Domain Name System

Lecture -03

Department of Physical Science

Faculty of Applied Science

University of Vavuniya

Contents

- Domain Name system
- DNS Components
- DNS naming Hierarchy
- DNS Resolution process
- Features of DNS
- Benefits and Limitations of DNS

Introduction

- IP assigns 32-bit addresses to hosts (interfaces)
- Binary addresses easy for computers to manage.
- All applications use IP addresses through the TCP/IP protocol software.
- Difficult for humans to remember.
- For example, remembering www.yahoo.com is much easier than remembering the 32- bit IP address.
- The Domain Name System (DNS) provides translation between symbolic names and IP addresses.
- The Domain Name System (DNS) is a supporting client-server program that is used by other programs such as e-mail.

Purpose of Naming

- Addresses are used to locate objects
- Names are easier to remember than numbers
- You would like to get to the address or other objects using a name
- DNS provides a mapping from names to resources of several types

History of DNS

- In the early days of the Internet, the responsibility for maintaining unique host names for the computers on the Internet was given to the Stanford Research Institute's Network Information Center (**SRI-NIC**).
- computer names and IP addresses were mapped using a **flat list** and stored in a simple centralized text file called the **hosts.txt file.**
- This file was maintained separately by each user on their own computer.
- The host file had only two columns: name and address. Every host could store the host file on its disk and update it periodically from a master host file.
- When a program or a user wanted to map a name to an address, the host consulted the host file and found the mapping.

Cont..

- Now Internet is not small, it is impossible to have only one host file to relate every address with a name and vice versa.
- **Problems**:

1. The file grew with the growth of the Internet. After some time, it became extremely difficult to maintain the file(The file getting bigger).

2. With the large number of computers connecting to the Internet, it became more impractical to guarantee the uniqueness of a host name.

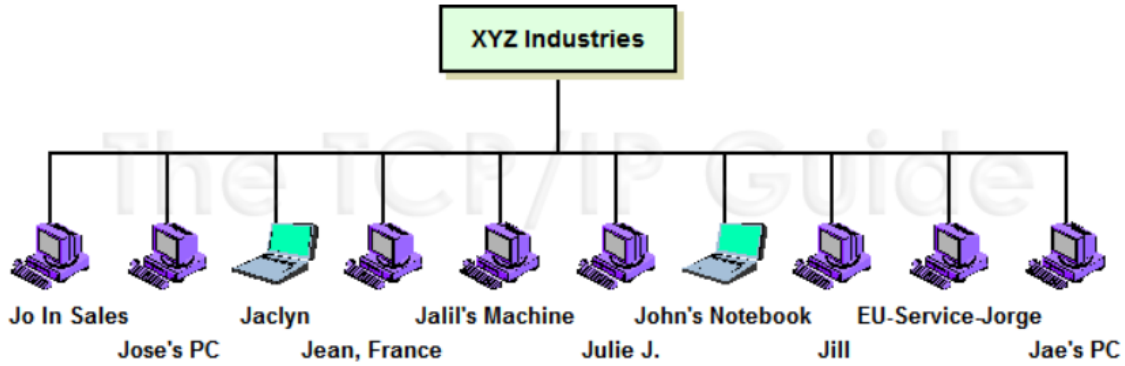
- These shortcomings created a need for things like **decentralized hierarchical naming structure and distributed management of host names**. This led to the **evolution of a lookup facility** known as **DNS**.
- One solution would be to **store the entire host file in a single computer and allow access to this centralized information to every computer that needs mapping.** But we know that this would create a huge amount of traffic on the Internet.
- Another solution, the one used today, is to **divide this huge amount of information into smaller parts and store each part on a different computer**. In this method, the host that needs mapping can contact the closest computer holding the needed information. This method is used by the Domain Name System (DNS).

Name space

- The names must be unique as the IP addresses are unique.
- A namespace that maps each address to a unique name.
- It can be organized in two ways: **Flat , Hierarchical**

Flat name spaces:

