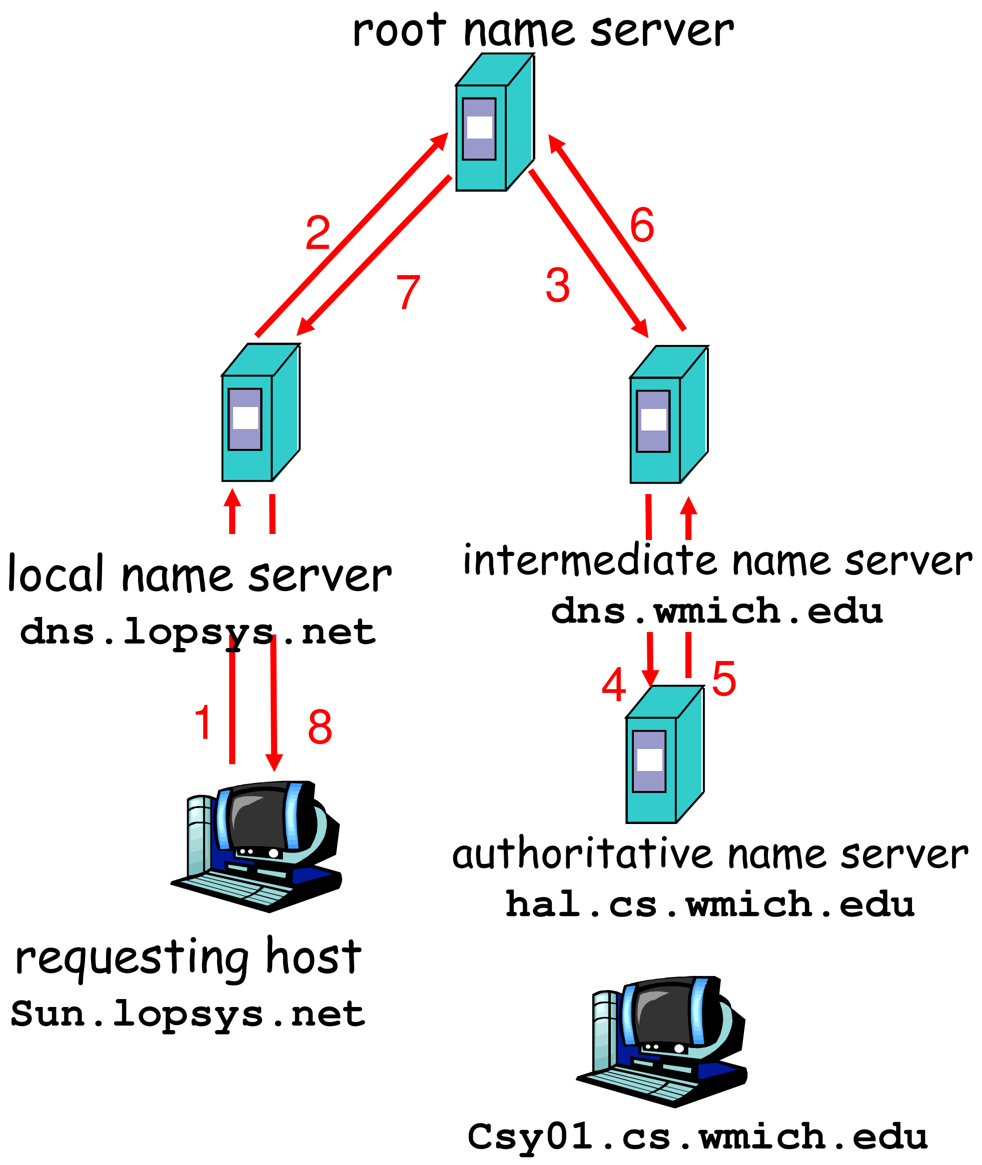
1. Contacts its local DNS server, **dns.lopsys.net**
2. **dns.lopsys.net** contacts root name server, if necessary
3. root name server contacts authoritative name server, **dns.yahoo.com**, if necessary



DNS example

Root name server:

- may not know authoratiative name server
- may know *intermediate name server:* who to contact to find authoritative name server



