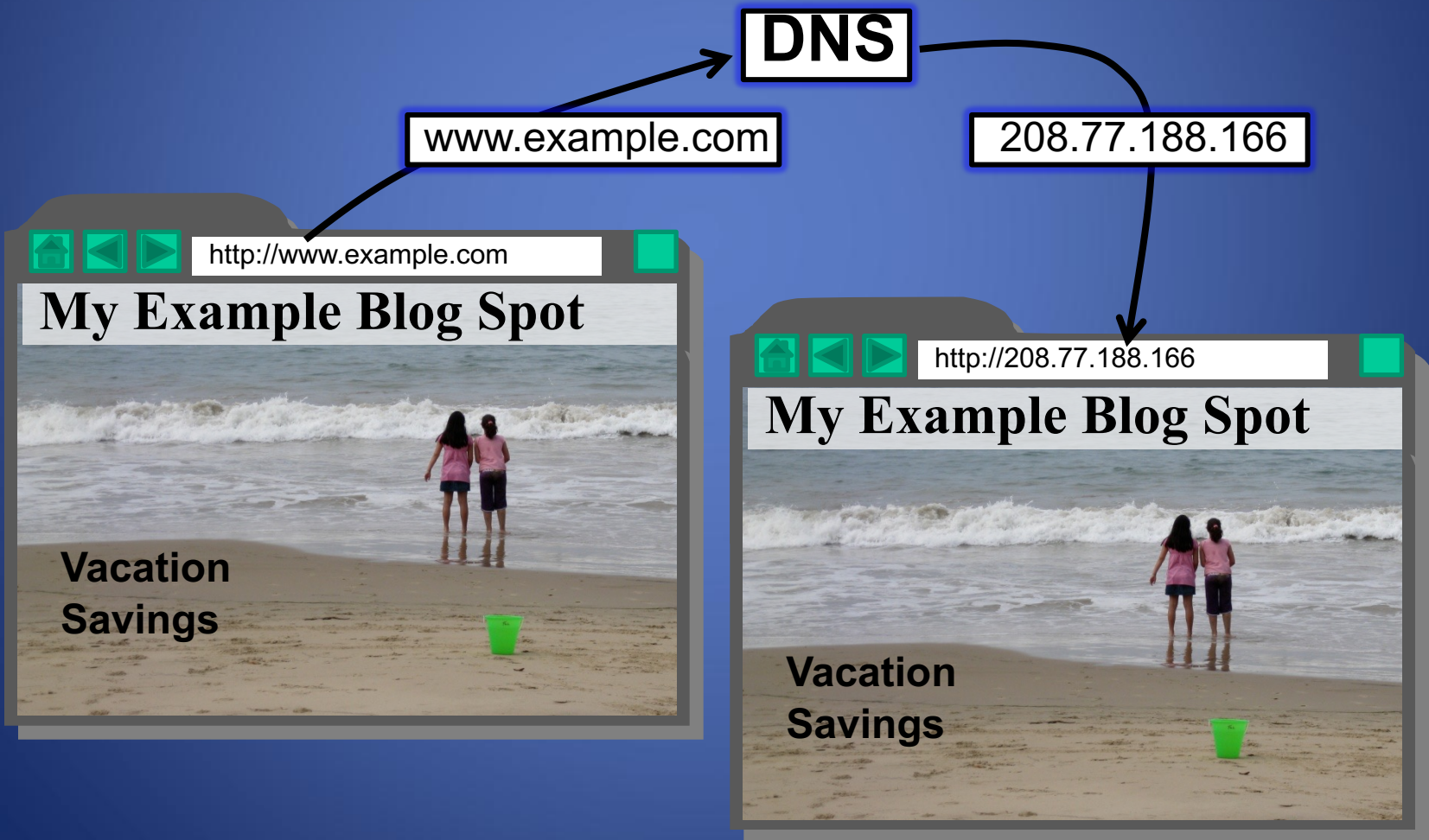


# Computer Networks: Domain Name System

# Domain Name System

- The **domain name system** (DNS) is an application-layer protocol for mapping domain names to IP addresses



# Domain Name System

- DNS provides a distributed database over the internet that stores various **resource records**, including:
  - **Address (A)** record: IP address associated with a host name
  - **Mail exchange(MX)** record: mail server of a domain
  - **Name server (NS)** record: authoritative server for a domain

For example, if example.com wishes to sub-delegate "john.example.com." to John who works at Example, inc., lines like this can be added to the example.com zone file:

```
john.example.com. NS ns1.john.example.com.
john.example.com. NS ns2.john.example.com.
# It's important to provide "glue"; in other words, let the world know
# the IPs for these name servers.
ns1.john.example.com. 10.9.8.7
ns2.john.example.com. 10.5.77.65
```

