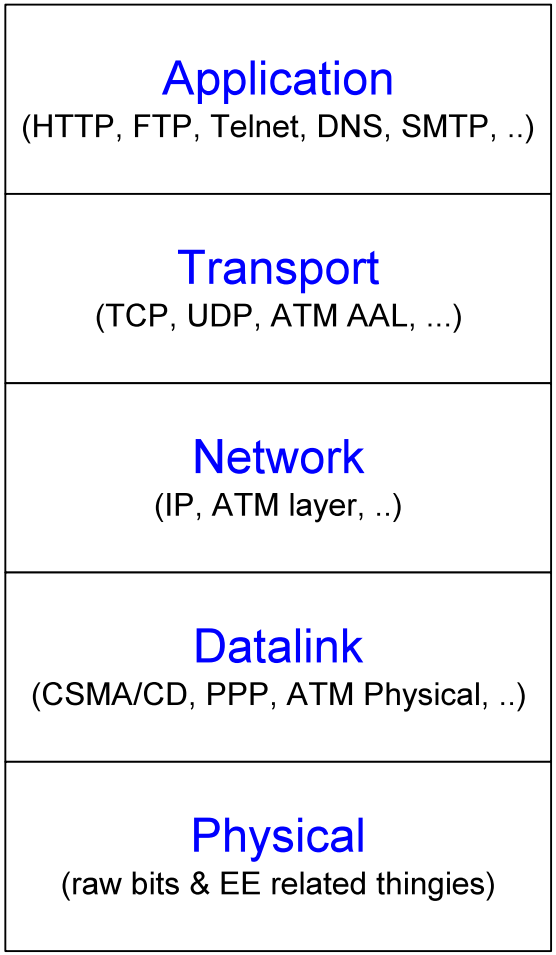
Last Lecture

- Unix Network Programming
- Berkeley Socket API

This Lecture

- Start the *Application Layer*
- DNS

TCP/IP Protocol Suite



Supports Network Applications

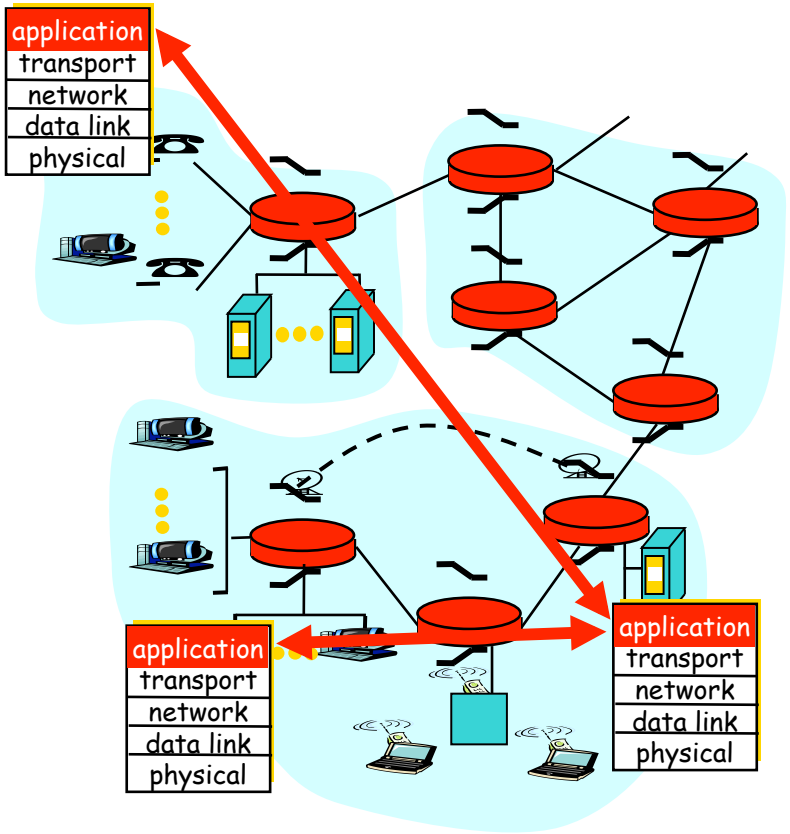
Transports applications' messages
TCP: connection-oriented, reliable
UDP: connectionless, unreliable

Routes data packets from hosts to hosts
IP: Internet Protocol, and many routing protocols

Deals with algorithms to achieve reliable, efficient communication between two adjacent machines

Moves raw bits (0/1) between adjacent nodes depending on the physical medium used

The Application Layer



A Network Application

- Is a set of processes communicating over a network
  - Web clients and servers
  - Mail clients and servers
  - FTP clients and servers
  - File sharing programs
  - DNS clients and servers
- Within the same host
  - Processes can communicate using IPC mechanisms
- Over the network
  - Processes make use of services provided by the transport layer (UDP, TCP, etc.)