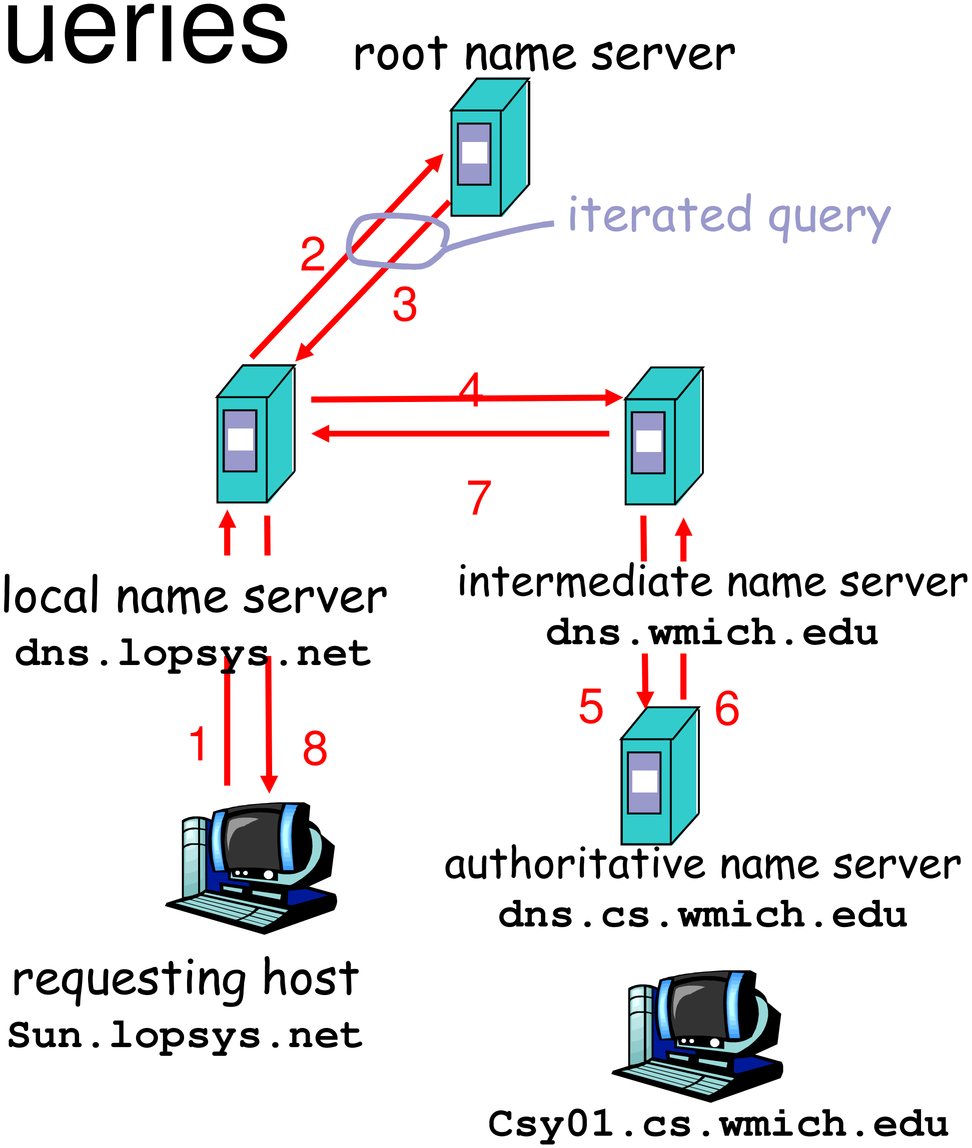
DNS: Iterated queries

recursive query:

- puts burden of name resolution on contacted name server
- heavy load?

iterated query:

- contacted server replies with name of server to contact
- “I don’t know this name, but ask this server”



DNS: caching and updating records

- once (any) name server learns mapping, it *caches* mapping
  - cache entries timeout (disappear) after some time (TTL usually 24 hours)

