CS 330: Network Applications & Protocols

Application Layer: FTP, SMTP, DNS

Galin Zhelezov
Department of Physical Sciences
York College of Pennsylvania

Based on Computer Networking: A Top-Down Approach, 7th Edition by Kurose and Ross

Overview of Application Layer

- **Network Application Architectures**
- **HyperText Transfer Protocol (HTTP)**
- **File Transfer and Email protocols (FTP, SMTP)**
- **Domain Name System (DNS)**
  - Function
  - Distributed Structure
  - DNS Caching
  - DNS Records
  - DNS Vulnerabilities
- **Peer-to-Peer Applications (P2P)**

DNS: Domain Name System

- **DNS servers translate a host name to IP address**
  - e.g. www.ycp.edu ➔ 54.210.214.116
  - Would be painful to browse Internet and remember IP addresses
- **Hosts and name servers communicate to resolve names**
  - address ➔ name translation
- **Distributed database of all hosts in the universe**
  - Avoids single point of failure
  - Distributes name resolution traffic
  - Geographically distributed
  - Easier to maintain
- **Often used by other application-layer protocols (e.g. SMTP, HTTP, FTP) to translate hostnames to IP addresses**

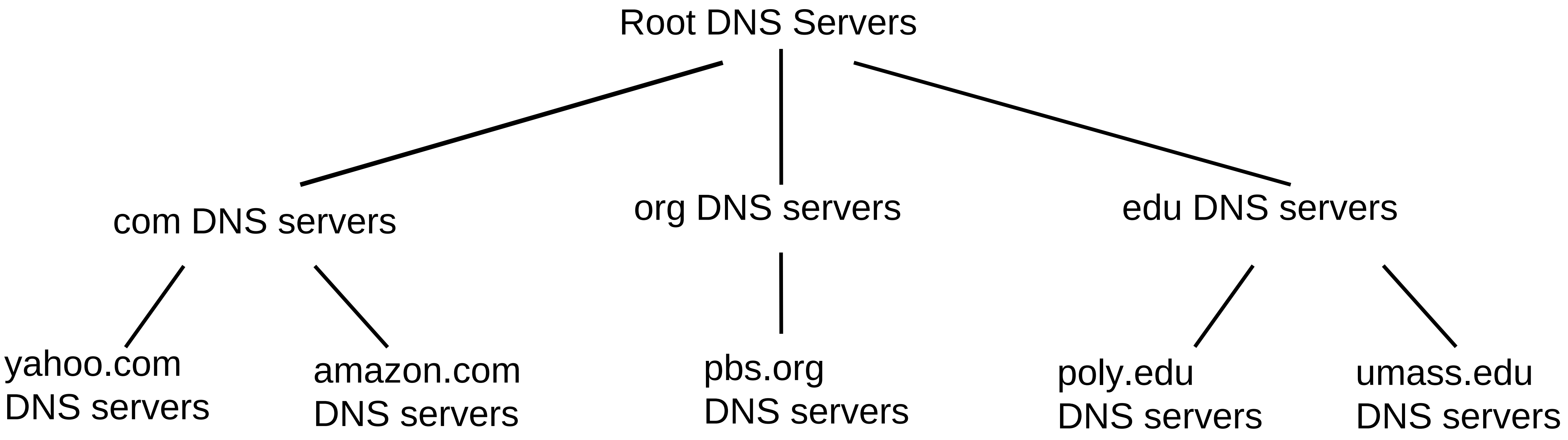
Other Services Provided By DNS

- **Host aliasing**
  - Provides canonical name when alias name is provided
  - www.gmail.com ➔ googlemail.l.google.com
- **Mail server aliasing**
- **Load distribution**
  - Why not centralize DNS?
  - Replicated web servers, many IP addresses correspond to one name

DNS Example

- `Nslookup -type=a yahoo.com`
- `Nslookup -type=ns yahoo.com`
- `Nslookup -query=mx yahoo.com`
- `Nslookup -type=any yahoo.com`