Application Protocol

- For an application to work, need a protocol
- *Public-domain protocols*
  - HTTP for web clients and servers
  - SMTP for email clients and servers
  - Bit-Torrent, Gnutella, etc. for P2P servents
  - …
- *Proprietary protocols*
  - Real
  - KaZaA
  - Skype
  - The *chatty* protocol you will implement
  - …

Transport Requirements by Common Apps

Application	Data loss	Bandwidth	Time Sensitive
file transfer	no loss	elastic	no
e-mail	no loss	elastic	no
Web documents	no loss	elastic	no
real-time audio/video	loss-tolerant	audio: 5kbps-1Mbps video:10kbps-5Mbps	yes, 100's msec
stored audio/video	loss-tolerant	same as above	yes, few secs
interactive games	loss-tolerant	few kbps up	yes, 100's msec
instant messaging	no loss	elastic	yes and no

Transport Services Used by Common Apps

Application	Application layer protocol	Underlying transport protocol
e-mail	SMTP [RFC 2821]	TCP
remote terminal access	Telnet [RFC 854]	TCP
Web	HTTP [RFC 2616]	TCP
file transfer	FTP [RFC 959]	TCP
streaming multimedia	proprietary (e.g. RealNetworks)	TCP or UDP
Internet telephony	proprietary (e.g., Dialpad)	typically UDP

Case Studies of Application Layer Protocols

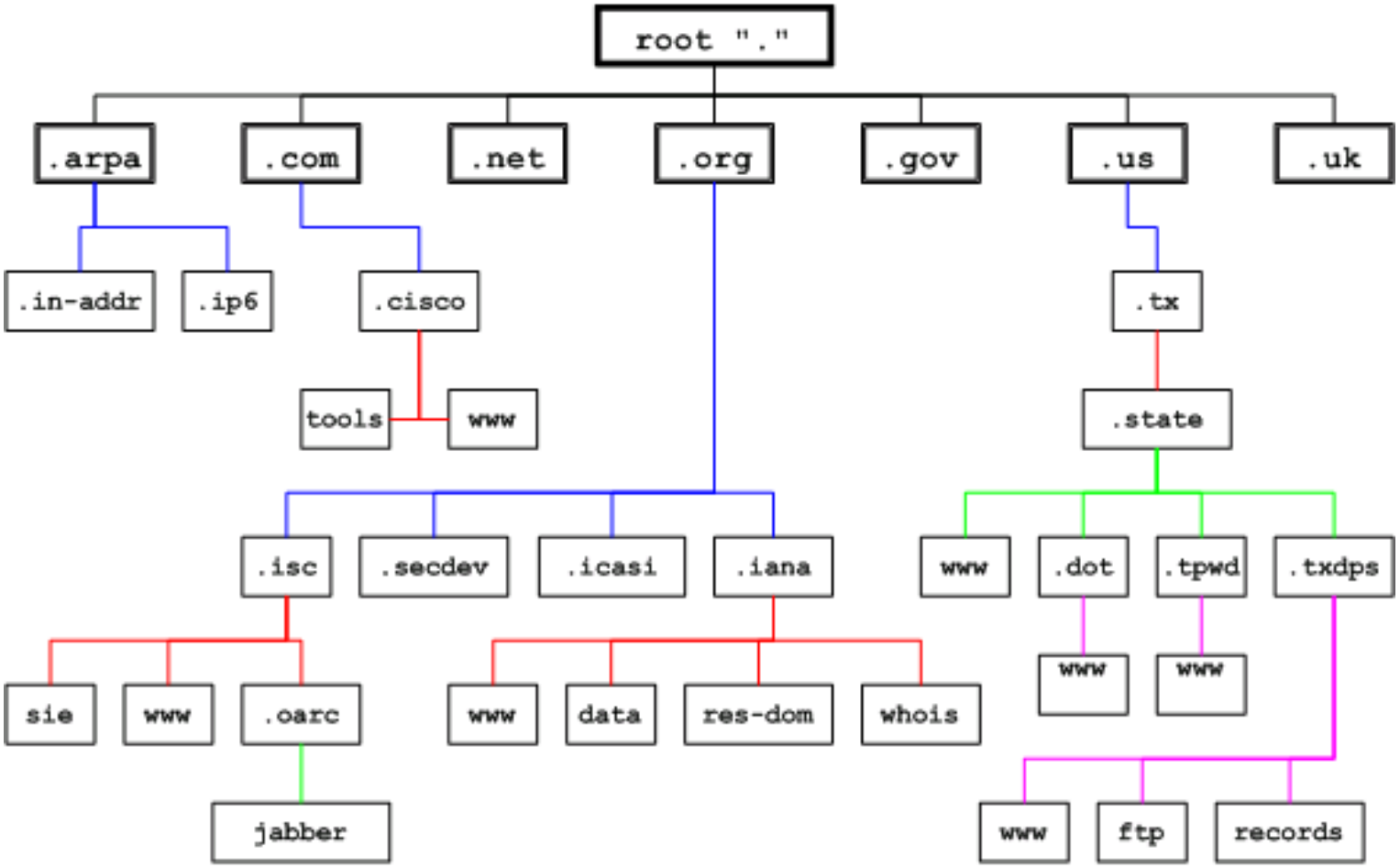
- Domain Name System (DNS)
- Email (SMTP)
- Peer-to-peer (DHT, Bit-Torrent, Gnutella)