John, who is running his own nameservers with the IPs 10.9.8.7 and 10.5.77.65 then has a zone file for john.example.com. that looks something like this:

```
# It is best if the NS records for a subzone agree with the delegation
# records above
john.example.com. NS ns1.john.example.com.
john.example.com. NS ns2.john.example.com.

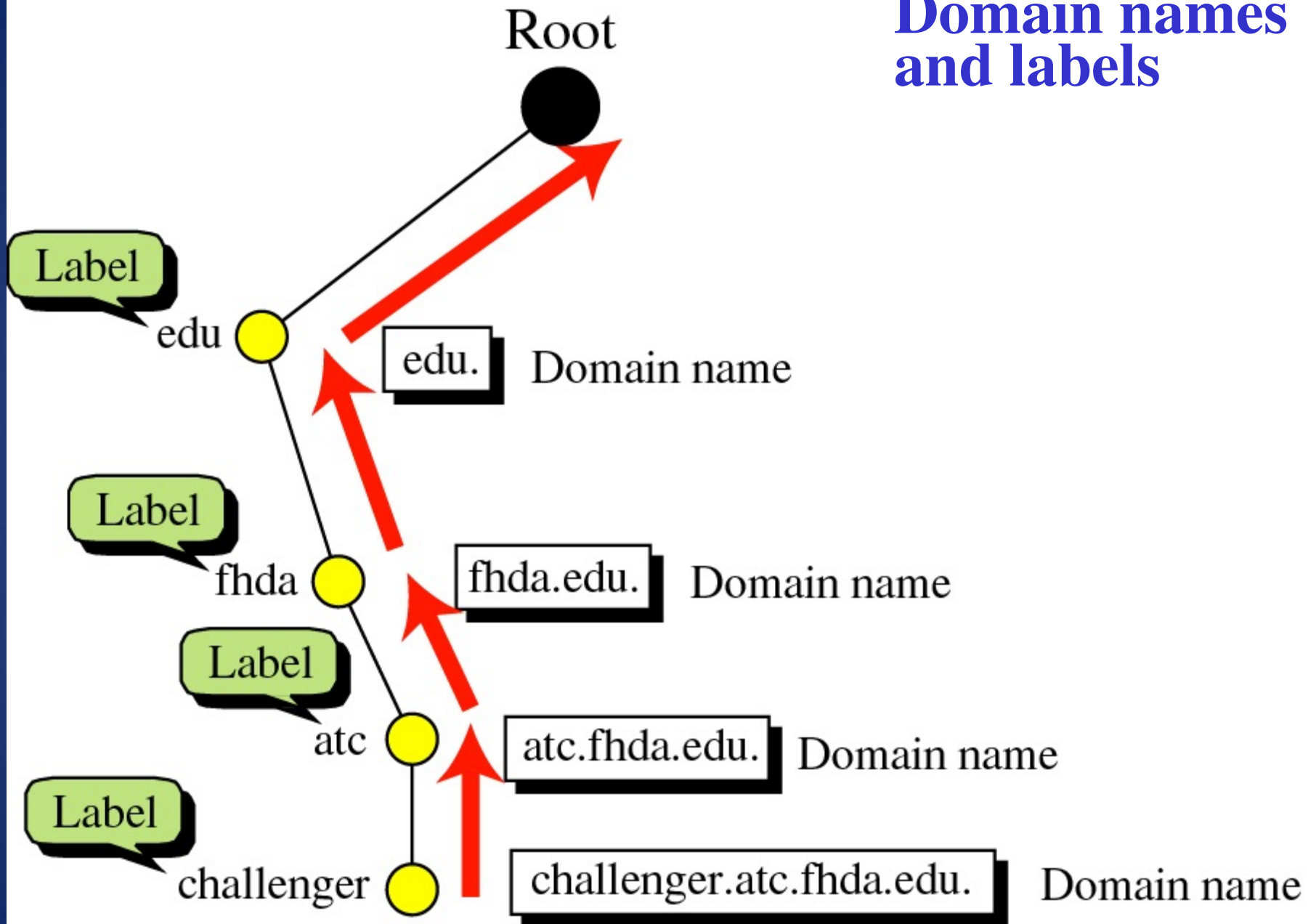
ns1.john.example.com. 10.9.8.7
ns2.john.example.com. 10.5.77.65

# Now that that is out of the way, here is the rest of the zone
john.example.com. 10.9.8.7
www.john.example.com. 10.5.77.65
john.example.com. MX 10 mail.john.example.com.
mail.john.example.com. 10.9.8.7
```

# Name Servers

- Domain names:
  - Two or more labels, separated by dots (e.g., cs166.net)
  - Rightmost label is the **top-level domain** (TLD)
- Hierarchy of **authoritative name servers**
  - Information about root domain
  - Information about its subdomains (A records) or references to other name servers (NS records)
- The authoritative name server hierarchy matches the domain hierarchy: root servers point to DNS servers for TLDs, etc.
- Root servers, and servers for TLDs change infrequently
- DNS servers refer to other DNS servers by name, not by IP: sometimes must bootstrap by providing an IP along with a name, called a glue record