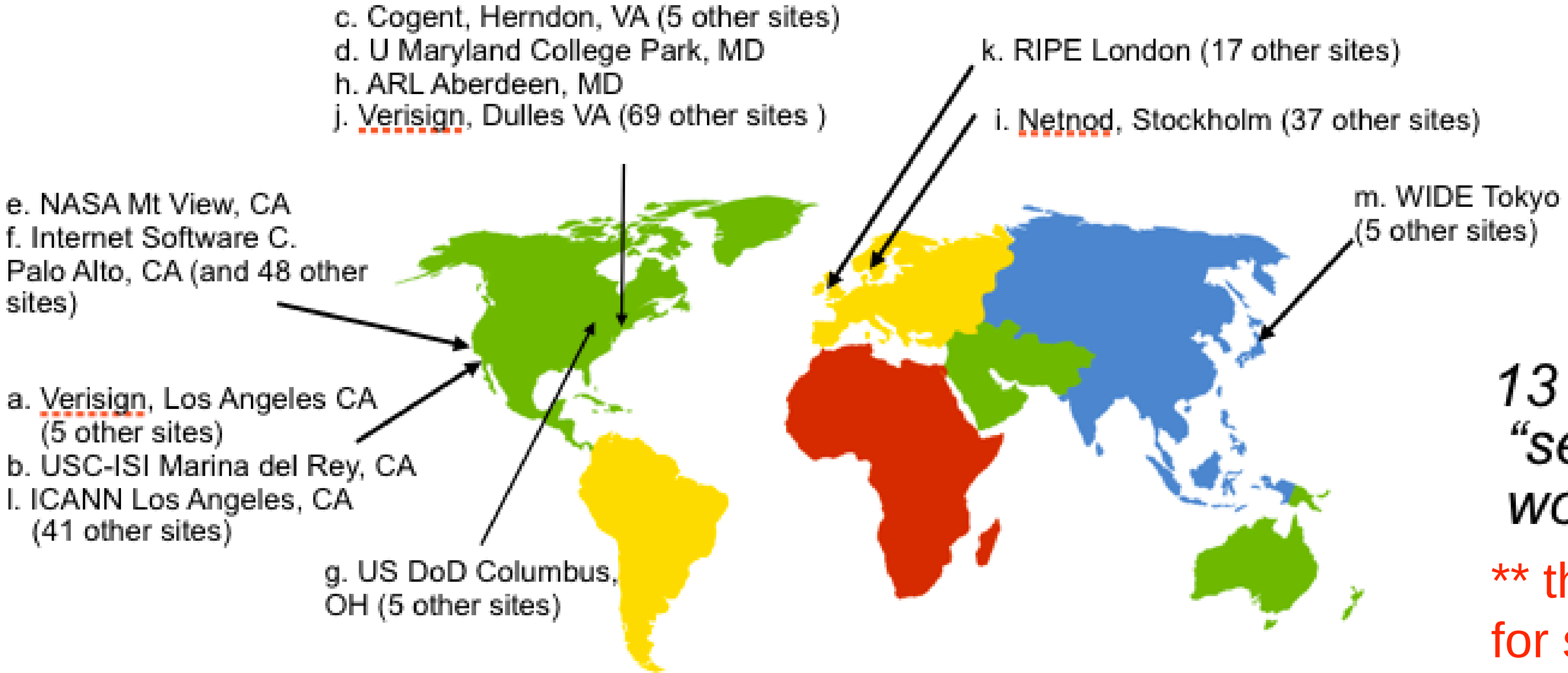
DNS: A Distributed, Hierarchical Database



- **Client wants IP for www.amazon.com**
  - Client queries root server to find com DNS server
  - Client queries .com DNS server to get amazon.com DNS server
  - Client queries amazon.com DNS server to get IP address for www.amazon.com

