

Network Protocols

Domain Name System (DNS)

largely based on slides from D. Comer

Names

- Internet communication requires an IP address
- Humans prefer to use easy to remember names
- Need an automated system to translate IP to name
- This is the Domain Name System
- A static database (e.g. /etc/hosts) can also be used
 - Scaling issues with these static databases

Basic DNS functionality

- Given a name of a computer/host
- Return the IP address associated with the name
- Method used:
 - Distributed lookup of names in a hierarchy
 - Client contacts server(s) as necessary

Domain name syntax

- Alphanumeric segments separated by dots
 - www.depaul.edu
 - www.cs.depaul.edu
- Top level hierarchy starts from the right
 - .edu – top level domain (TLD)

Obtaining a domain name

- An organization:
 - chooses a name
 - it must be unique and available
 - register name with a central authority
 - placed under a top level domain
- Names subject to international law
 - trademark
 - copyright

Top Level Domains

<i>Domain Name</i>	<i>Assigned To</i>
<i>aero</i>	<i>Air transport industry</i>
<i>arpa</i>	<i>Infrastructure domain</i>
<i>biz</i>	<i>Businesses</i>
<i>com</i>	<i>Commercial organization</i>
<i>coop</i>	<i>Cooperative associations</i>
<i>edu</i>	<i>Educational institution</i>
<i>gov</i>	<i>United States Government</i>
<i>info</i>	<i>Information</i>
<i>int</i>	<i>International treaty organizations</i>
<i>mil</i>	<i>United States military</i>
<i>museum</i>	<i>Museums</i>
<i>name</i>	<i>Individuals</i>
<i>net</i>	<i>Major network support center</i>
<i>org</i>	<i>Non-commercial organizations</i>
<i>pro</i>	<i>Credentialed professionals</i>
<i>country code</i>	<i>A country</i>

Hierarchy with a organization

- Subdivision possible
- Arbitrary levels allowed
- Not standardized
- Controlled locally by the organization

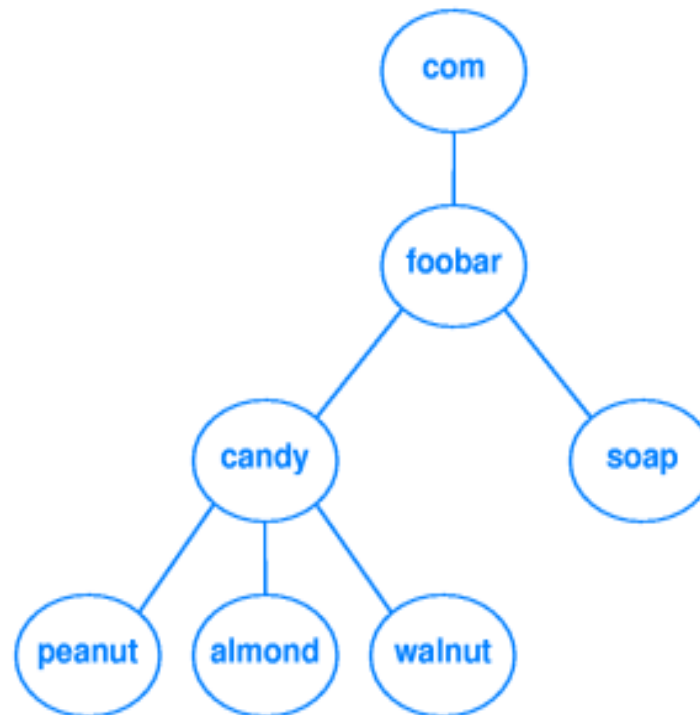
Example name structure

- First level is .com
- Second level is company name
- Third level is a division within a company
- Fourth level is either
 - company subdivision
 - host

A domain name example

- Company foobar with candy and soap divisions
- Candy has subdivisions, soap does not
- Names in soap division take the form:
 - host.soap.foobar.com
- Names in candy division take the form:
 - host.subdivision.candy.foobar.com

