CSCE 463/612
Networks and Distributed Processing
Fall 2023

Application Layer IV

Dmitri Loguinov
Texas A&M University

February 14, 2024

Chapter 2: Roadmap

2.1 Principles of network applications

2.2 Web and HTTP

2.3 FTP

2.4 Electronic Mail

- SMTP, POP3, IMAP

2.5 DNS

2.6 P2P file sharing

2.7 Socket programming with TCP

2.8 Socket programming with UDP

2.9 Building a Web server

DNS: Domain Name System

- People: many identifiers
  - Name, SSN, passport #
- Internet hosts, routers:
  - IP address (32 bit) used for routing datagrams
  - Names (e.g., yahoo.com) used by humans

Q: how to map between IP addresses and names?

  - Original technique: global file hosts.txt with all known hosts; flat namespace

Domain Name System:

- Distributed database
  - Implemented in hierarchy of many *name servers*
- Application-layer protocol
  - Hosts communicate with name servers to *resolve* names/IPs
  - UDP port 53
  - Single-packet query
  - Single-packet response

DNS

Services

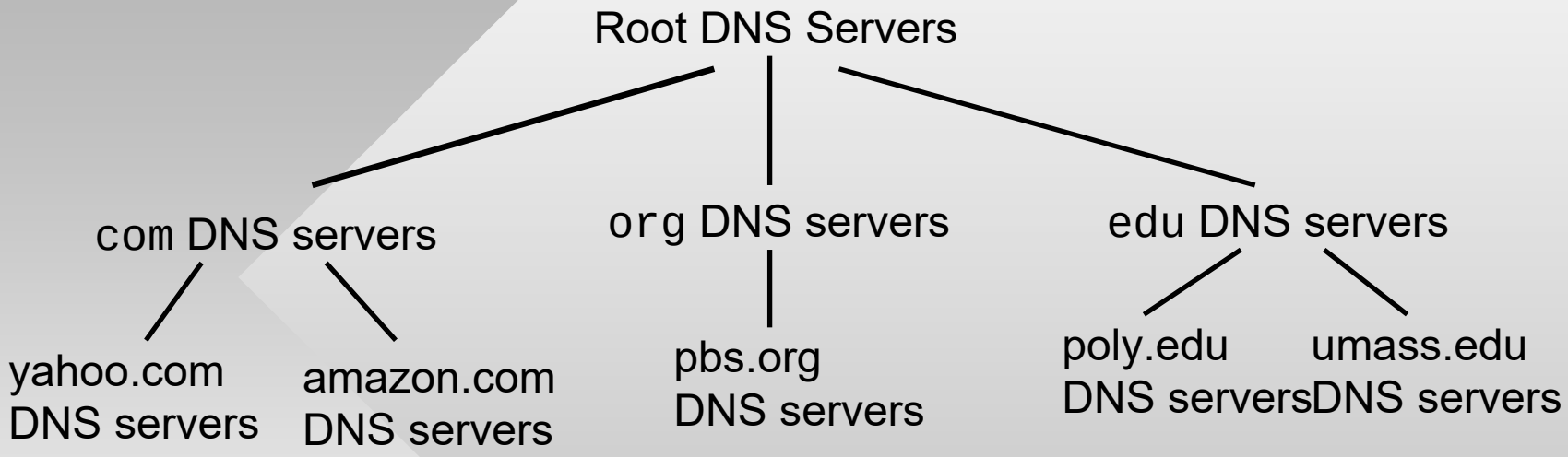
- Forward lookup
  - Hostname to IP translation
- Reverse lookup
  - IP to hostname
- Host aliasing
- Mail server lookup
- Load distribution
  - E.g., replicated web servers: set of IP addresses for one DNS name

Why not centralize DNS?