Root DNS Servers

- **Contacted by local name server that can not resolve name**
- **Root name server:**
  - Contacts authoritative name server if name mapping not known
  - Gets mapping
  - Returns mapping to local name server


c. Cogent, Herndon, VA (5 other sites)
d. U Maryland College Park, MD
h. ARL Aberdeen, MD
j. Verisign, Dulles VA (69 other sites)
k. RIPE London (17 other sites)
i. Netnod, Stockholm (37 other sites)
m. WIDE Tokyo (5 other sites)
e. NASA Mt View, CA
f. Internet Software C. Palo Alto, CA (and 48 other sites)
a. Verisign, Los Angeles CA (5 other sites)
b. USC-ISI Marina del Rey, CA
l. ICANN Los Angeles, CA (41 other sites)
g. US DoD Columbus, OH (5 other sites)

13 root name “servers” worldwide

** though they are replicated for security and reliability

Top-Level Domain & Authoritative DNS servers

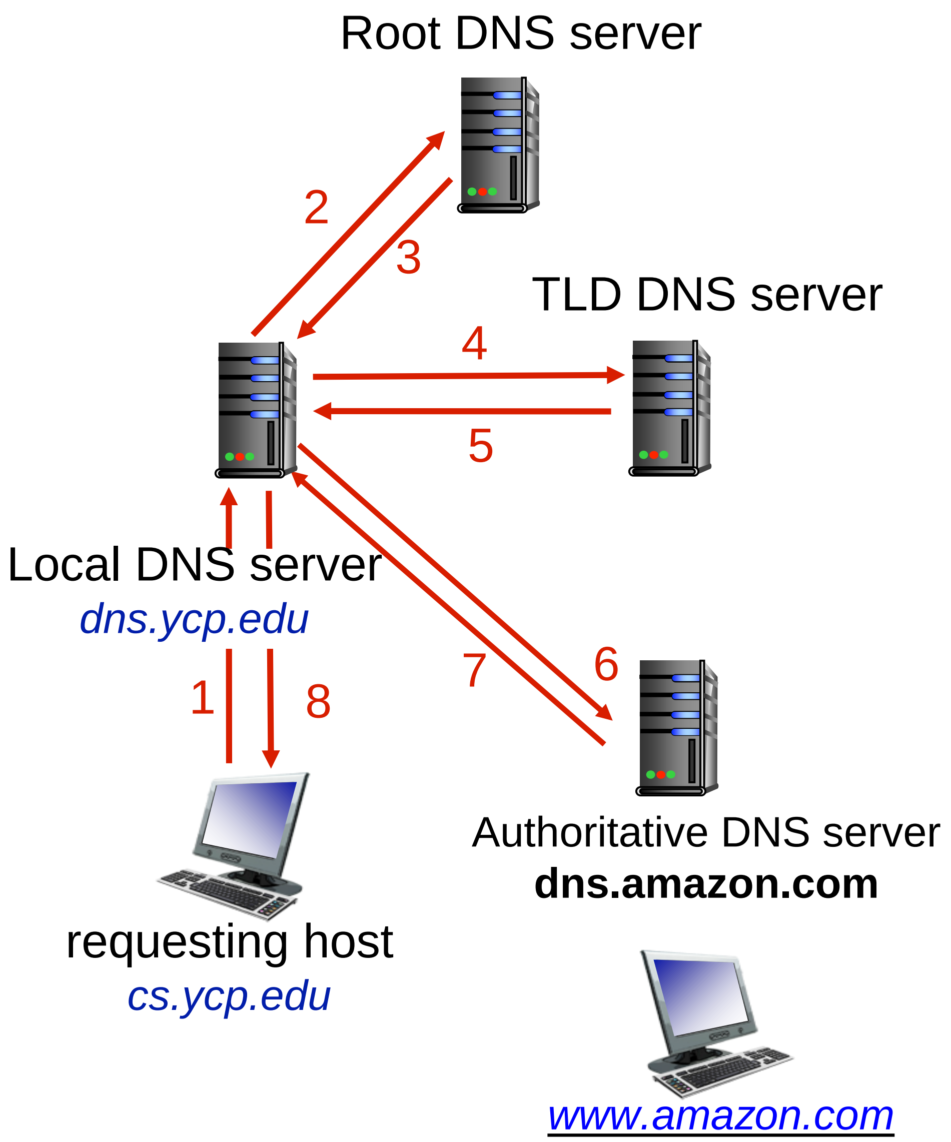
- **Top-Level Domain (TLD) Servers**
  - Responsible for com, org, net, edu, aero, jobs, museums, and all top-level country domains (e.g. uk, fr, ca, jp)
  - Verisign maintains servers for .com TLD (and many others)
  - Educause maintains servers for .edu TLD
- **Authoritative DNS Servers**
  - Organization's own DNS server(s), providing authoritative hostname to IP mappings for organization's named hosts
  - Can be maintained by organization or service provider