# Domain names and labels



# Namespace Management

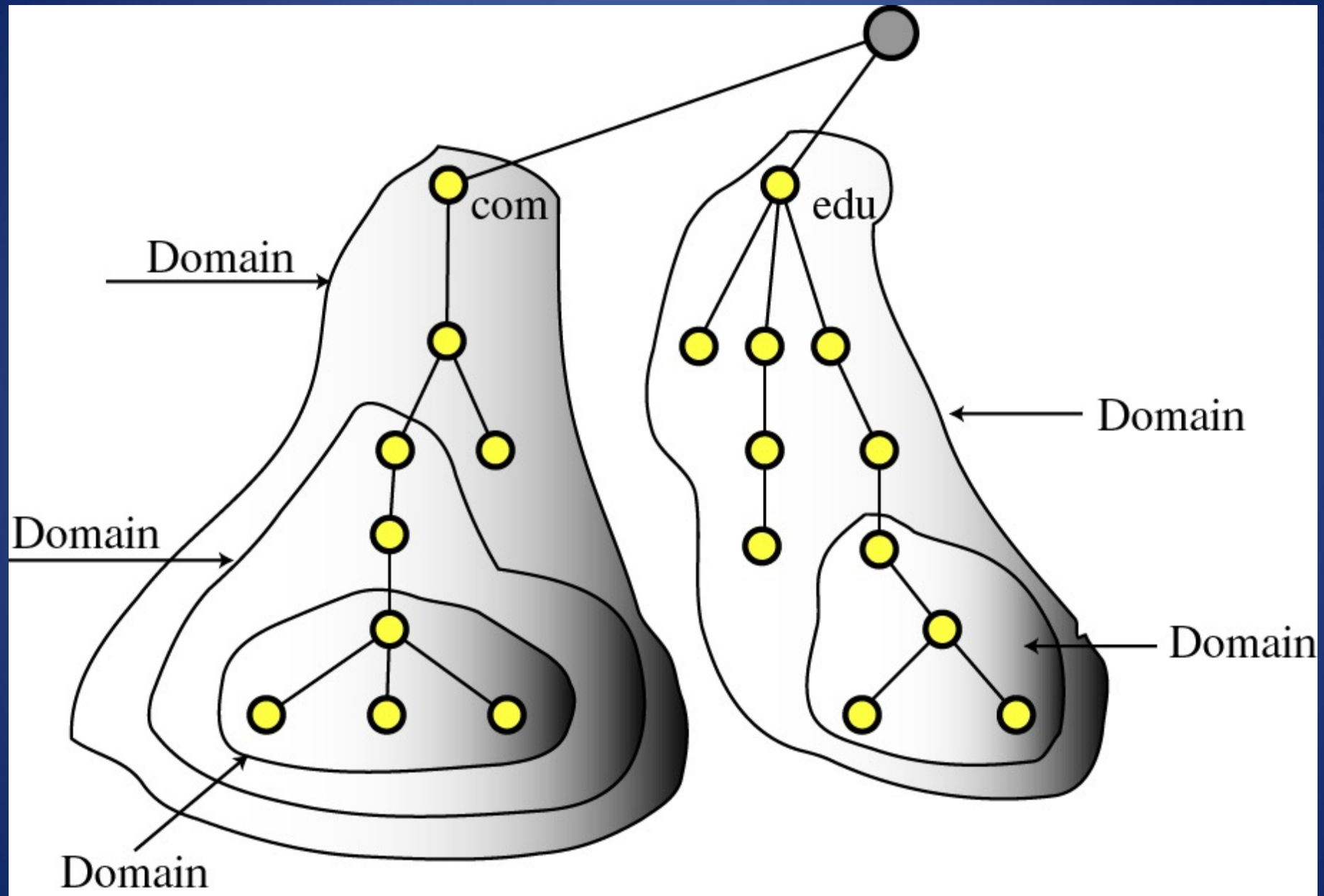
- ICANN: Internet Corporation for Assigned Names and Numbers
- ICANN has the overall responsibility for managing DNS. It controls the root domain, delegating control over each top-level domain to a domain name registry
- Along with a small set of general TLDs, every country has its own TLD -- (cTLDs) – controlled by the government.
- ICANN is the governing body for all general TLDs
- Until 1999 all .com, .net and .org registries were handled by Network Solutions Incorporated.
- After November, 1999, ICANN and NSI had to allow for a shared registration system and there are currently over 500 registrars in the market
- Also since 1999, ICANN has created additional gTLDs including some which are sponsored by consortiums or groups of companies.

# Top Level Domains

- Started in 1984
- Originally supposed to be named by function
  - .com for commercial websites, .mil for military
- Eventually agreed upon unrestricted TLDs for .com, .net, .org, .info
- In 1994 started allowing country TLDs such as .it, .us
- Tried to move back to hierarchy of purpose in 2000 with creation of .aero, .museum, etc.