- Single point of failure
- Traffic volume (bandwidth, request rate)
- Lack of geographic proximity to user, hence high latency
- Inflexible (can't run code customized for domain)

Doesn't *scale!*

Distributed, Hierarchical Database



- Client wants IP for www.amazon.com:
  - Queries a root server to find the `com` DNS server
  - Queries the `com` server to get the `amazon.com` server
  - Queries the `amazon.com` DNS server to get IP address for `www.amazon.com`
- Who to ask about the location of root servers?
  - Nobody, their IPs are hardwired into OS

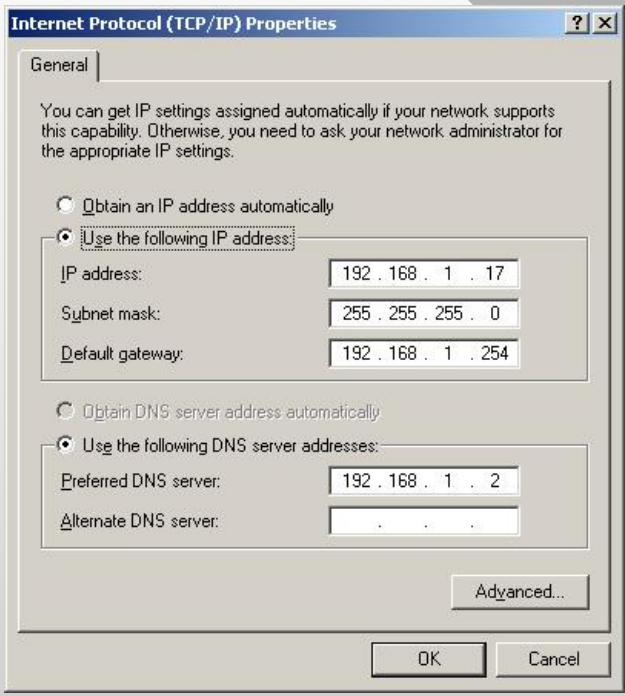
Types of DNS Servers

- There are 13 named root servers (called A-M), each with a fixed IP address
  - Some servers (e.g., A, C, F, I) are geographically distributed across multiple sites (1369 total in 2021)
  - More info: http://root-servers.org/

Types of DNS Servers

- Top-level domain (TLD) servers: responsible for generic TLDs (e.g., .com, .org, .net, .edu) and all country-code TLDs (e.g., .uk, .fr, .ca, .jp)
  - Around 1530 total gTLDs and cc-TLDs (2019)
  - Verisign runs .com , Educause .edu
- Authoritative servers: provide authoritative mappings for company servers (e.g., Web and mail)
  - Can be maintained by organization (e.g., amazon.com) or service provider (e.g., ISP or hosting company)
- Local name server: does not belong to the hierarchy
  - Any computer that accepts requests and then finds out the answer by traversing the DNS tree