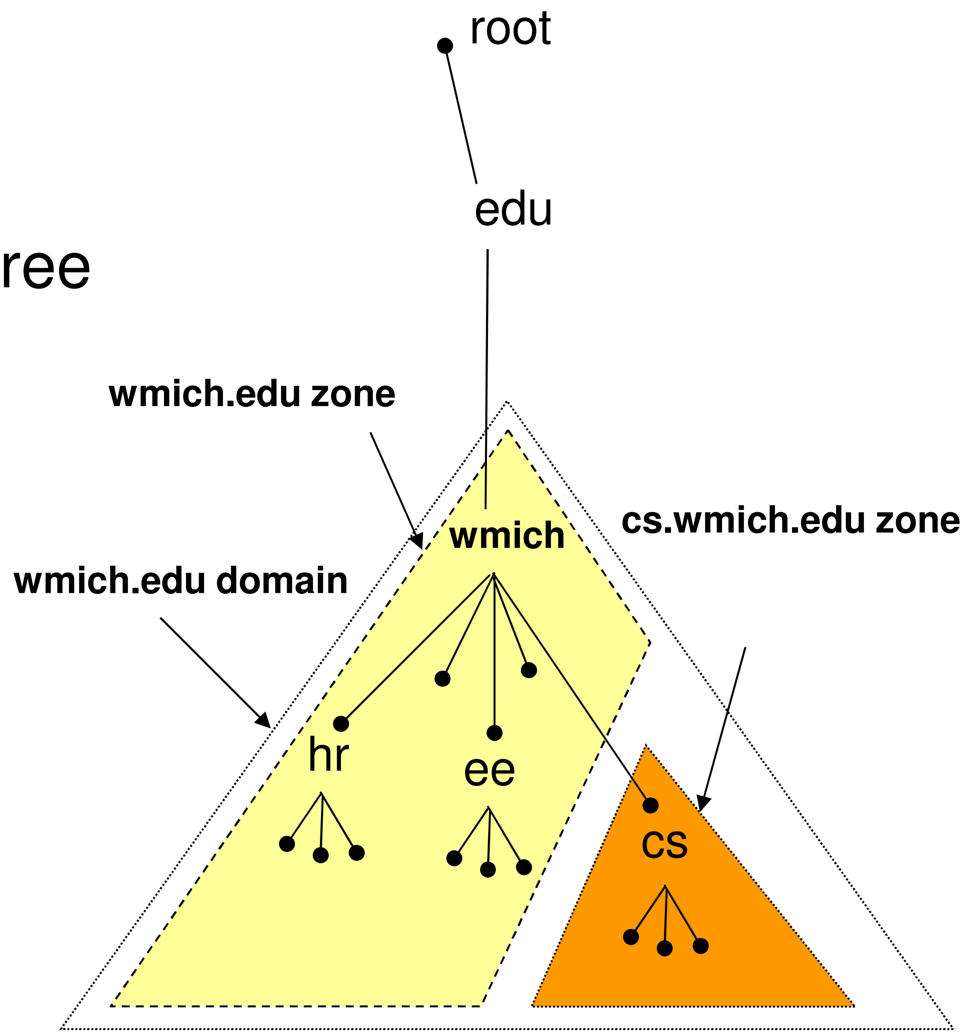
- update/notify mechanisms under design by IETF
  - RFC 2136
  - http://www.ietf.org/html.charters/dnsind-charter.html