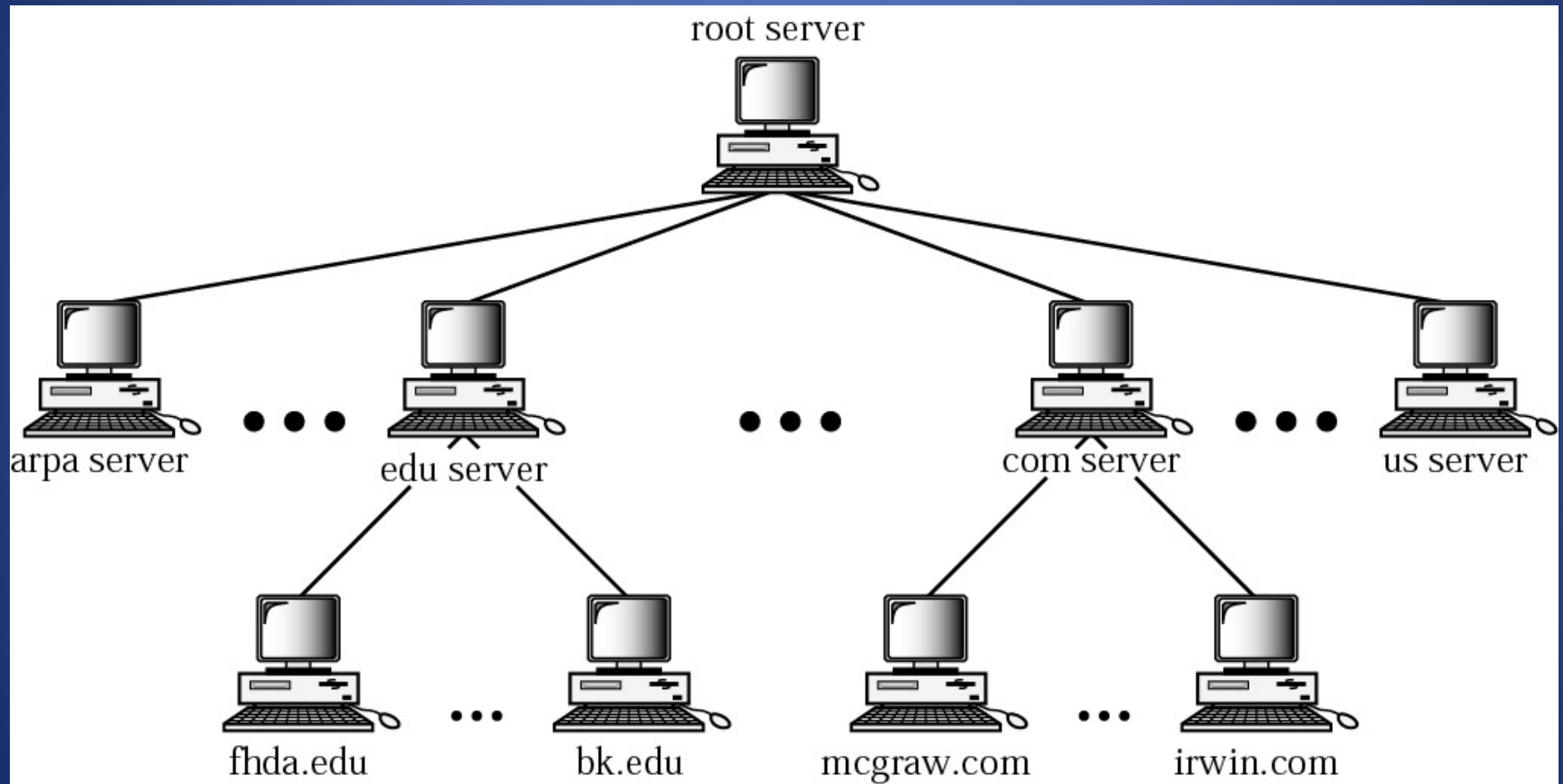


# Domains



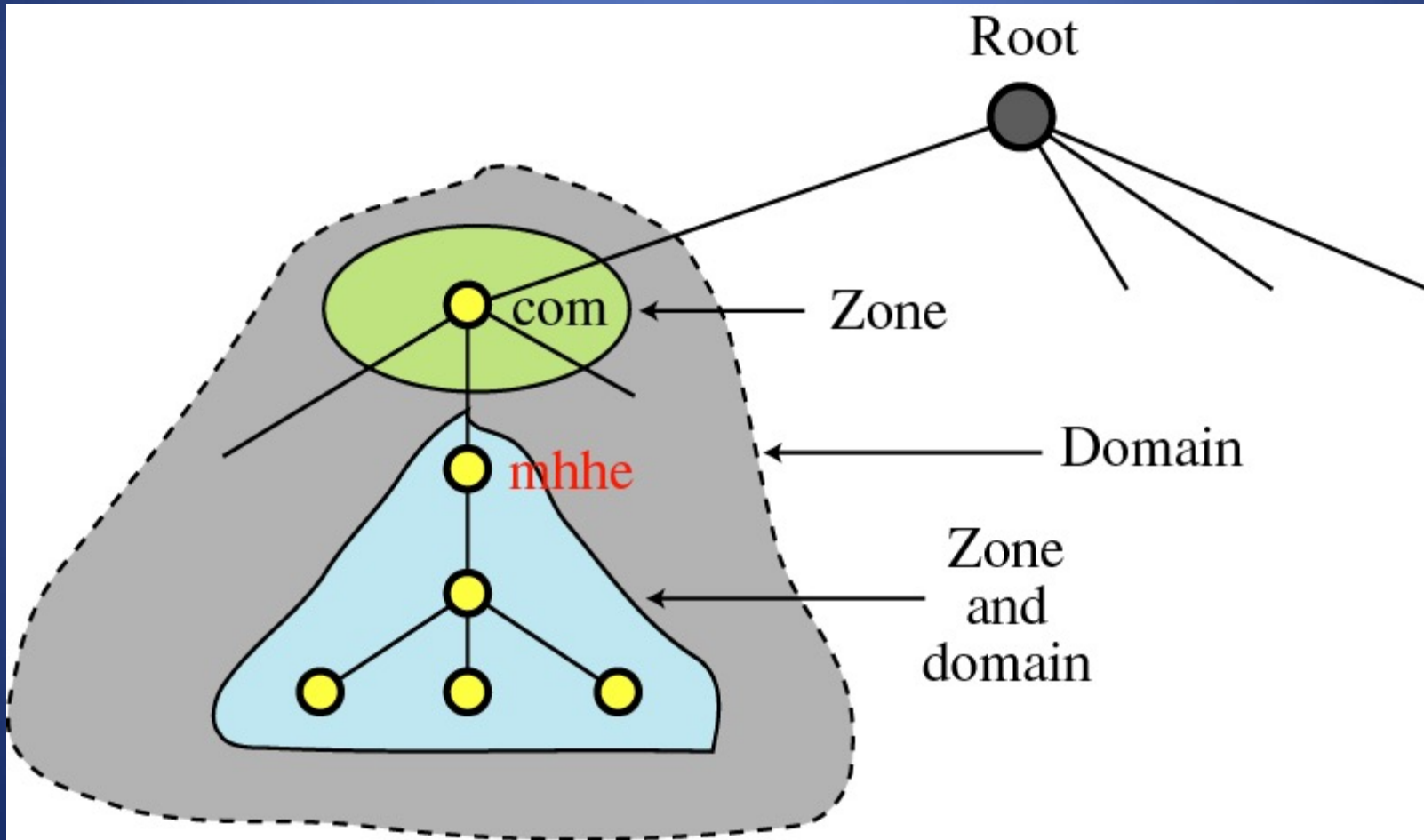


# Hierarchy of name servers



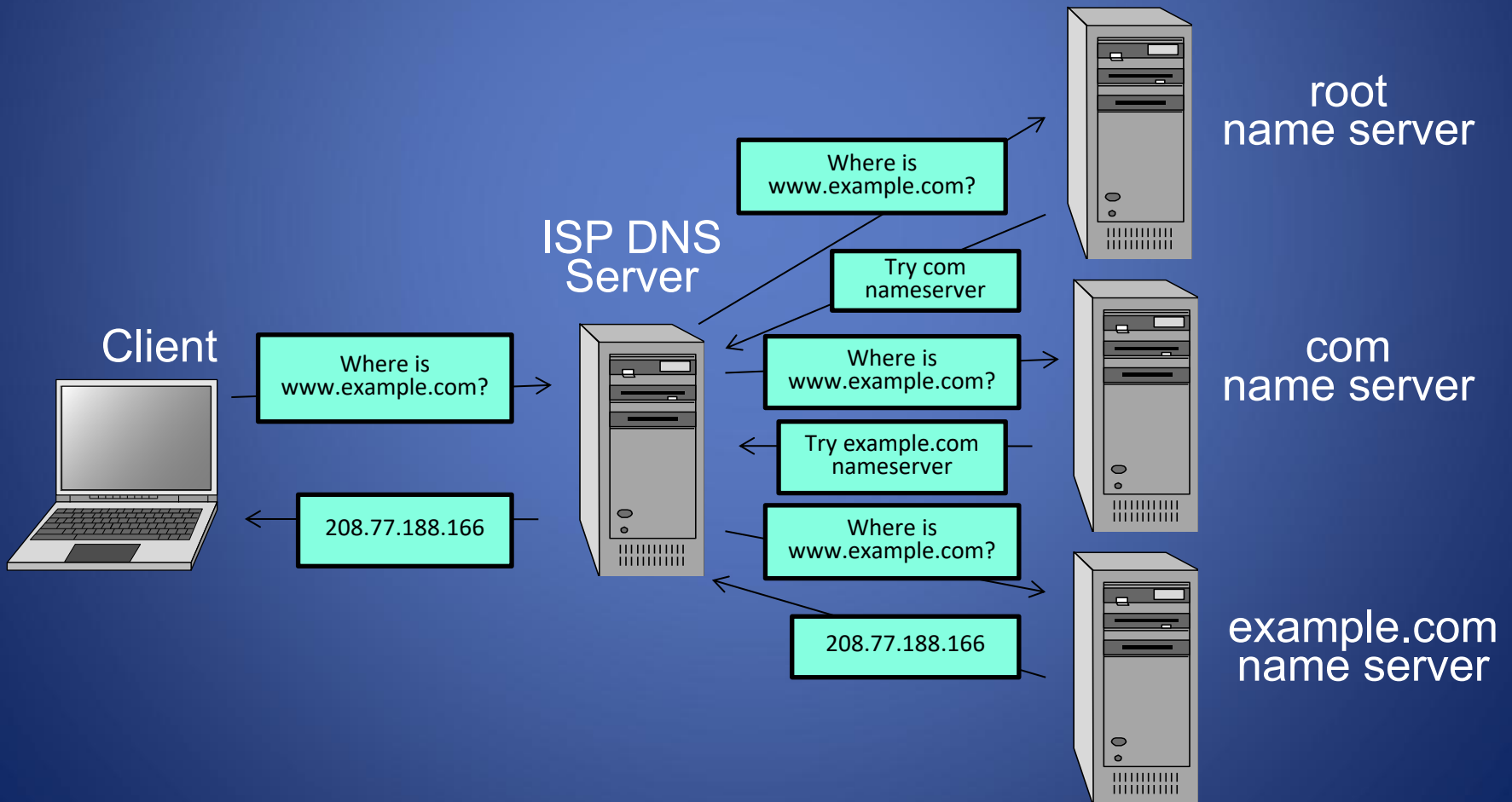
# Zones and domains

- **Zone**: collection of connected nodes with the same authoritative DNS server

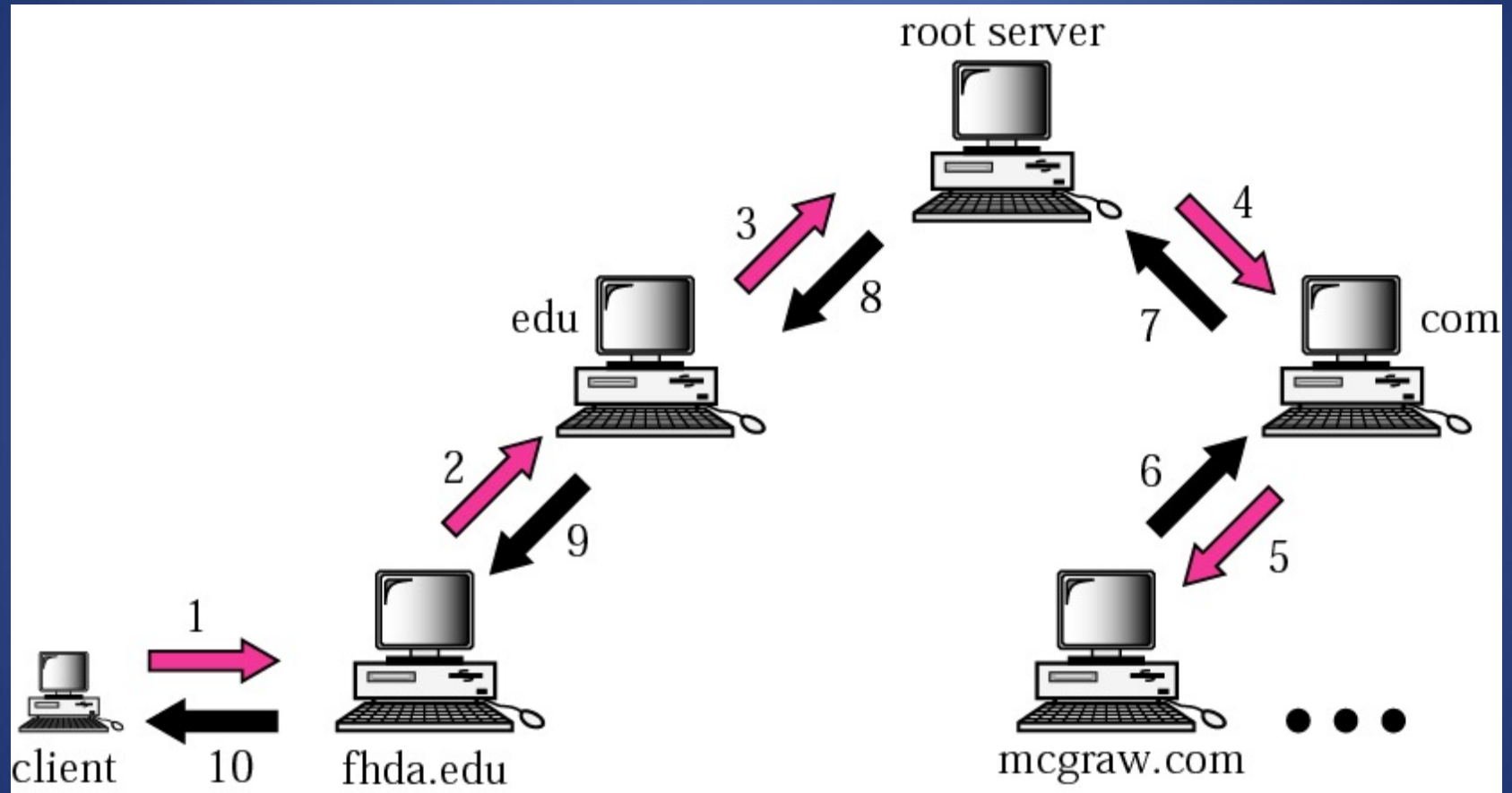


# Name Resolution

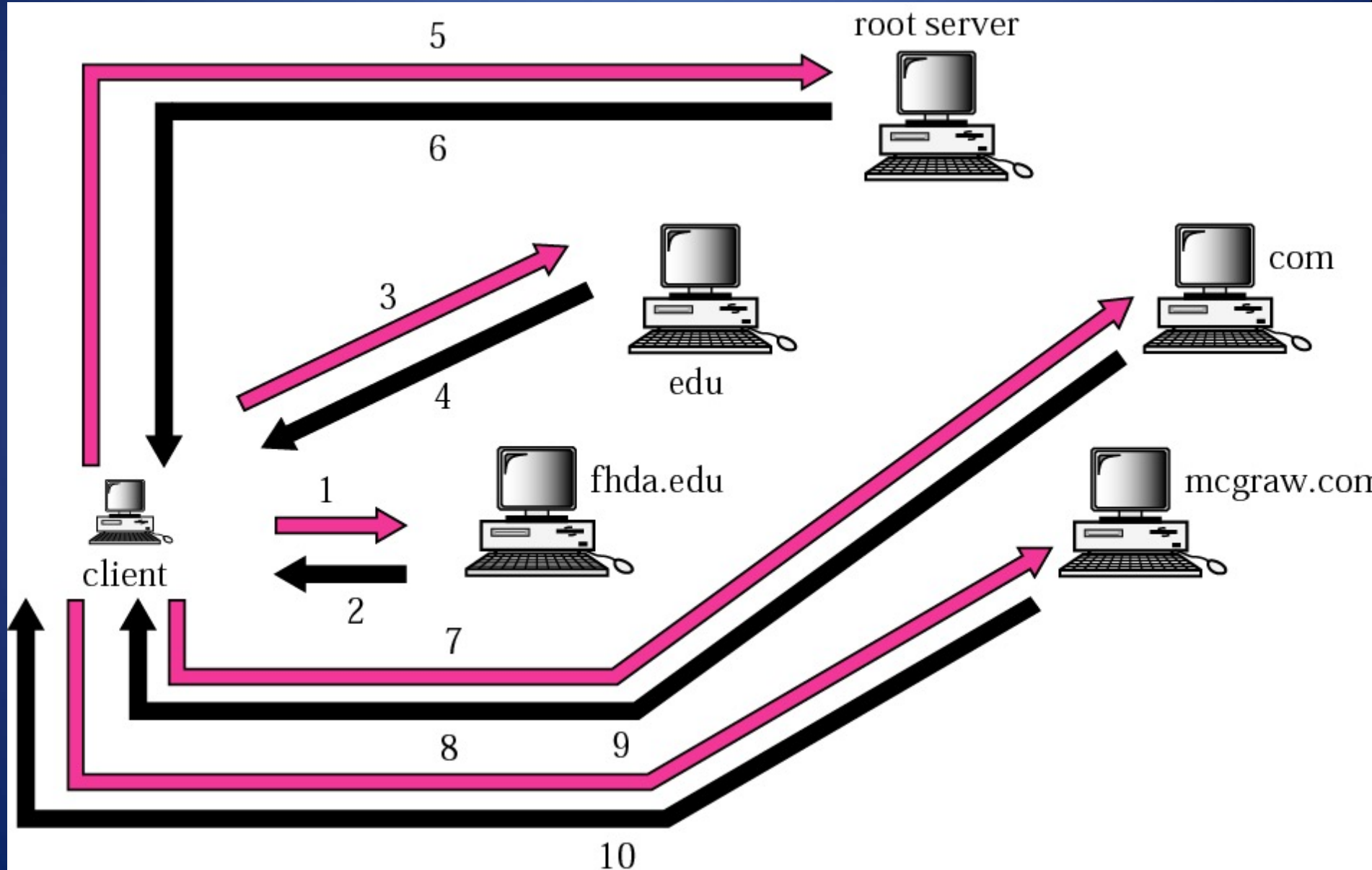
- Resolution method when answer not in cache:



# Recursive resolution



# Iterative resolution



# Authoritative Name Servers

- Control distributed among authoritative name servers (ANSs)
  - Responsible for specific domains
  - Can designate other ANS for subdomains
- ANS can be master or slave
  - Master contains original zone table
  - Slaves are replicas, automatically updating
- Makes DNS fault tolerant, automatically distributes load
- ANS must be installed as a NS in parents' zone

# Dynamic Resolution

- Many large providers have more than one authoritative name server for a domain
- Problem: need to locate the instance of domain geographically closest to user
- Proposed solution: include first 3 octets of requester's IP in recursive requests to allow better service
- Content distribution networks already do adaptive DNS routing