The Domain Name System (DNS)

DNS is a *distributed database*

- Containing information about names in the *domain name space*
- Realized by *name servers*
- Maintaining a *many-to-many mapping* between *domain name space* and *IP address space*
- Allowing clients to *query* for information about a domain name
- (Partially) allowing *reverse query* (IP-to-name) too
- Providing *mail server aliasing* service

The Domain Name Space



.arpa: primarly used for address to host mappings

.com, **.net**, **.org**, **.org**: are generic TLDs (gTLD)

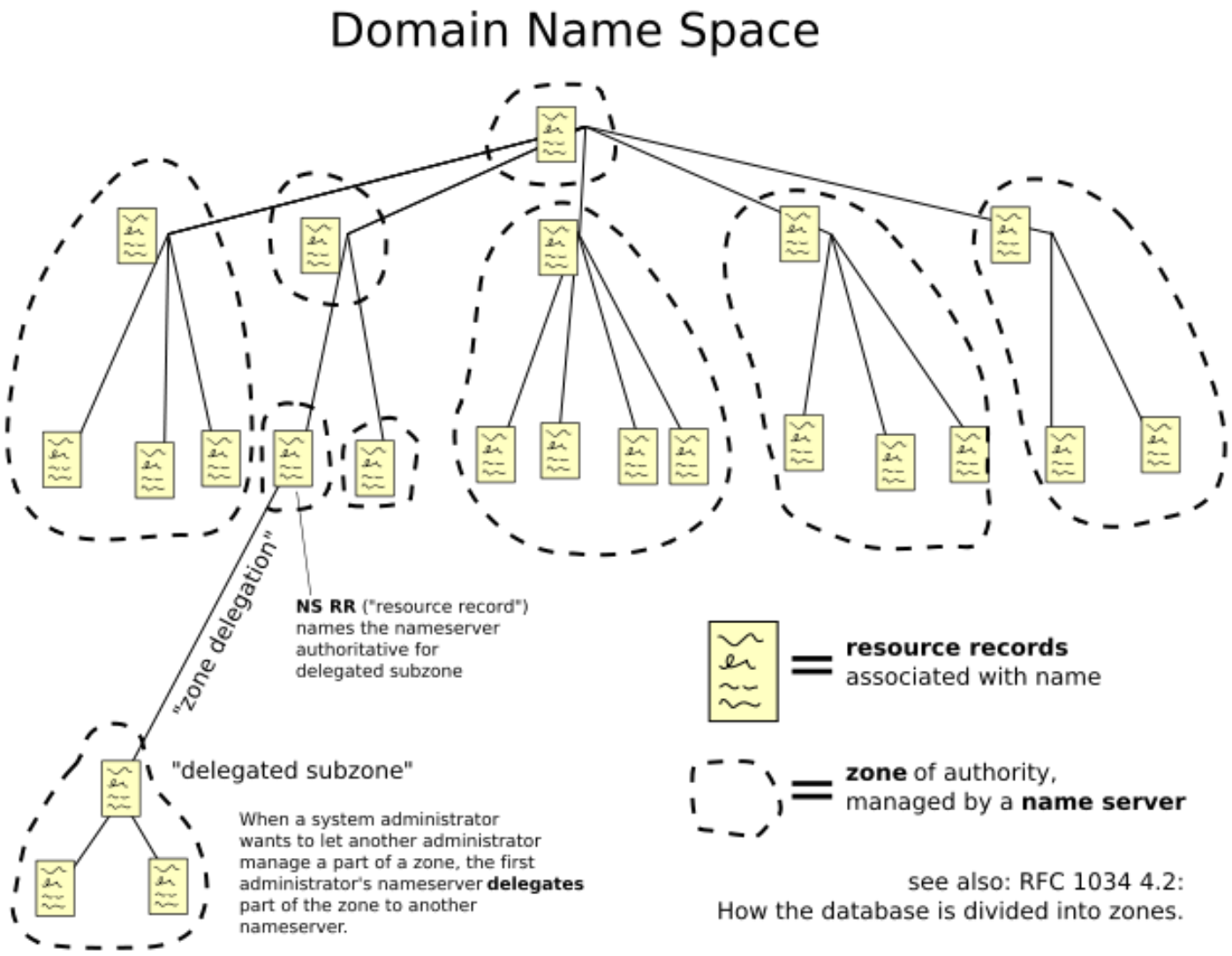
.us, **.uk**: are country code TLDs (ccTLD)

