# DNS - Domain Name System

2023-Oct-17

# The Domain Name System

- translates hostnames to IP addresses
- and vice versa
- computers like numbers, humans like names
- have you ever texted a cell phone number?
- DNS is like a global contact list of names → numbers

10010111.01100101.00101110.10000101

[ 32 bit IP address ]

[ 151.101.46.133 ]

# DNS Query

**A record** ("forward" lookup):

```
$ host www.unix.com
www.unix.com has address 209.126.104.117
```

**PTR record** ("reverse" lookup)

```
$ host 209.126.104.117
117.104.126.209.in-addr.arpa domain name pointer www.unix.com.
```

# DNS Query

Try entering 209.126.104.117 in a web browser.

www.unix.com is easier to remember

# Brief History

- 1960's : DARPA funds ARPAnet (research network)
- 1970's : few hundred hosts on the ARPAnet
- Single HOSTS.TXT file manager by Stanford Research Institute (SRI)
- The file HOSTS.TXT maps hostnames to addresses
  - Unix /etc/hosts is a descendent of HOSTS.TXT
- sysadmins would email changes/additions to the SRI
- users periodically downloaded the latest HOSTS.TXT file
- resulted in load on SRI's FTP servers, network traffic

# History

- 1980's : TCP/IP included in the free BSD Unix o/s, makes connecting to the ARPAnet much easier
- Went from hundreds to tens of thousands of hosts on the ARPAnet
- 1988 : DARPA ends experiment, DoD begins dismantling ARPAnet
- NSFNET replaces ARPAnet backbone
- 1995 : Publicly funded NSFNET transitions to commercial backbones run by MCI, Sprint, PSINet, UUNET,...
- ARPAnet → Internet

# /etc/hosts

```
$ ping athena (not found)

$ sudo gedit /etc/hosts
10.1.1.1    athena

$ ping athena
```

One line in /etc/hosts for every host on the internet?

# Limits of single text file

- TCP/IP resulted in an explosion of new hosts
- Network and server load from continuous downloading of HOSTS.TXT from SRI
- name collisions – all hostnames had to be unique, mail outages caused when overlaps occurred
- consistency – users could not access new hosts until their admins installed the latest HOSTS.TXT file

# Solving the Scalability Problem

- Paul Mockapetris (USC) tasked with architecting a new scalable system.
- RFC-882 and RFC-883
- evolved into  RFC-1034 and RFC-1035

*** Read the RFC's (Request for Comments) ***

Implementors use RFC's

Code is only as good as the specs

# Goals/Architecture

- Distributed database – decentralize
- Delegation – allow local control
- Hierarchical namespace would ensure unique names (avoid collisions)
- Robustness and performance achieved through caching and replication
- Client/Server model - Name Servers/Resolvers

# Client/Server Model

- Name servers make the database available for query
- Resolvers query the name servers
- Use “host” on Linux to perform manual queries
- Use “nslookup” on Windows
- Software apps use DNS transparently

# Structure of the DNS database

host1.library.yale.edu.

```
                       root
                        .
                        |
               ------------
                |
               edu
                |
          ---------
           |
      yale |
           |
      -------
       |
      library
       |
 -------------
  |         |
host1     host2
```

host1.library.yale.edu.

```
                       root <-- managed by ICANN
                        .
                        |
               ------------
                |
               edu <-- managed by NSI
                |
          ---------
           |
      yale <-- managed by Yale ITS
           |
      -------
       |
      library <-- managed by LIT
       |
 -------------
  |         |
host1     host2
```

# Structure continued

- Inverted tree, root node at top
- Root node is written as a single "."
- Top-level domains can be sub-divided into subdomains
- The root servers delegate authority for the top-level domains
  - .com, .edu,...
- They in turn delegate to hosts w/in their scope (yale.edu)
- Network admins at yale.edu can delegate subdomains as they see fit
- ...