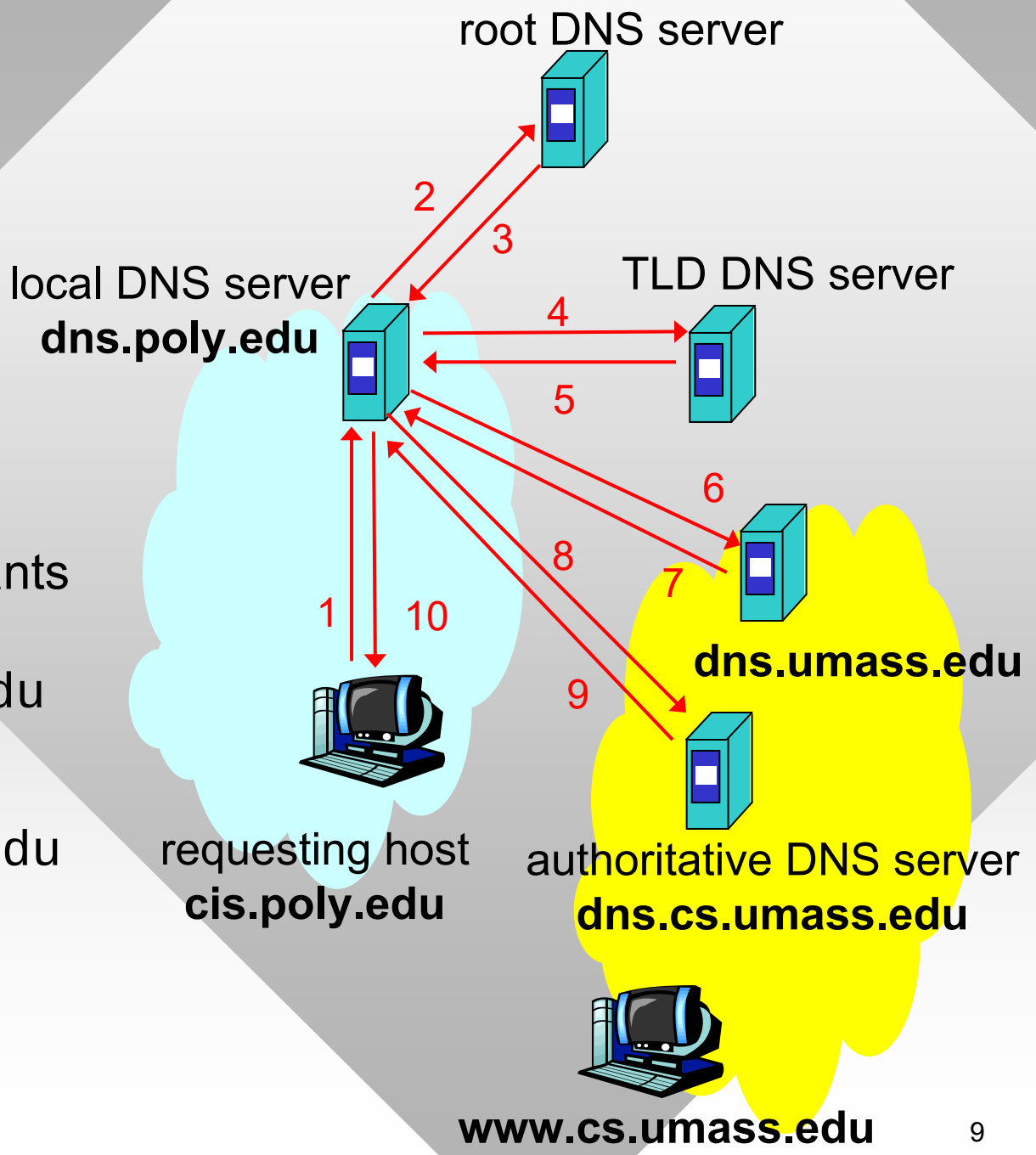
Local Name Servers



- Each network (ISP, company, university) has a few
  - Preferred DNS server in network options (alternate used for backup)
  - If you run BIND, set this to 127.0.0.1
  - Auto-configure via DHCP or set to 8.8.8.8 (Google DNS)
- When a host makes a DNS query (application calls gethostbyname), query is sent to local DNS server
  - Local server acts as a proxy (cache) and forwards query into hierarchy if it cannot answer it from cache
- Command-line tool for DNS queries is nslookup
  - Homework #2 implements essentially this

Example

- `cis.poly.edu` wants the IP address for `www.cs.umass.edu`
- Next request for `joe.cs.umass.edu` – What happens?