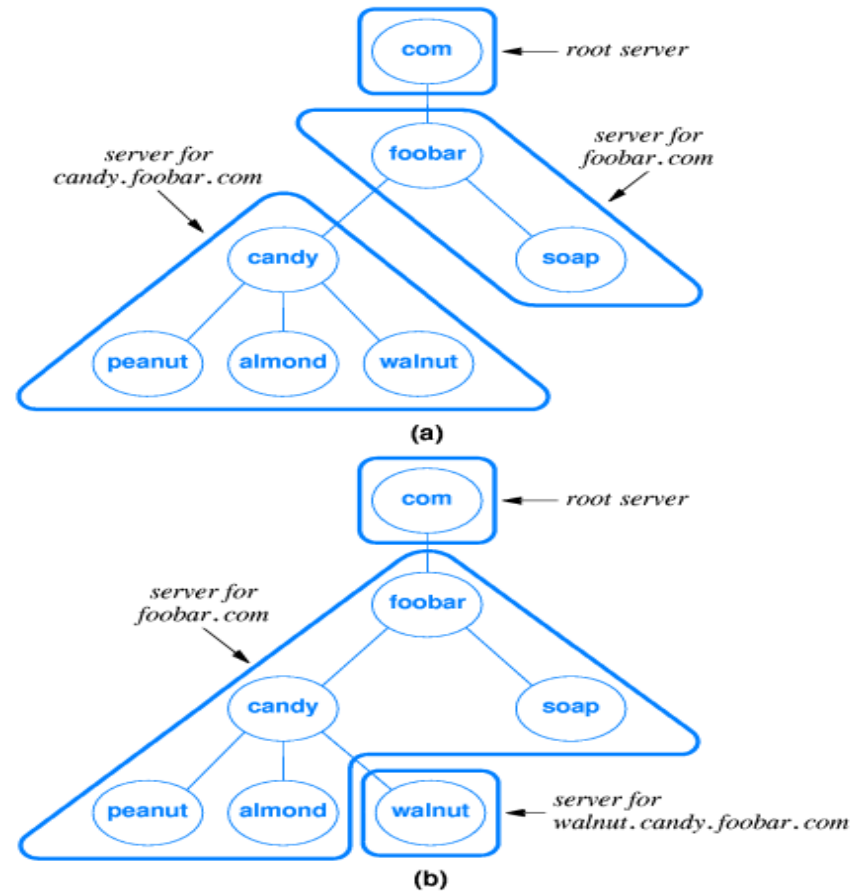
DNS example illustrated



DNS client-server interaction

- Client is known as a resolver
- Multiple DNS servers are typically used
- Arranged in a hierarchy
- Each server corresponds to its part in the hierarchy

Two possible DNS hierarchies



Inter-server links

- Servers know reach parent domain server
 - Each server knows how to reach the root (.)
- Servers know child domain servers

DNS deployment in practice

- Redundant servers are used
- Root servers A-M are:
 - geographically dispersed
 - often each root is redundant, dispersed
 - generally run by very clueful people
- ISPs/organizations offer DNS services to its users
- Small organizations can get DNS from an upstream

DNS lookup

- Application triggers request to local name server
- If local server knows the answers, it returns it
- Otherwise, the local server go finds it for the client
 - Starts at top level domain and follows links
- Iterative lookups force client to follow links
- Recursive lookups result in server following links

DNS caching

- Servers cache answers for some period of time
 - Usually controlled by the TTL in an answer
- Hosts can also cache previous answers
- Caching
 - improves efficiency
 - eliminates unnecessary searching
 - works well because of high locality reference
- Cache poisoning attacks can be a problem

DNS types

- Each entry in a server consists of
 - domain name
 - DNS type for the name
 - value to which the name corresponds
- Client asks for the name and specifies the type
- Server matches name and type and returns answer

Example DNS types

- Type A (address)
 - value is an IP address assigned to a host
- Type MX (mail exchanger)
 - value is an IP address that handles mail for host
- Type CNAME (canonical name aka alias)
 - value is another domain name
- Type PTR (pointer)
 - value is a name, this is used for reverse lookups

Domain name abbreviations

- DNS uses fully qualified domain names (FQDN)
- Users/apps sometimes do not specify the full name
- Configure resolver with a list of common suffixes
 - e.g. depaul.edu, [cs.depaul.edu](#)
- User enters www, resolver tries until match:
 - www
 - [www.depaul.edu](#)
 - [www.cs.depaul.edu](#)