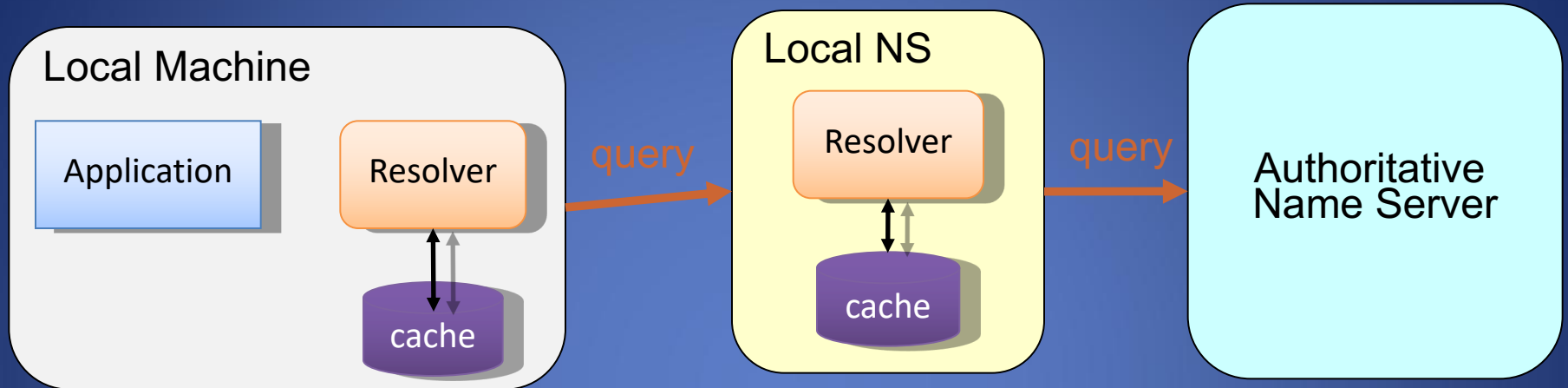


# DNS Caching

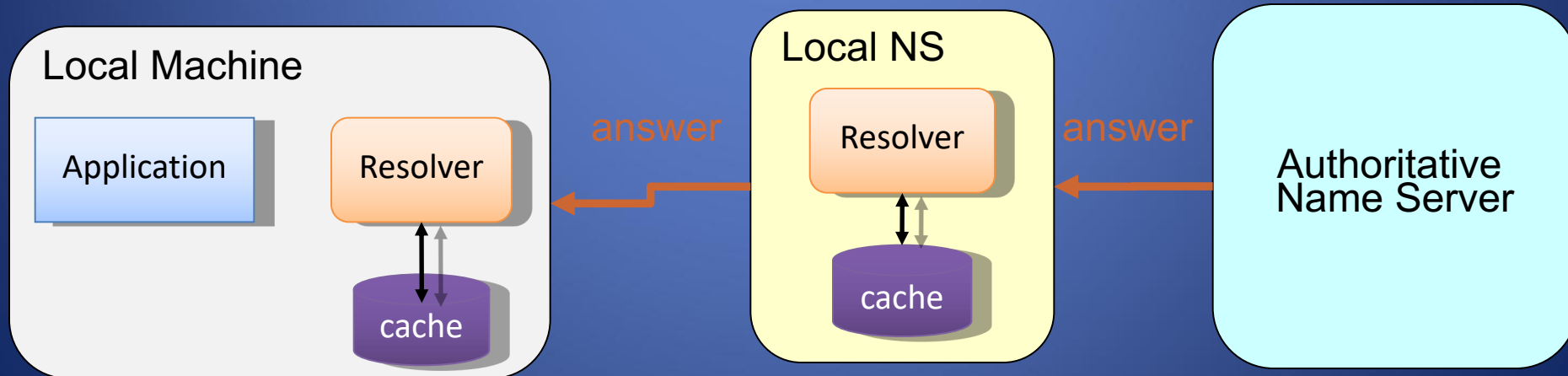
- There would be too much network traffic if a path in the DNS tree would be traversed for each query
  - Root zone would be rapidly overloaded
- DNS servers **cache** results for a specified amount of time
  - Specified by DNS **reply's time-to-live field**
- Operating systems and browsers also maintain resolvers and DNS caches
  - View in Windows with command **ipconfig /displaydns**
  - Associated privacy issues
- DNS queries are typically issued over UDP on port 53
  - 16-bit request identifier in payload

