Domain Name System

Mapping names to (IP) addresses

- distributed: multiple servers cooperate
- efficient: most mappings done locally
- general purpose: in theory, any mapping
- reliable: no single point of failure; multiple (≥ 2) servers for each *zone of authority*

➠ implemented by *name servers* (hosts) arranged in tree, used by *resolver* clients

Review of domain names

- hierarchical tree: host.department.company.country
- depth varies from 2 to > 5
- can't tell from name what's a domain and what's a host
- important top-level domains: .com, .edu, .org, countries (.de, .uk)
- each level *delegates* authority to lower levels

Name resolution

- hierarchy top-down, search bottom-up (most searches local?)
- client contacts local server (hard wired `/etc/resolv.conf`)
- each server knows address of root server
- server *may* know *parent* server (one level up)
- local server conceptually walks tree top down
- query: name, type of answer, flag: recursive/iterative
- response: either complete answer or next server to contact

Caching

- can't contact root server for every query
- each server must maintain cache, hosts may
- if server has cached copy, return *non-authoritative* mapping, plus source of information
- answers include time-to-live (TTL) value ➠ decrease TTL before updates
- typical TTL: around a day
- mostly UDP (port 53), but can use TCP

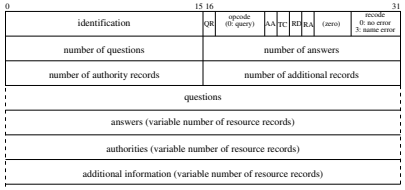
DNS query

single question:
5 lupus 5 fokus 3 gmd 3 de 0; 16-bit query type; 16-bit query class (1: Internet address)

query type:

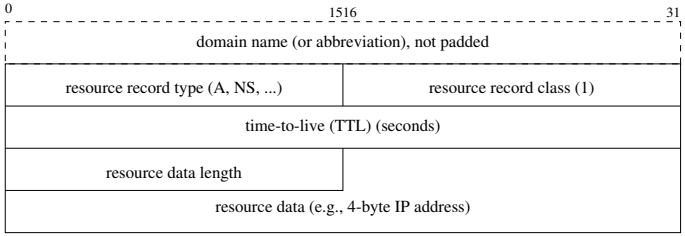
A	IP address
NS	authoritative name server
CNAME	canonical name
PTR	pointer record
HINFO	host info
MX	mail exchange record
AXFR	zone transfer

DNS message format



QR	query (0) or response (1)
opcode	0: query; 1: inverse query; 2: server status
AA	authoritative answer
TC	truncated (only first 512 bytes returned)
RD	recursion desired
RA	from server: recursion available
rcode	0: no error; 3: name does not exist

Resource records



- Domain name is *abbreviation* if 2 high-order count bits on: 16-bit pointer to location within DNS message
- may return extra records (e.g., MX also returns A, just in case)

Inverse mappings and pointer queries

- IP address → domain name(s)
- needed (e.g.) for diskless machines
- IP addresses are not assigned by geography or administration
- 192.35.149.52 ➠ 52.149.35.192.in-addr.arpa

RR examples

```
> set query=a
Name:     lupus.fokus.gmd.de
Address:  192.35.149.52
> set query=mx
> tu-berlin.de
tu-berlin.de preference=100, mail exchanger=mail.zrz.TU-Berlin.DE
tu-berlin.de preference=100, mail exchanger=mailgzrz.TU-Berlin.DE
tu-berlin.de preference=150, mail exchanger=sc.ZIB-Berlin.DE
> set query=hinfo
> lupus
lupus.fokus.gmd.de CPU=SS20 OS=Solaris
> set query=soa
> fokus.gmd.de
origin = gaia.fokus.gmd.de
mail addr = wasserroth.fokus.gmd.de
serial = 236
refresh = 10800 (3 hours)
retry   = 1800 (30 mins)
expire  = 3600000 (41 days 16 hours)
minimum ttl = 86400 (1 day)
```


nslookup

```
> set norecurse
> www.sun.com
Server:  gaia.fokus.gmd.de
Address:  192.35.149.140
Name:    www.sun.com
Served by:
- NS.SUN.com
          192.9.9.3
          SUN.COM
- VGR.ARL.MIL
          128.63.2.6, 128.63.16.6, 128.63.4.4, 26.2.0.29
          SUN.COM
```


DNS: Summary

- *not* a general directory service (can't find company name → domain name) ➠ whois, whois++, X.500, …
- but: currently no "real" directory service (except Yahoo, Lycos, …) ➠ need memorable domain names
- trademarks, overloading (single .COM domain for Apple Records and Apple Computers)
- flu.com, stupid.com, diaper.com, mafia.com, … ➠ 86,000 .COM domains
- ideas:
  - charge for domain names ($50/year) ➠ doesn't deter P&G
  - encourage geographic registration (.us domain)
  - new trade domains (.computer.com)