Domains and Domain Names

- A *domain* is a subtree
- A *domain name* is a node in the tree, may point to
  - Network addresses (IP address)
  - Hardware information
  - Mail routing information
  - Information about the domain rooted at that node
- Example: `buffalo.edu`
  - Is a domain name, at root of domain `buffalo.edu`
  - Points to the IP address(es) of `http://www.buffalo.edu`
    - 128.205.4.175
  - Points to the buffalo.edu *mail exchangers*

buffalo.edu.	1	IN	MX	10 mxc.acsu.buffalo.edu.
buffalo.edu.	1	IN	MX	10 mxd.acsu.buffalo.edu.
buffalo.edu.	1	IN	MX	100 smtp5.acsu.buffalo.edu.
buffalo.edu.	1	IN	MX	10 mxa.acsu.buffalo.edu.
buffalo.edu.	1	IN	MX	10 mxb.acsu.buffalo.edu.

Zones and Delegation



Name Servers

- *Name servers* are programs storing info. about the domain name space, answering queries on it

- Each zone has ≥ 1 *authoritative name server*
  - which has ***the*** info. about all nodes in the zone
  - and has delegation information for the sub-domains (i.e. authoritative name servers for the delegated sub-domains)
  - There are often > 1 authoritative name server for a given zone; e.g. buffalo.edu. has 4

- The root zone has "13" *root name servers*
  - Each of the root name servers is actually a collection of servers; more later …

Locations of "13" Root Name Servers

http://www.root-servers.org/map/

The Many-to-Many Mapping

- Each domain name can point to a list of IP addresses
  - For load balancing
  - E.g., there are quite a few web-servers for yahoo.com
    - Try "`dig yahoo.com a`"
- Each IP address can be pointed to by many domain names
  - For aliasing
  - E.g., www.cse.buffalo.edu = alfred.cse.buffalo.edu = 128.205.32.53
- By varying the "record type", cse.buffalo.edu can point to both the webserver(s) and the mail exchanger(s)

