DNS (Domain Name Service)

- Distributed database of hostname-IP address mappings
- Application-level protocol
- Hierarchical: “zones” of responsibility
- BIND is most common implementation (Berkeley Internet Name Domain server)
- RFC 1034, 1035
- Typically UDP packets

DNS: client side

- **Resolver**: client-side program
  - contacts proper server
  - returns the result
- Library functions:
  - `gethostbyname()`
  - `gethostbyaddr()`

DNS namespace

![DNS namespace diagram](image)
DNS namespace (cont.)

- Example: www.mathcs.carleton.edu
  - edu: top-level domain (TLD)
  - carleton: network
  - mathcs: subnetwork
  - www: hostname
- Note that sometimes hostname != machine name
  - Q: how does DNS handle this case?

DNS architecture

- Servers divided into zones
- Each organization has at least one local DNS
  - mathcs: uses Carleton's DNS servers
- Requests go to local servers first
  - if local server cannot resolve the name, it goes up to the next level
- All of the TLDs have one or more DNS servers associated with them
- Responses can be cached at each level

DNS architecture (cont.)

- Root-level servers
  - currently there are 13 of them
    - 10 USA, 2 Europe, 1 Asia
    - letters A-M
  - queries for unknown TLDs or when TTLs expire
  - every DNS server must know how to contact these root name servers

Server types

- Authoritative: server has the “last word” on the existence of a name
  - root level servers
  - local servers can be authoritative over their own domains
- Iterative: if this server cannot answer the query, it will return a list of other servers who may be able to do so
- Recursive: if this server cannot answer the query, it will contact other servers and try to find the answer on its own
**DNS packet fields**

- **ID**: identification
- **Flags**:  
  - QR (1 bit): 0=query, 1=response
  - opcode (4 bits): typically 0
  - AA (1 bit): 1=“authoritative answer”
  - TC (1 bit): response was longer than 512 bits, truncated
  - RD (1 bit): recursion desired
  - RA (1 bit): recursion available
  - 3 bits of 0s
  - rcode (4 bits): return code; typically 0 (no error) or 3 (name does not exist)

**Question format**

- **Name**: variable size  
  - format: <label1><part1><label2><part2>...0  
  - <labeln>: 1 byte, specifies the number of bytes in <partn>  
  - <partn>: portion of the name or IP address  
  - Example: 3www6mathcs8carleton3edu0
  - **Type**: typically A (request is for IP address), CNAME (request is for “canonical name”), PTR (request is for name corresponding to IP address)
  - **Class**: almost always 1 (for Internet address)

**Response format (Resource Records)**

- **Domain name (variable)**: same as query name  
- **Type (2 bytes)**: see previous slide  
- **Class (2 bytes)**: ditto  
- **TTL (4 bytes)**: number of seconds this response can be cached  
- **Resource data length (2 bytes)**: the amount of data in the reply (RR)  
- **Resource data (variable)**: the response
UNIX commands, files

- `/etc/resolv.conf`: lists nameservers and domains
  - domain: default domain
- `host`: returns IP address for hostname, and vice versa
- `dig`: returns more detailed information from DNS query

Caching responses

- A good name server will cache both queries and responses
- Applications may also cache responses
- TTL field indicates for how long response can be cached

Implementation note

- A name server is typically not just one server
  - particularly at the root server level
- Instead, queries are load-balanced among a set of servers
  - `redirector`: distributes queries to the servers according to some policy (RR, weighted RR, etc.)
- Example: F root server = 4 machines in San Francisco and Palo Alto

More DNS implementation goodies

- Study in October 2002 at F root server (24 hours):
  - 2% of requests were legitimate
  - 70% were repeated requests
  - 7% were “A for A” requests (IP address of IP address)
  - 13% were for unknown TLDs
  - 220 sources generated 50% of queries
  - average 1768 queries/sec
- Moral: DNS is poorly implemented in many places
  - packet filtering is a big culprit

Summary

- DNS is a large-scale distributed system for name-to-IP address mapping
- 13 root servers plus many localized servers (at least one for each TLDs)
- Servers follow the hierarchical naming structure of Internet hosts
- Utilizes UDP (most often); packet sizes are \( \leq 512 \) bytes (typically)
- Proper configuration of servers is key!