# DNS Caching

Step 1: query yourdomain.org

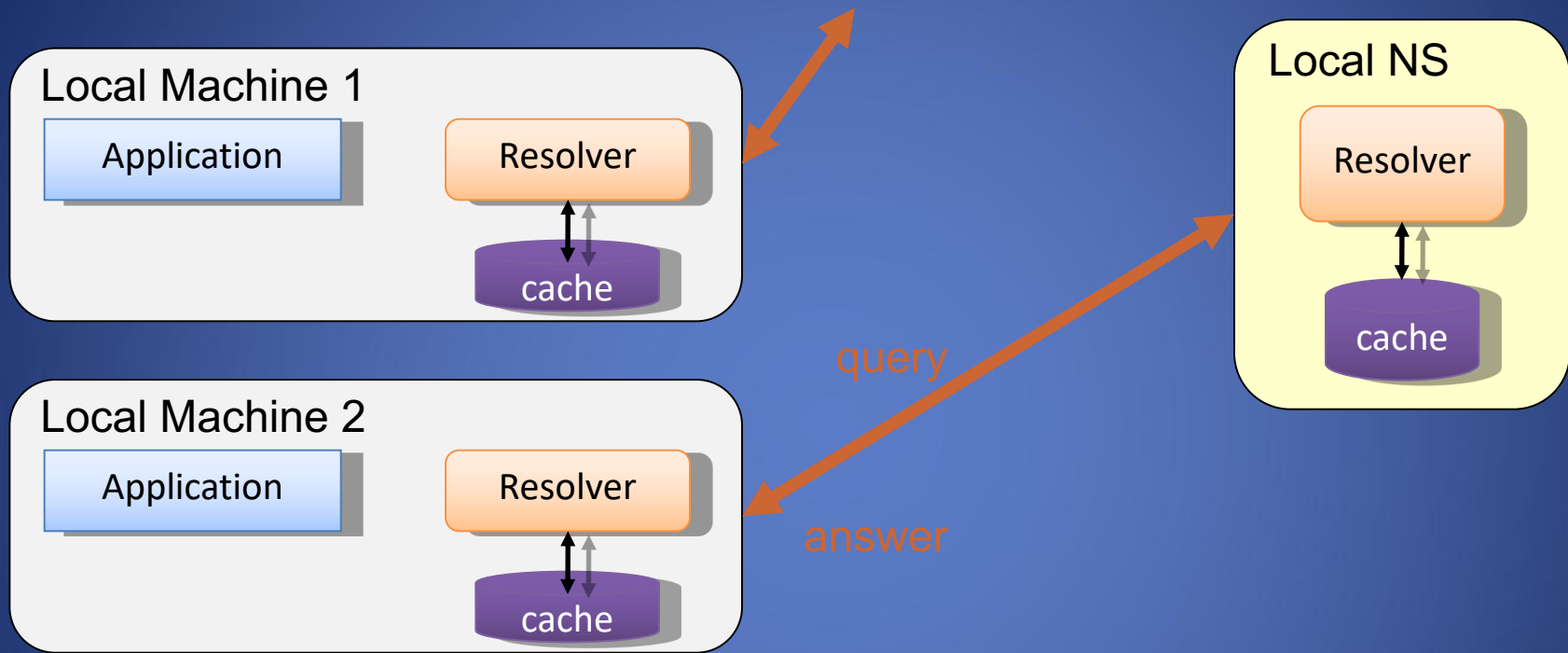


Step 2: receive reply and cache at local NS and host



# DNS Caching (con'd)

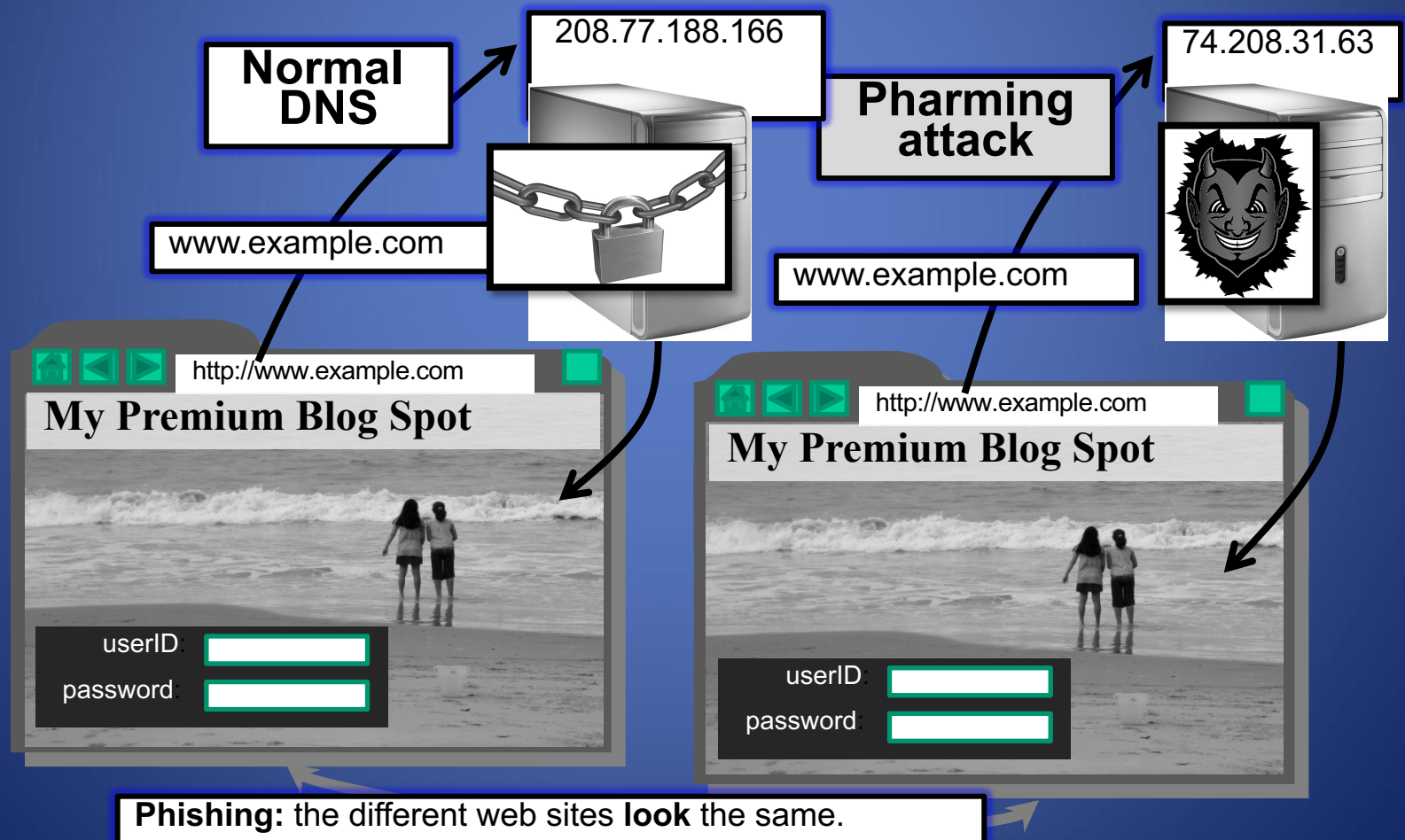
Step 3: use cached results rather than querying the ANS



Step 4: Evict cache entries upon ttl expiration

# Pharming: DNS Hijacking

- Changing IP associated with a server maliciously:



# DNS Cache Poisoning

- Basic idea: give DNS servers false records and get it cached
- DNS uses a 16-bit request identifier to pair queries with answers
- Cache may be poisoned when a name server:
  - Disregards identifiers
  - Has predictable ids
  - Accepts unsolicited DNS records

# DNS Cache Poisoning Prevention

- Use random identifiers for queries
- Always check identifiers
- Port randomization for DNS requests
- Deploy DNSSEC
  - Challenging because it is still being deployed and requires reciprocity

# DNSSEC

- Guarantees:
  - Authenticity of DNS answer origin
  - Integrity of reply
  - Authenticity of denial of existence
- Accomplishes this by signing DNS replies at each step of the way
- Uses public-key cryptography to sign responses
- Typically use trust anchors, entries in the OS to bootstrap the process



# DNS Signing

Refer to page 286

# DNSSEC Deployment

- As the internet becomes regarded as critical infrastructure there is a push to secure DNS
- NIST is in the process of deploying it on root servers now
- May add considerable load to dns servers with packet sizes considerably larger than 512 byte size of UDP packets
- There are political concerns with the US controlling the root level of DNS