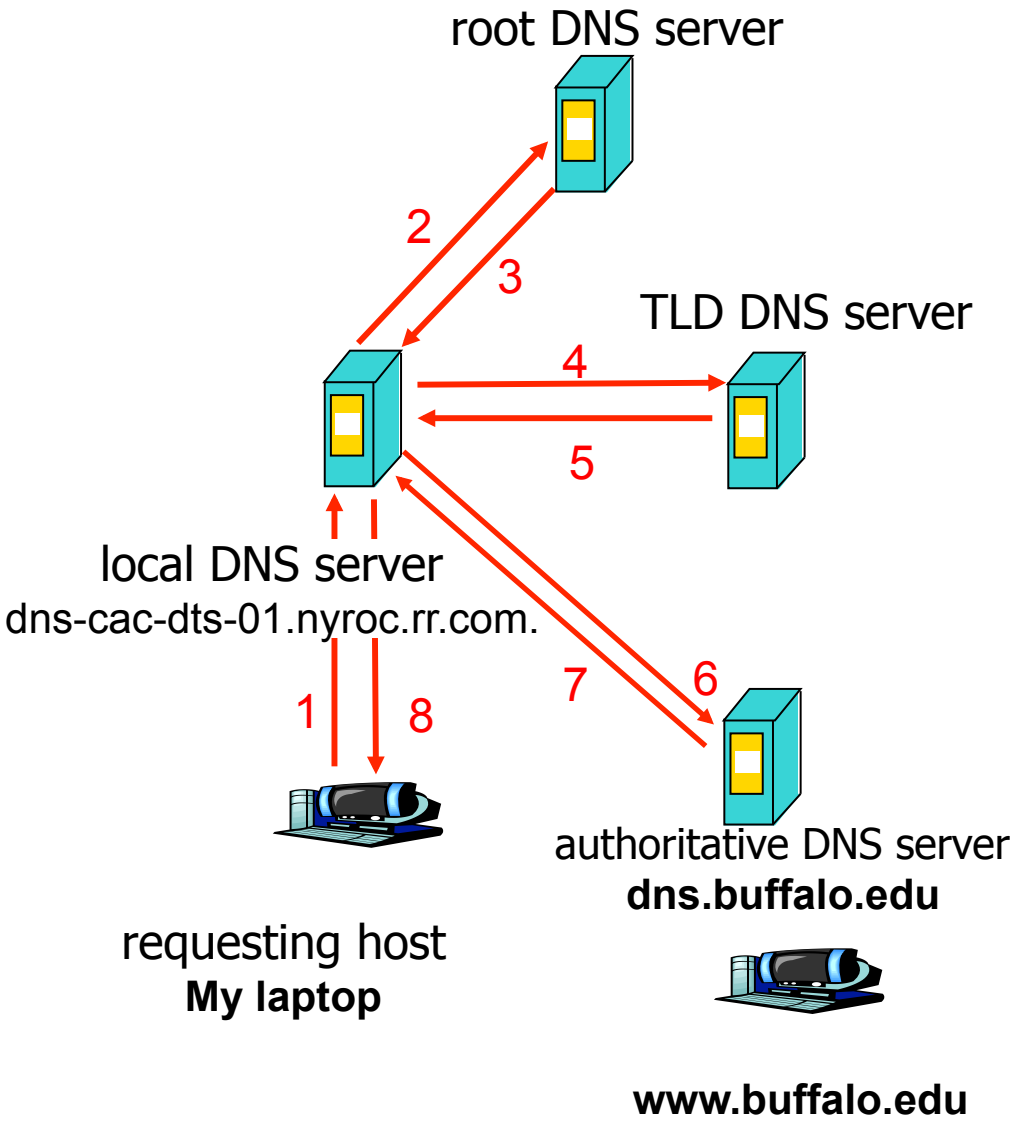
The Querying and Resolution Process

The illustrated process is *iterative*

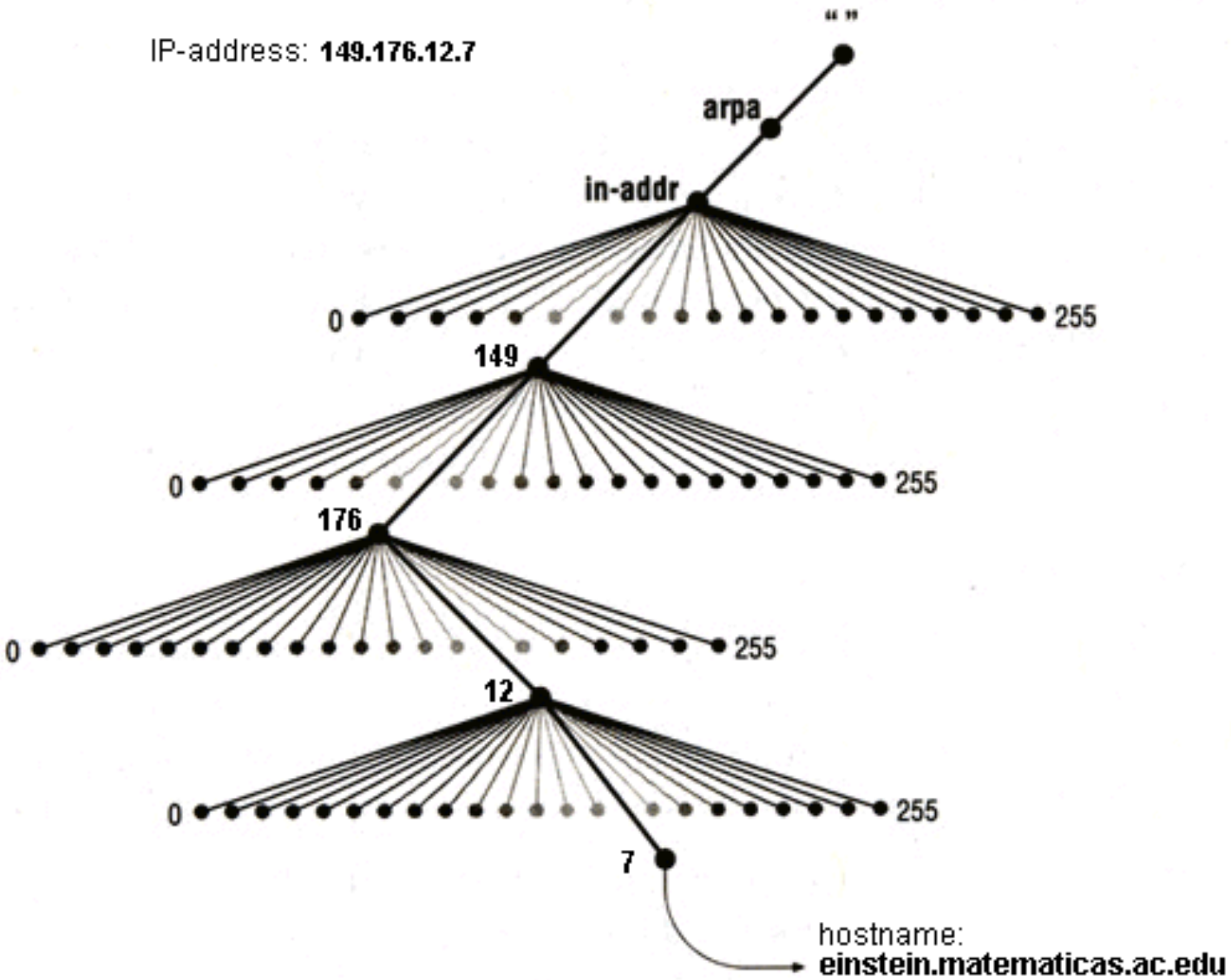
Can be *recursive* too

UDP is used! (Why?)

Most implementations of DNS servers are iterative
(that's more polite)



Reverse DNS Lookup



Records in the DNS Distributed Database

- *Resource Record* (RR): (name, value, type, TTL)
- *Type=A*:
  - name is hostname, value is IP address(es)
- Type=NS:
  - name is domain name, value is authoritative name server(s) for domain name
- *Type=CNAME*:
  - name is alias for some "canonical" name; e.g., alias=www.cse.buffalo.edu and canonical name=alfred.cse.buffalo.edu
- *Type=MX*:
