Need

- Network addresses are numbers
- Addresses are topologically oriented
  - Used for routing purposes
  - Moving a host may require change of address
- Are not easy to remember
- Names can be used for users and for applications
  - Easy for humans
  - Can be used as a low level service discovery mechanism
  - Changing the server machine requires just changing the name->IP binding
- Names can have a logical structure

Some history

- In the beginning, there was the hosts file
  - A file containing the names and addresses of all hosts in the network
  - Problematic: maintainability, size
  - Still used as a backup (local network host information)
- DNS/116 Name service
  - Non scalable, topology-oriented
- DNS
  - Tree-structured
  - Delegation
  - Separated from network structure and topology
  - Uses UDP, port number 53 for queries, TCP for zone transfers
**DNS system**

- **Terms:**
  - **name server** = set of possible names, flat or hierarchical
  - naming system maintains a collection of bindings of names to values
  - given a name, a resolution mechanism returns the corresponding value
  - a name server is an implementation of the resolution mechanism
  - DNS (Domain Name System) = name service in Internet
  - Zone is an administrative unit, domain is a subtree

**DNS domain hierarchy**

- **First level hierarchy**
  - Top level domains (common domain names + country codes)
  - DNS first level managed by Internet Corporation for Assigned Names and Numbers (ICANN), also manages address allocations

- **Hierarchy is partitioned into subdomains, zones**
  - Zone corresponds to administrative boundaries in DNS and often also DNS servers.

---

**Name structure**

1. Root
   - Hosts, e.g., org, inc, ltd
2. Top level domains (common domain names + country codes)
3. Organisation type domain name
   - in some countries, e.g., uk
4. Organisation
   - Registration (but, ltd)
   - Registered trademark
5. Organisation subdomain
6. Host name

**Fully Qualified Domain Name (FQDN)**

- Host subdomain (org type, ltd)
- Hostname + domain name = .
- Read from right to left
- A host can be addressed by
  - FQDN
  - Hostname + partial domain names
- E.g.
  - www.netlab.twc55 (FQDN)
  - www.netlab.twc55/host + partial domain name (subject to supplement)
  - www.netlab
**Getting a domain name**

- **TLD**: ICANN delegated name registrars
- **Country level**: local administrations.
- **Finland**:
  - Ficora (Viestintävirasto)
  - Companies, registered associations
  - For public institutions, their name or administration
  - Must not violate registered trademarks.

---

**Elements**

- **RESOLVER**
  - A library within the operating system, provides an API and handles queries
  - Contains a cache
- **PRIMARY NAME SERVER**
  - One per domain. Contains the binding information for all hosts
- **SECONDARY NAME SERVER**
  - Duplicates the information of primary servers, used for distributing load and for redundancy.
- **CACHE NAME SERVER**
  - Contains cache, but no binding info. Queries other DNS servers
- **PROXY NAME SERVER**
  - An cache NS but without cache (For load balancing etc.)

---

**Bind (1)**

- Zones defined in two or more name servers (Redundancy)
  - Clients send queries to name servers
  - Servers respond with final answer or pointer to another server
- Name binding database consists of resource records
  - format: (Name, Value, Type, Class, TTL)
  - Type: how Value is interpreted.
    - A means that Value is an IP address, name-address mapping
    - AAAA: All IPv6 address
    - NS: Value contains name to host that knows how to resolve the name
    - CNAME: Value is a canonical name for host, used to define aliases
    - DKIM: Subdomain redirecting
    - mail-to information
    - MX: Value gives the domain name for a host running a mail server
    - PTR: Pointer to domain name Reverse DNS

---

**Bind (2)**

- **RP**: Responsible person
- **LOC**: Owner of the host
- **TXT**: Detailed
- **SIG, TSIG, DKIM CERT**: security attributes
- **Class**: only widely used class IN (Internet)
- **TTL**: Time how resource record is valid (used by servers that cache resource results from other servers)
- can use service-specific aliases (www, smtp, mdm, print, etc.)
- MX allows administrators to redirect all mail of a host to a specified mail server
Example

```
Resolution: 130.233.154.176
Resolution: keskus
WWW: keskus
SMTP: keskus
ctkit.hut.fi IN SOA keskus.ctkit.hut.fi. (100000800; serial number 10800; Refresh 3 hours 3600; Retry 1 hour 604800; Expires 1 week 86400) ; TTL 1 day
IN NS keskus.ctkit.hut.fi.; primary name server
IN NS ns1.hut.fi.; first secondary
IN NS ns2.hut.fi.; second secondary
IN MX 10 keskus; primary mail server
IN MX 20 smtp-1.hut.fi.; backup
IN MX 20 smtp-2.hut.fi.; second backup
keskus IN A 130.233.154.176
```

DNS domain hierarchy (cont)

- Root name server: NS record for each 2nd level server + A record that translates name into IP address
  - example.edu, alt-princeton.edu, NS, BNS-1, BNS-2, BNS-3, BNS-4
  - example.edu, alt-princeton.edu, A, BNS-1, BNS-2, BNS-3, BNS-4

- At 2nd level, records contain either final answers or pointer to 3rd level name servers
  - example.edu, alt-princeton.edu, NS, BNS-1, BNS-2, BNS-3, BNS-4
  - example.edu, alt-princeton.edu, A, BNS-1, BNS-2, BNS-3, BNS-4

- Lowest level contains final records, aliases for hosts (CNAME) and MX records

Name resolution

- How did the client locate the root server in the first place?
  - name-to-address mapping for one or more name servers is well known (published outside the naming system itself)
  - in practice, resolver initialized with the address of a local name server
    - client makes a query to local server -> local server makes queries further
  - advantages
    - only the servers need to know about root name servers
    - local server gets to see the responses (can cache them)
    - on a busy naming DNS (in Unix), try "dig" "nslookup", or "host <hostname>"

- Note: Internet has identifiers at several levels - domain names, IP addresses, and physical network addresses
  - users give domain names in applications - applications use DNS to translate these into IP addresses - IP does forwarding at each router, so it maps IP addresses into another (next hop route) - IP engages AIP to translate the next hop IP address into a physical address
Reverse DNS
- Finding the name when knowing the address
- A different hierarchy: in-high sites
- E.g., What is the hostname of 130.233.154.148?
  - Query 148.154.233.100 to valid site
- A separate hierarchy, organized as the address space
- Used for “security purposes”
  - A server might ask if the client name and address match

DNS as a Service
- Requires high reliability
- No single failure should affect → servers located in different parts of the network
  - E.g.: Hydra, Heliski, fi.
    - hi-service.net
    - griffluence.fi
    - netfin-global.anchors.fi
    - tina.actor.fi
    - hi-us.net
- Difficult to organize → Secondary DNS is an easy and important service to provide

Future
- Security still weak
- Using DNS as a directory structure (?)
  - Service Location
  - Generalization of MX records
- Mapping Telephone numbers to IP addresses?
  - Problems of policy (secret numbers, value)
- Character set