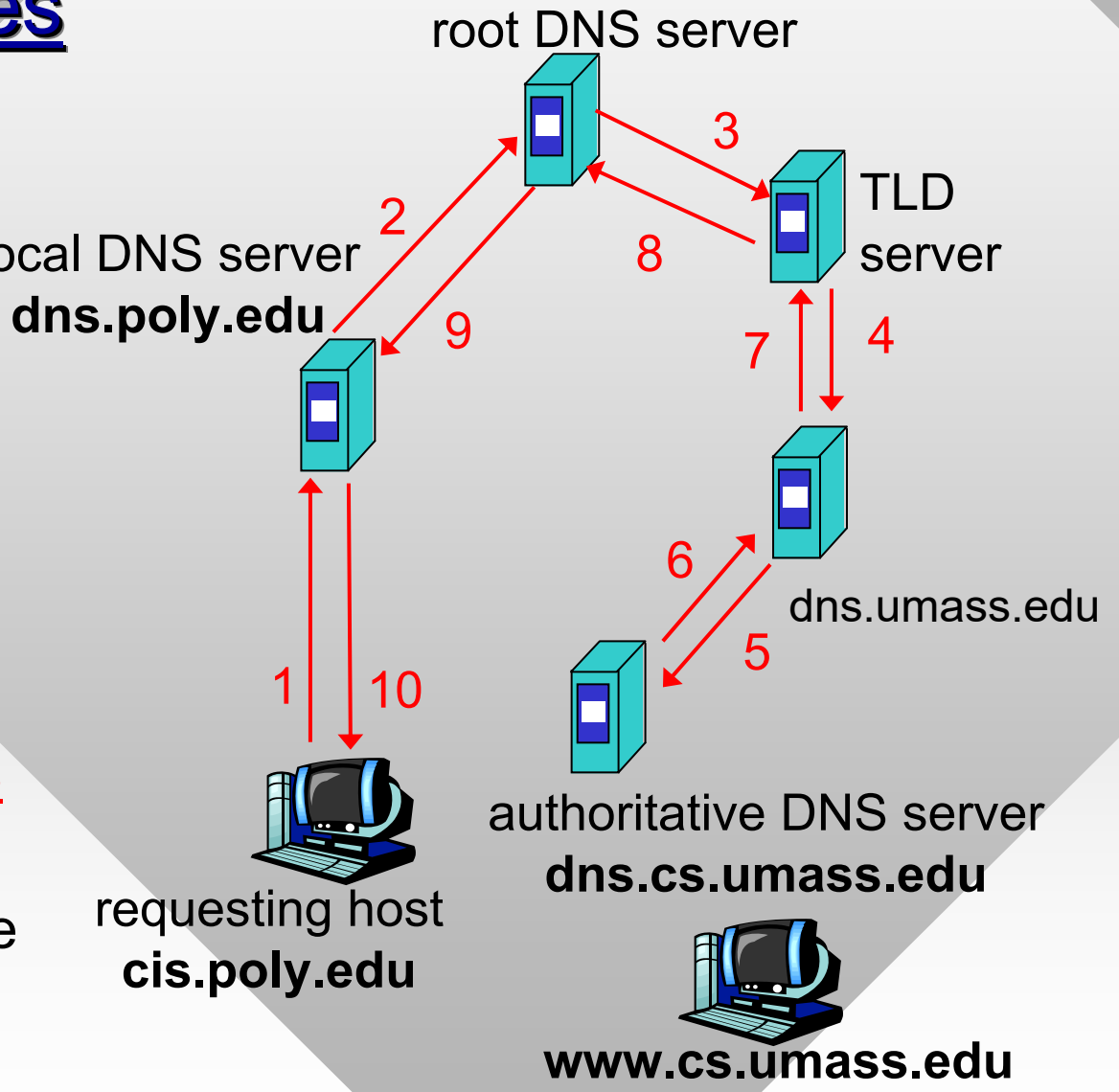
Recursive queries

Iterative query (previous page):

- Contacted server replies with name of server to contact
- "I don't know this name, but ask this other server"

Recursive query (this page):

- Puts burden of name resolution on contacted name server



DNS: Caching and Updating Records

- Once (any) name server learns a mapping, it *caches* the mapping
  - Cache entries time out (disappear) after some time (TTL)
  - Unexpired entries are served directly from cache, in which case they are called non-authoritative
  - If the DNS server of the target domain is contacted, the response is authoritative
- TLD servers are typically cached in local name servers
  - Thus root name servers not supposed to be visited often
- Study in 2007 showed load on individual root servers A-M was 6-16K queries/sec
  - During DDoS attacks in 2001 it was 38K/sec