  - value is name(s) of mail exchanger(s) associated with name; e.g., name=cse.buffalo.edu, value=ares.cse.buffalo.edu, themis.cse.buffalo.edu
- And dozens of other types

Performance Issues

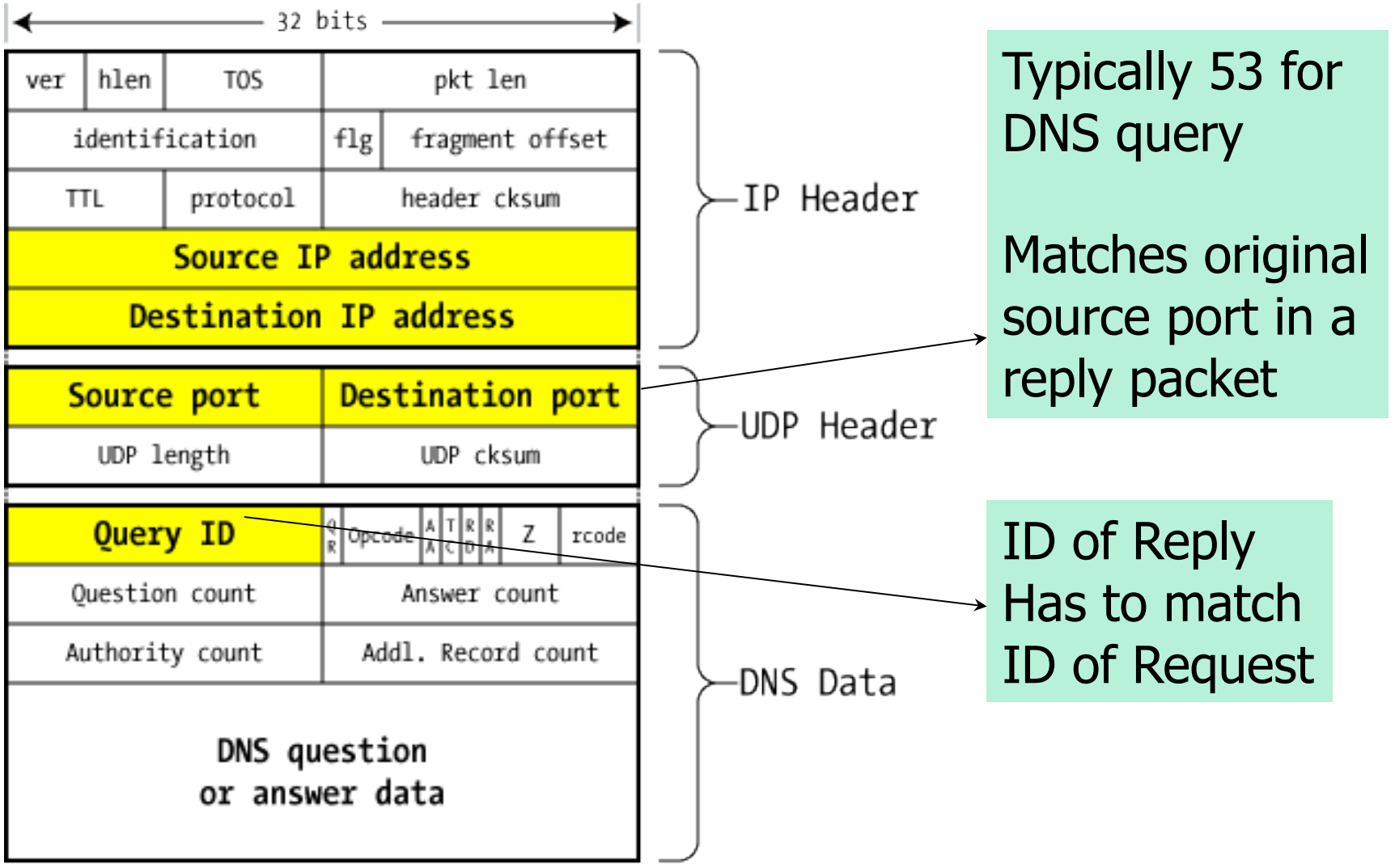
- How to reduce DNS traffic & query time?
  - Sep 2008 report: "*VeriSign processed peak loads of more than 48 billion Domain Name System (DNS) queries per day in the second quarter of 2008*"
  - Solution: **caching**
- Which server to query from a list of authoritative servers?
  - Server with shortest RTT
  - Initially all RTTs set to zero, choose random one server
  - Update RTTs dynamically with real values
- Each of the 13 root servers is actually a set of servers, how does routing work?
  - *Anycast!*