Local DNS Name Server

- **Does not strictly belong to the DNS hierarchy**
- **Each ISP (residential ISP, company, university) has one**
  - Also called “default name server”
- **When host makes DNS query, query is sent to its local DNS server**
  - Has local cache of recent name-to-address translation pairs (but may be out of date!)
  - Acts as proxy, forwards query into hierarchy

DNS Name Resolution Example

- **Host at cs.ycp.edu wants IP address for www.amazon.com**
- **Iterated query:**
  - Contacted server replies with name/address of server to contact
  - "I don't know this name, but ask this server"



DNS Caching / Updating Records

- **Once (any) name server learns mapping, it caches mapping**
  - Cache entries timeout (disappear) after some time (TTL)
  - TLD servers typically cached in local name servers
    - Thus root name servers not often visited
- **Cached entries may be out-of-date (best effort name-to-address translation!)**
  - If a named host changes its IP address, may not be known Internet-wide until all TTLs expire

DNS Records

- **Resource Record (RR) format stored by DNS servers**
  - `RR: (name, value, type, ttl)`
- **Four different RR types**

Type=A

`name` is hostname

`value` is IP address

Type=NS

`name` is domain (e.g. foo.com)

`value` is hostname of authoritative name server for this domain

Type=CNAME

`name` is alias name for some "canonical" (the real) name

`www.ibm.com` is really `servereast.backup2.ibm.com`

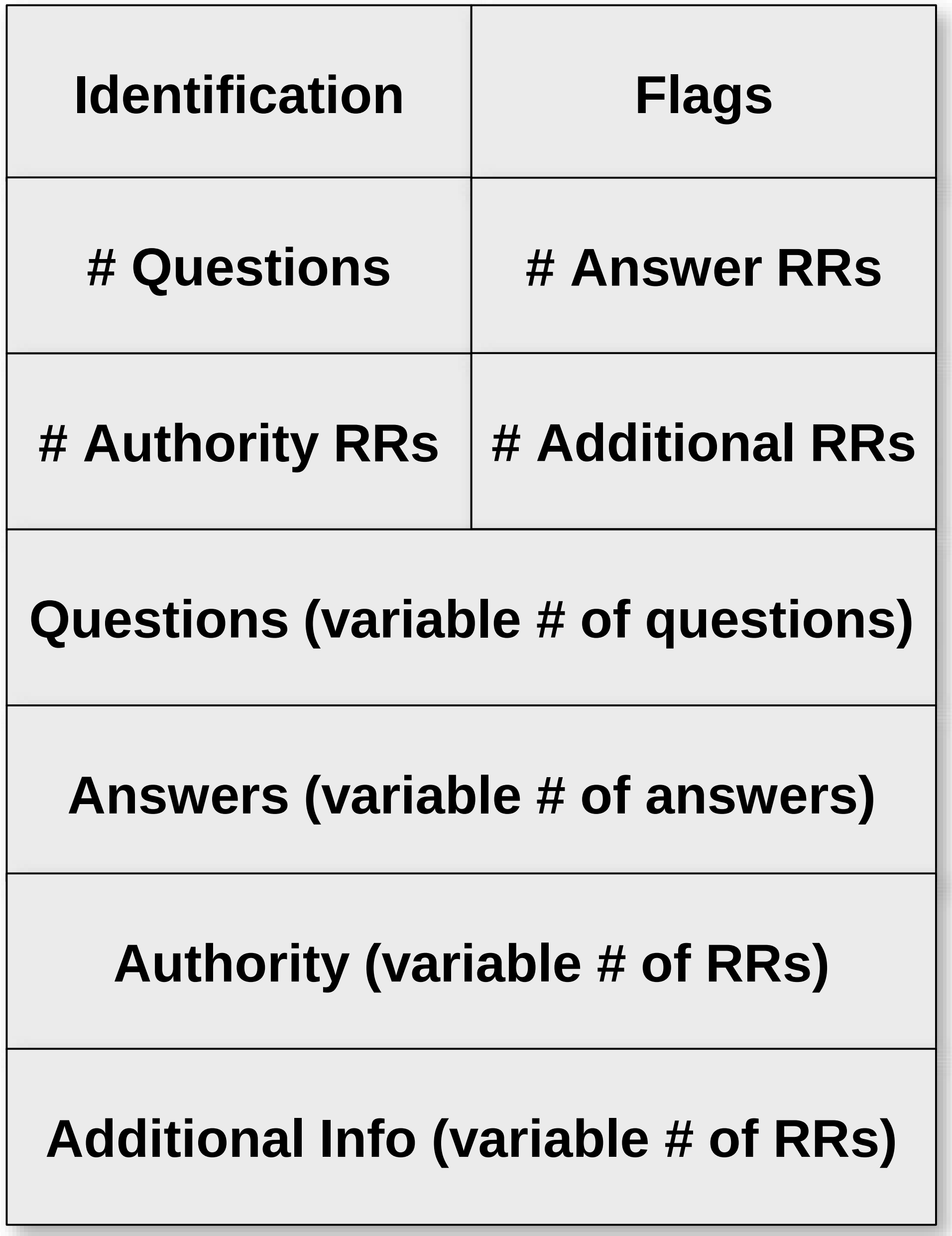
`value` is canonical name

Type=MX

`value` is name of mail server associated with `name`

DNS Message Format

- **Query and Reply messages, both use same message format**