Domains, Zones, Authority, Delegation

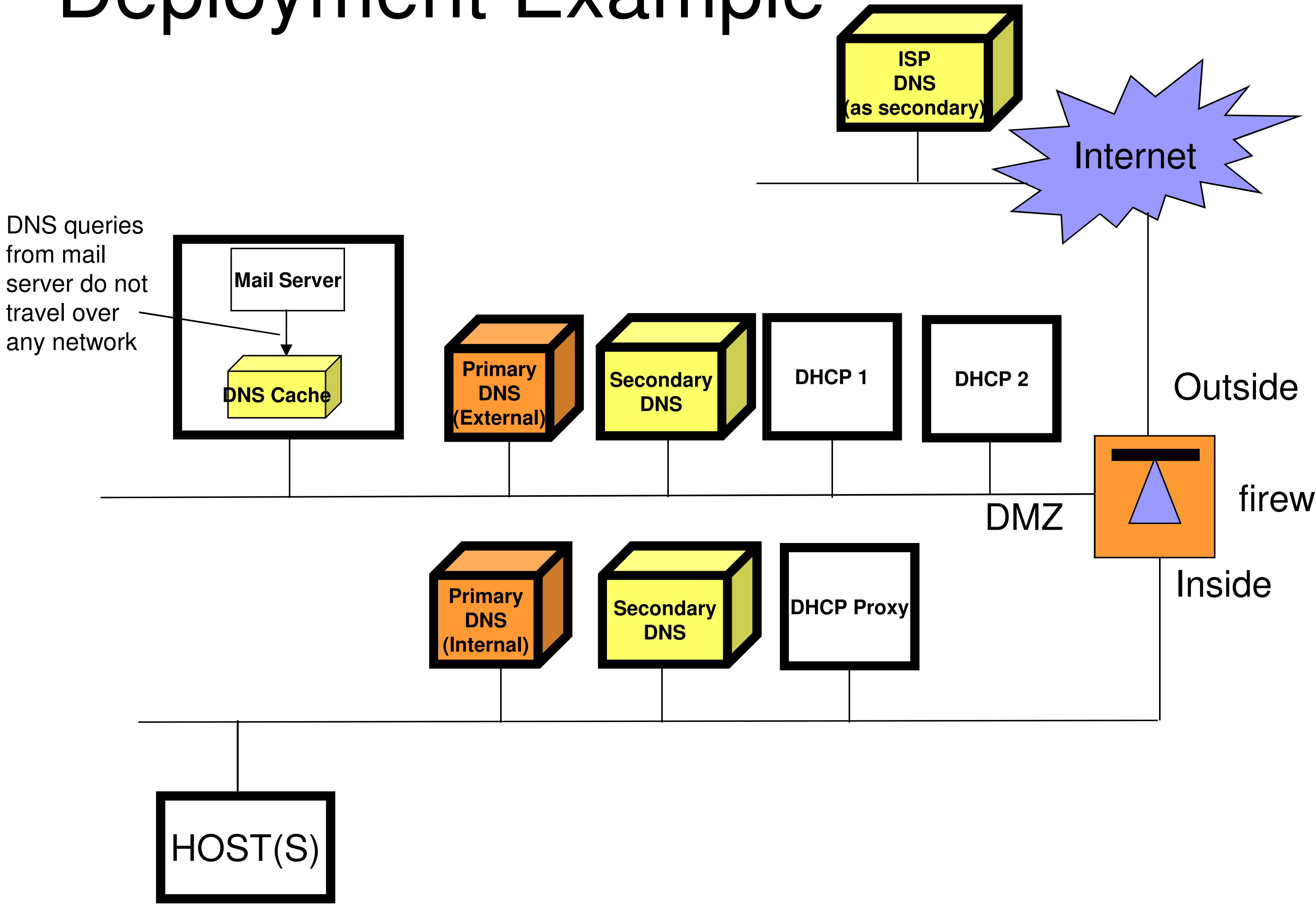
- Domain: is a node in the DNS tree, which includes all the nodes (domains) underneath it.

- Zone: is a portion of the DNS tree that a particular DNS server is ***authoritative*** for.

- A DNS Server may ***delegate*** authority of its subdomains to other organizations or departments.