Security Issues

- DNS system susceptible to DDoS attacks
  - Slashdot, Jan 23, 2009:

    "CircleID is reporting a large-scale DDoS attack affecting all of Network Solutions' name servers for the past 48 hours, potentially affecting millions of websites and emails around the world hosting their domain names on the company's servers. The NANOG mailing list indicates that it is due to a very large-scale UDP/53 DDoS which Network Solutions has also confirmed: 'There is a spike in DNS query volumes that is causing latency for the delay in web sites resolving. This is a result of a DDOS attack. We are taking measures to mitigate the attack and speed up queries."
- DNS system susceptible to *DNS cache poisoning*

DNS Packet



Typically 53 for DNS query

Matches original source port in a reply packet

ID of Reply
Has to match
ID of Request

DNS packet on the wire

(Picture courtesy of unixwiz.net)

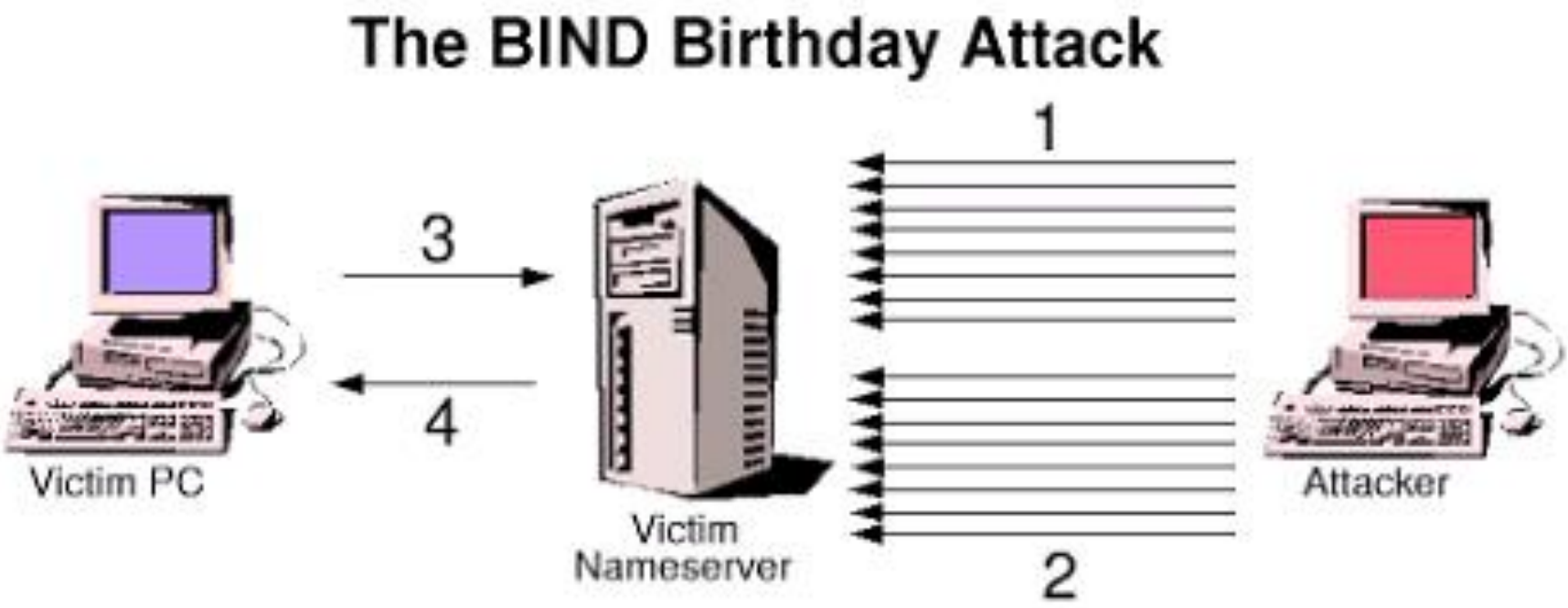
In Older Versions of BIND (8 and before)