# Structure continued

- Every domain is unique
- Each domain/subdomain can be administered by a different organizations
- An organization can break out administration of its subdomains as needed (delegation)

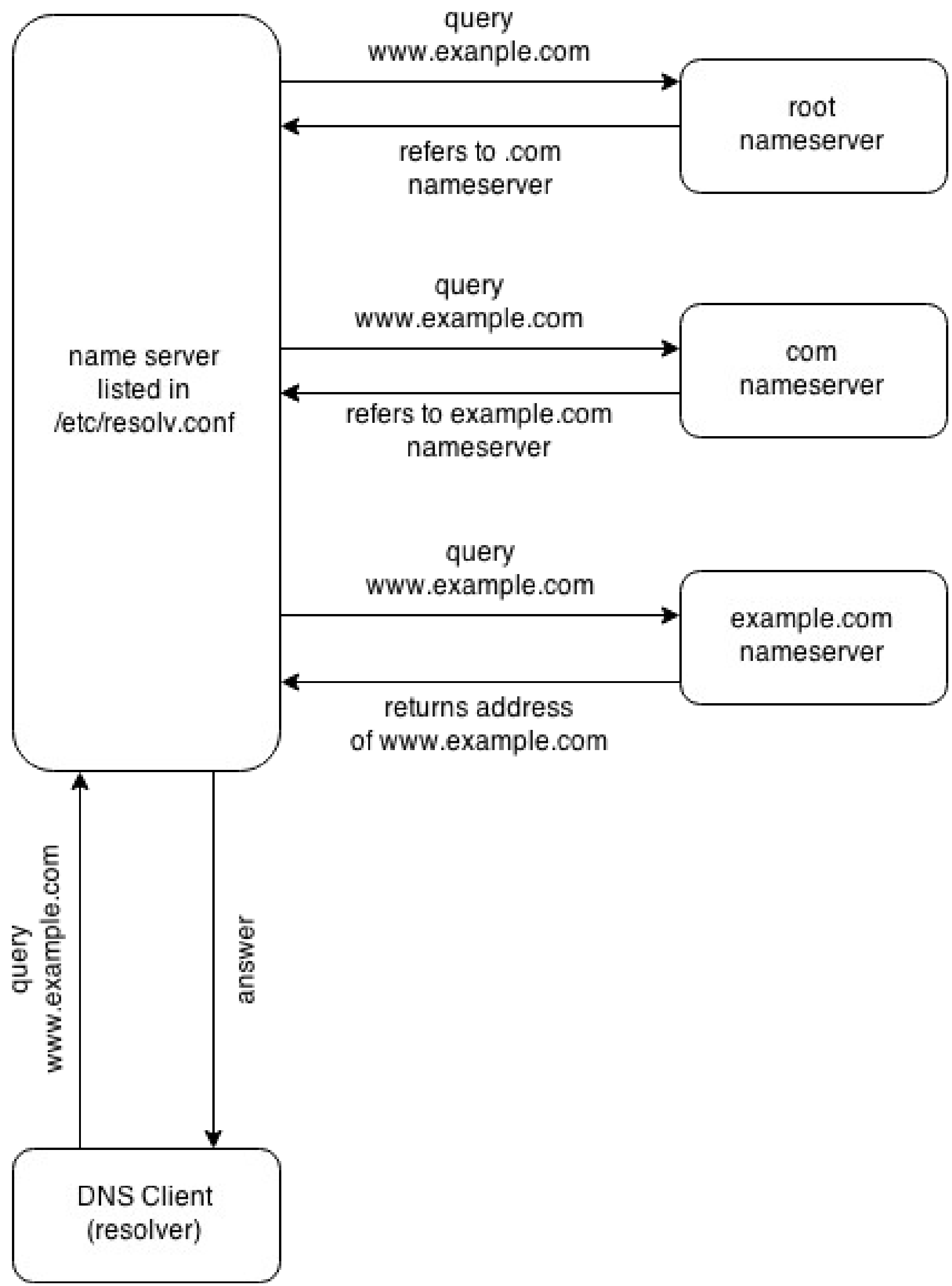
# Structure continued

- Each host has a domain name
- A hostname resolves to one or more IP
- A hostname can be canonical or it may be an alias (CNAME vs A record)
- Making domain names hierarchichal solves collision issues

frodo.harvard.edu vs frodo.yale.edu

# Anatomy of a Query

- What happens when you enter "www.yale.edu" in your web browser?


name server
listed in
/etc/resolv.conf
query
www.exanple.com
root
nameserver
refers to .com
nameserver
query
www.example.com
com
nameserver
refers to example.com
nameserver
query
www.example.com
example.com
nameserver
returns address
of www.example.com
query
www.example.com
answer
DNS Client
(resolver)

# Resources

- http://docstore.mik.ua/orelly/
- https://www.isc.org/downloads/bind/
- The RFC's
  - https://www.ietf.org/rfc/rfc1034.txt
  - https://www.ietf.org/rfc/rfc1035.txt