Deployment Example

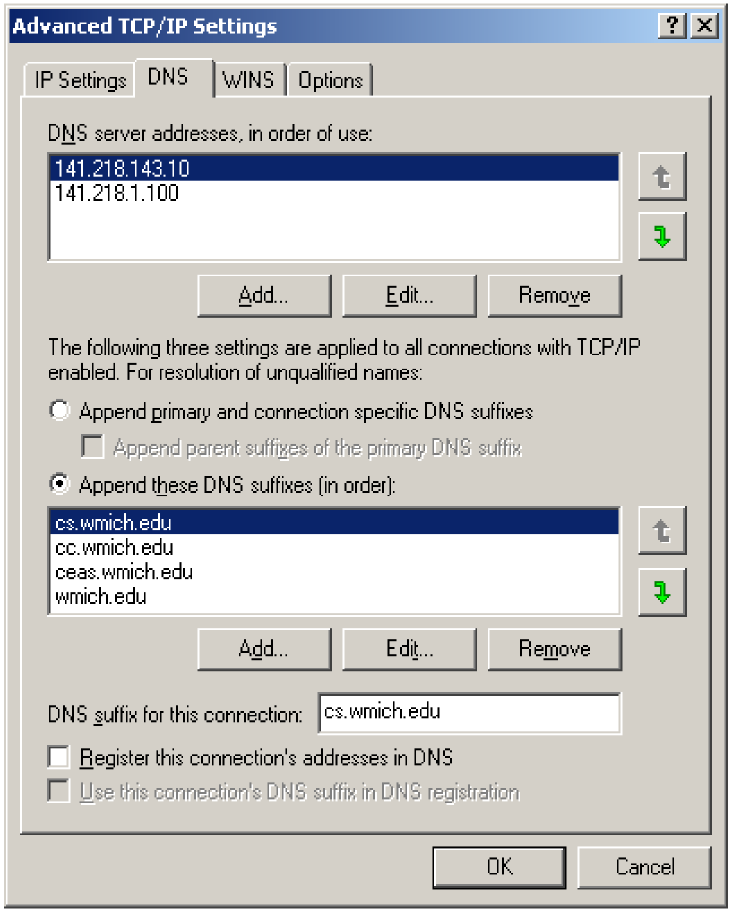


DNS Clients (resolver configuration)

- A DNS client is called a *resolver*.
- A call to **`getByName()`** is handled by a resolver (typically part of the client).

UNIX: /etc/resolv.conf

```
nameserver 141.218.143.12
nameserver 141.218.40.10
nameserver 141.218.1.100
domain cs.wmich.edu
```



Advanced TCP/IP Settings

DNS server addresses, in order of use:

141.218.143.10
141.218.1.100

The following three settings are applied to all connections with TCP/IP enabled. For resolution of unqualified names:

○ Append primary and connection specific DNS suffixes

□ Append parent suffixes of the primary DNS suffix

● Append these DNS suffixes (in order):

cs.wmich.edu
cc.wmich.edu
ceas.wmich.edu
wmich.edu

DNS suffix for this connection: cs.wmich.edu

□ Register this connection's addresses in DNS

□ Use this connection's DNS suffix in DNS registration

DNS Servers

- The name of the DNS server in UNIX is ***named***
- The configuration file for ***named*** can be found usually in /etc/named.conf
- The zone files are usually kept in /var/named with all the the zone resource records (e.g., A, PTR, MX, NS, CNAME).
- BIND (Berkeley Internet Name Domain) is an common implementation of DNS server, source code and binaries are freely available http://www.isc.org

DNS records

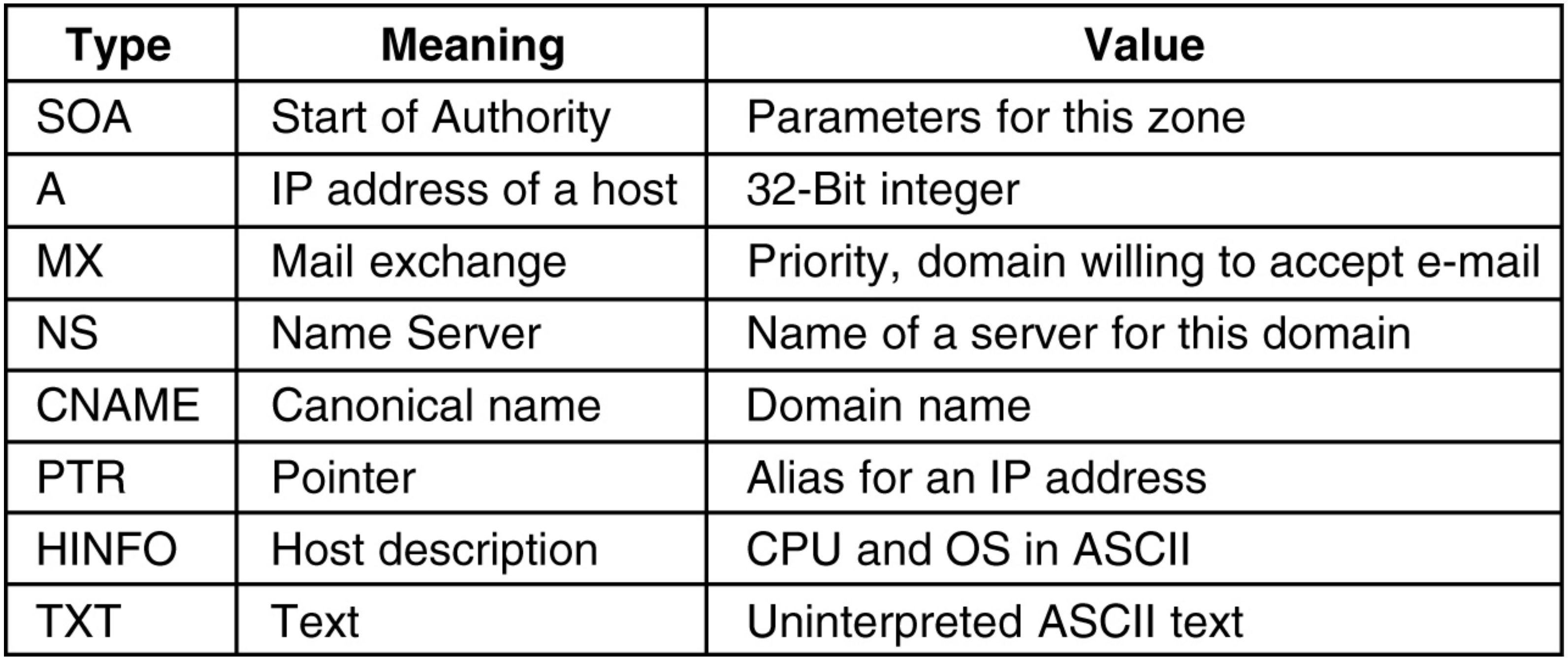
DNS: distributed db storing resource records (RR)