- Source port for QUERY packet is fixed for each client
- Query IDs are sequential

- Thus, cache poisoning was very *easy* before 2001
  - Query your own domain first to get Source port
  - Then query `HSBC.com` and fake the replies
  - You're now the owner of `HSBC.com`

- Solution: they randomize Query ID, the "guessing space" is now 65535

Cache Poisoning – The Birthday Attack



Assuming Source Port is Known

- Faked replies only need to match Query ID
- There are $n = 65535$ possible IDs
- Attacker sent, say, k faked replies following m faked querries,
  - Probability[no reply ID matches any query ID] =

$$(1 - m/n)^k \leq e^{-mk/n}$$

  - This "failure" probability is at most ½ if

$$mk \geq n \ln 2 \approx 45425$$

  - Thus, attacker can choose, say,

$$m = k = \sqrt{45425} \approx 213$$

Headlines in 2008

- Jul 08 - Largest Synchronized Internet Security Effort Underway to Patch Newly Found DNS Bug
- Jul 09 - An Astonishing Collaboration
- Jul 14 - Not a Guessing Game
- Jul 21 - DNS Security Flaw Secret Leaked Prior to Set Date: Patch DNS as Fast as Possible
- Jul 22 - Just a Matter of Time Before DNS Attack Code Might Surface
- Jul 23 - DNS Attack Code Has Been Published
- Jul 24 - US-CERT Says They Are Aware of DNS Exploit Code, Emphasizes Urgent Patching
- Jul 28 - Possible First Attacks on DNS Flaw Have Been Reported
- Jul 30 - DNS Attack Creator Becomes a Victim of His Own Creation
- Aug 06 - Kaminsky DNS Bug Disclosure

Kaminski's Bug

- If you ask some DNS server for www.hsbc.com, the answer typically looks like this:

```
;; ANSWER SECTION:
;; empty

;; AUTHORITY SECTION:
hsbc.com.          86400   IN  NS   ns1.hsbc.com.

;; ADDITIONAL SECTION:
ns1.hsbc.com.      604800  IN  A    10.10.10.20
```

- In English: "*I don't know, but I know ns1.hsbc.com knows, and its IP is 10.10.10.20*"

- The last part is called the *glue* to help the questioner contact the authoritative server

Kaminski Says

- Put this in the fake replies:

```
;; ANSWER SECTION:
;; empty
;; AUTHORITY SECTION:
hsbc.com.          86400   IN  NS   ns1.hsbc.com.

;; ADDITIONAL SECTION:
ns1.hsbc.com.      604800   IN  A    10.10.10.99
```

- *Or, even more blatantly*

```
;; ANSWER SECTION:
;; empty

;; AUTHORITY SECTION:
hsbc.com.          86400   IN  NS   www.hsbc.com.

;; ADDITIONAL SECTION:
www.hsbc.com.      604800   IN  A    99.99.99.99
```

But

- Some system administrators say
  - I don't accept DNS requests from the outside, only from my own customers

- Multiple requests (few hundreds) for hsbc.com must be sent for the birthday attack to work

Kaminski Says

- Create a webpage at http://phishing.org with lots of images
  - Image 1: <img src="1.hsbc.com/img1.jpg">
  - Image 2: <img src="2.hsbc.com/img2.jpg">
  - …
- Trick a customer to click on a link to http://phishing.org
  - Go here to get cheap Viagara
  - Go here to submit your paper to an IEEE conference
- Visiting the site triggers multiple DNS requests, voila!

A Fix: Randomize Query IDs **and** Ports

- This is the current recommended method
- Most DNS server implementations obey, some don't!
- Potential Problems:
  - Bad (pseudo) random number generators, allowing guessing a query ID from a previous query ID
  - Has to set up firewall to ring alarm when there's a flood of DNS replies; how many replies should trigger alarm?
- DNSSEC is coming