  - Message Header
    - **Identification**: 16 bit # for query, reply includes same #
    - **Flags**:
      - Query or reply
      - Recursion desired
      - Recursion available
      - Reply is authoritative
    - **Question section**: contains name and type fields for the query
    - **Answer section**: contains RRs in response to a query
    - **Authority section**: contains RR for authoritative servers



Identification	**Flags**
# Questions	**# Answer RRs**
# Authority RRs	**# Additional RRs**
Questions (variable # of questions)	
Answers (variable # of answers)	
Authority (variable # of RRs)	
Additional Info (variable # of RRs)	

Inserting records into DNS

- **example: new startup "Network Utopia"**
- **register name networkuptopia.com at *DNS registrar* (e.g., Network Solutions)**
  - provide names, IP addresses of authoritative name server (primary and secondary)
  - registrar inserts two RRs into .com TLD server:

    ```
    (networkutopia.com, dns1.networkutopia.com, NS)
     (dns1.networkutopia.com, 212.212.212.1, A)
    ```
- **create authoritative server type A record for www.networkuptopia.com; type MX record for networkutopia.com**

DNS Vulnerabilities

- **Distributed Denial of service attacks on name server**
  - Bombard root servers with traffic
    - Not successful to date
    - Root servers are protected by traffic filters
    - Local DNS servers cache IP addresses of TLD servers, allowing root server bypass
  - Bombard TLD servers
    - Potentially more dangerous
- **Redirect attacks**
  - Man-in-middle
    - Intercept queries and return bogus replies
  - DNS poisoning
    - Send bogus replies to DNS server which then caches that info