RR format: `(name, value, type, ttl)`

- Type=A
  - **`name`** is hostname
  - **`value`** is IP address
- Type=NS
  - **`name`** is domain (e.g. foo.com)
  - **`value`** is IP address of authoritative name server for this domain
- Type=CNAME
  - **`name`** is an alias name for some “cannonical” (the real) name
  - **`value`** is cannonical name
- Type=MX
  - **`value`** is hostname of mailserver associated with **`name`**

Resource Records

The principal DNS resource records types.



Type	Meaning	Value
SOA	Start of Authority	Parameters for this zone
A	IP address of a host	32-Bit integer
MX	Mail exchange	Priority, domain willing to accept e-mail
NS	Name Server	Name of a server for this domain
CNAME	Canonical name	Domain name
PTR	Pointer	Alias for an IP address
HINFO	Host description	CPU and OS in ASCII
TXT	Text	Uninterpreted ASCII text

Resource Records (2)

```
; Authoritative data for cs.vu.nl
cs.vu.nl.          86400   IN   SOA      star boss (952771,7200,7200,2419200,86400)
cs.vu.nl.          86400   IN   TXT      "Divisie Wiskunde en Informatica."
cs.vu.nl.          86400   IN   TXT      "Vrije Universiteit Amsterdam."
cs.vu.nl.          86400   IN   MX       1 zephyr.cs.vu.nl.
cs.vu.nl.          86400   IN   MX       2 top.cs.vu.nl.

flits.cs.vu.nl.    86400   IN   HINFO    Sun Unix
flits.cs.vu.nl.    86400   IN   A        130.37.16.112
flits.cs.vu.nl.    86400   IN   A        192.31.231.165
flits.cs.vu.nl.    86400   IN   MX       1 flits.cs.vu.nl.
flits.cs.vu.nl.    86400   IN   MX       2 zephyr.cs.vu.nl.
flits.cs.vu.nl.    86400   IN   MX       3 top.cs.vu.nl.
www.cs.vu.nl.      86400   IN   CNAME    star.cs.vu.nl
ftp.cs.vu.nl.      86400   IN   CNAME    zephyr.cs.vu.nl

rowboat                    IN   A        130.37.56.201
                           IN   MX       1 rowboat
                           IN   MX       2 zephyr
                           IN   HINFO    Sun Unix

little-sister              IN   A        130.37.62.23
                           IN   HINFO    Mac MacOS

laserjet                   IN   A        192.31.231.216
                           IN   HINFO    "HP Laserjet IIISi" Proprietary
```

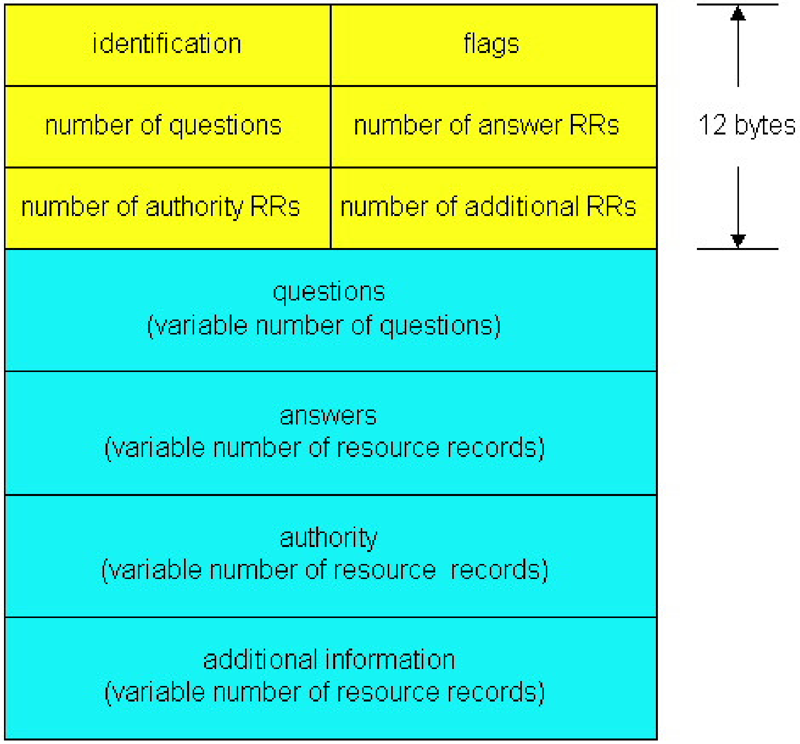
A portion of a possible DNS database for *cs.vu.nl.*

DNS protocol, messages

DNS protocol : *query* and *reply* messages, both with same *message format*

msg header

- identification: 16 bit # for query, reply to query uses same #
  - flags:
    - query or reply
    - recursion desired
    - recursion available
    - reply is authoritative



identification	flags
number of questions	number of answer RRs
number of authority RRs	number of additional RRs

12 bytes

questions
(variable number of questions)

answers
(variable number of resource records)

authority
(variable number of resource records)

additional information
(variable number of resource records)

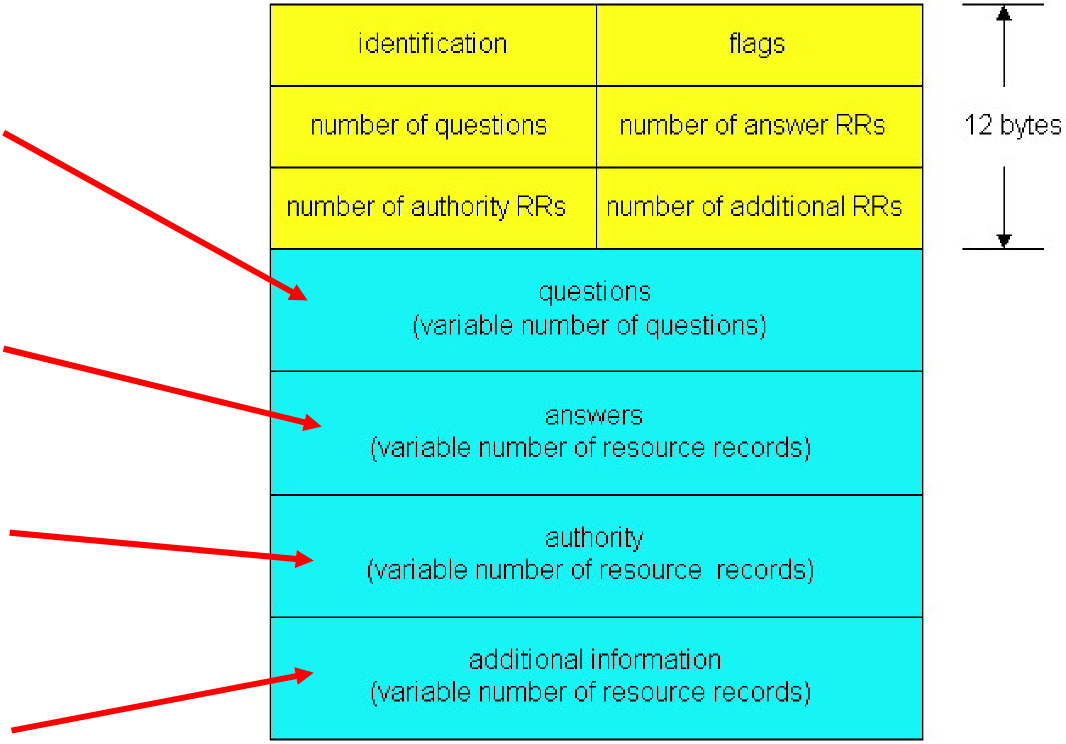
DNS protocol, messages

Name, type fields for a query

RRs in response to query

records for authoritative servers

additional "helpful" info that may be used



identification	flags
number of questions	number of answer RRs
number of authority RRs	number of additional RRs
questions (variable number of questions)	
answers (variable number of resource records)	
authority (variable number of resource records)	
additional information (variable number of resource records)	

nslookup

```
$ nslookup -d csy01.cs.wmich.edu

------------
Got answer:
   HEADER:
      opcode = QUERY, id = 6, rcode = NOERROR
      header flags:  response, auth. answer, want
recursion, recursion avail.
      questions = 1,  answers = 1,  authority records = 4,
additional = 4

   QUESTIONS:
      csy01.cs.wmich.edu, type = A, class = IN
   ANSWERS:
   ->  csy01.cs.wmich.edu
      internet address = 141.218.143.215
      ttl = 14400 (4 hours)
   AUTHORITY RECORDS:
   ->  cs.wmich.edu
      nameserver = gumby.cc.wmich.edu
      ttl = 14400 (4 hours)
   ->  cs.wmich.edu
      nameserver = hal.cs.wmich.edu
      ttl = 14400 (4 hours)
   ADDITIONAL RECORDS:
   ->  gumby.cc.wmich.edu
      internet address = 141.218.20.114
      ttl = 3120 (52 mins)
   ->  hal.cs.wmich.edu
      internet address = 141.218.143.10
      ttl = 14400 (4 hours)
------------
Name:    csy01.cs.wmich.edu
Address:  141.218.143.215
```

```
$ nslookup -querytype=MX cnn.com

Server:  hal.cs.wmich.edu
Address:  141.218.143.10

Non-authoritative answer:
cnn.com MX preference = 10, mail exchanger = atlmail1.turner.com
cnn.com MX preference = 10, mail exchanger = atlmail4.turner.com
cnn.com MX preference = 20, mail exchanger = atlmail2.turner.com
cnn.com MX preference = 30, mail exchanger = nymail1.turner.com
cnn.com MX preference = 5, mail exchanger = atlmail3.turner.com

com     nameserver = a.gtld-servers.net
com     nameserver = g.gtld-servers.net
com     nameserver = h.gtld-servers.net
com     nameserver = c.gtld-servers.net
com     nameserver = i.gtld-servers.net
com     nameserver = b.gtld-servers.net
com     nameserver = d.gtld-servers.net
com     nameserver = l.gtld-servers.net
com     nameserver = f.gtld-servers.net
com     nameserver = j.gtld-servers.net
com     nameserver = k.gtld-servers.net
com     nameserver = e.gtld-servers.net
com     nameserver = m.gtld-servers.net
atlmail1.turner.com     internet address = 64.236.240.146
atlmail4.turner.com     internet address = 64.236.221.5
atlmail2.turner.com     internet address = 64.236.240.147
nymail1.turner.com      internet address = 64.236.170.7
nymail1.turner.com      internet address = 64.236.170.8
atlmail3.turner.com     internet address = 64.236.240.169
g.gtld-servers.net     internet address = 192.42.93.30
h.gtld-servers.net     internet address = 192.54.112.30
```

Inserting records into DNS

- example: new startup “Network Utopia”
- register name networkuptopia.com at *DNS registrar* (e.g., Network Solutions)
  - □ provide names, IP addresses of authoritative name server (primary and secondary)
  - □ registrar inserts two RRs into .com TLD server:

    (networkutopia.com, dns1.networkutopia.com, NS)

    (dns1.networkutopia.com, 212.212.212.1, A)
- create authoritative server type A record for www.networkuptopia.com;
  - □ type MX record for networkutopia.com

Attacking DNS

DDoS attacks

- Bombard root servers with traffic
  - Not successful to date
  - Traffic Filtering
  - Local DNS servers cache IPs of TLD servers,
    - allowing root server bypass
- Bombard TLD servers
  - Potentially more dangerous

Redirect attacks

- Man-in-middle
  - Intercept queries
- DNS poisoning
  - Send bogus replies to DNS server,
    - which caches

Exploit DNS for DDoS

- Send queries with spoofed source address: target IP
  - Requires