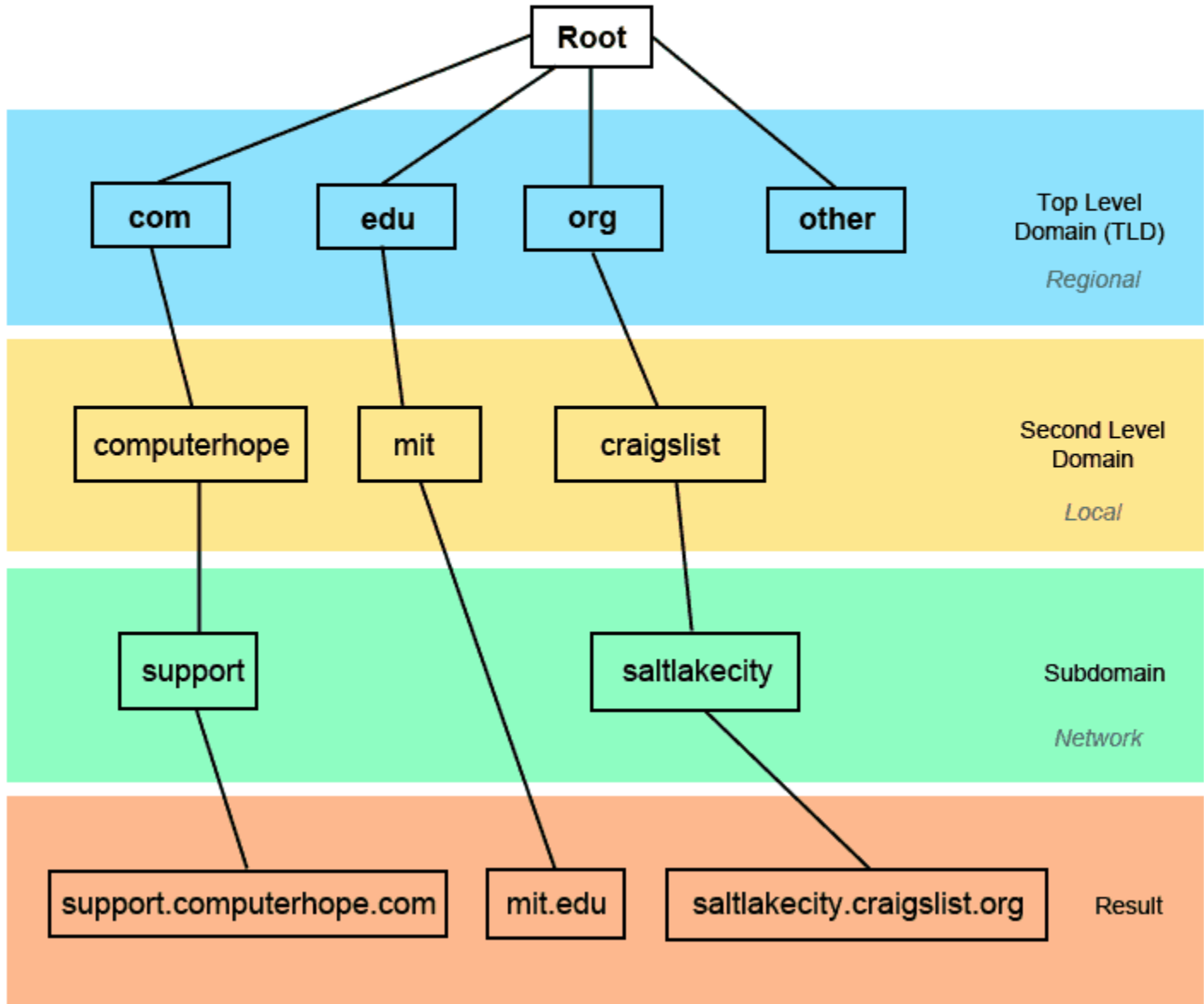
- In a flat name space, a name is a sequence of characters without structure.
- There is no clear relationship between any name and any other name.
- The main drawback of flat name space is that it cannot used in large system due to central control authority.



Hierarchical Name space:

- In a hierarchical name space, each name is made of several parts.
- The first part can define the **nature of the organization, the second part can define the name of an organization, the third part can define departments in the organization**, and so on.
- In this case, the authority to assign and control the name spaces can be decentralized.
- **E.g. unipune.ac.in** indicates university of pune is an academic institution and located in India.

Domain Naming Hierarchy

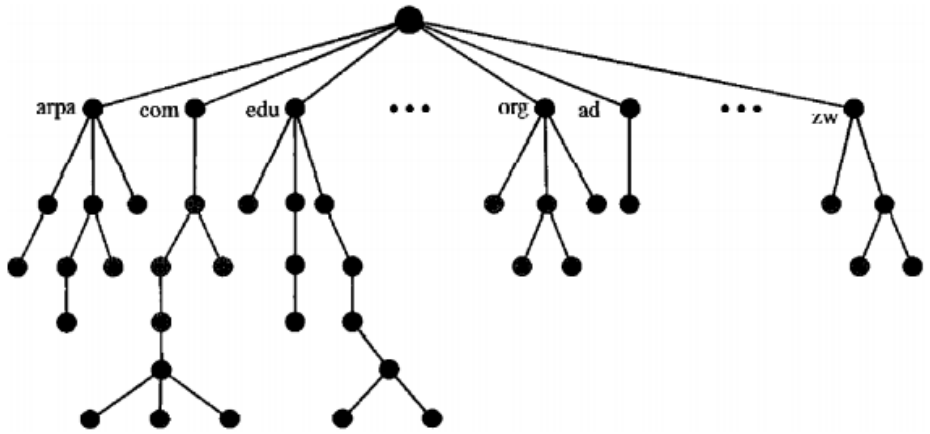


Domain namespace

- To have a hierarchical name space, a **domain name space** was designed.
- It is a name service provided by the internet for Transmission Control Protocol networks/Internet Protocol (TCP/IP).
- Each name consists of several parts. The tree can have only 128 levels 0(root) to level (127).
- The domain name space consists of a inverted tree structure. Each node or leaf in the tree has a ***label*** and **zero or more *resource records* (RR),** which hold information associated with the domain name.
- Authority for names in each partition is passed to each designated agent.

Label

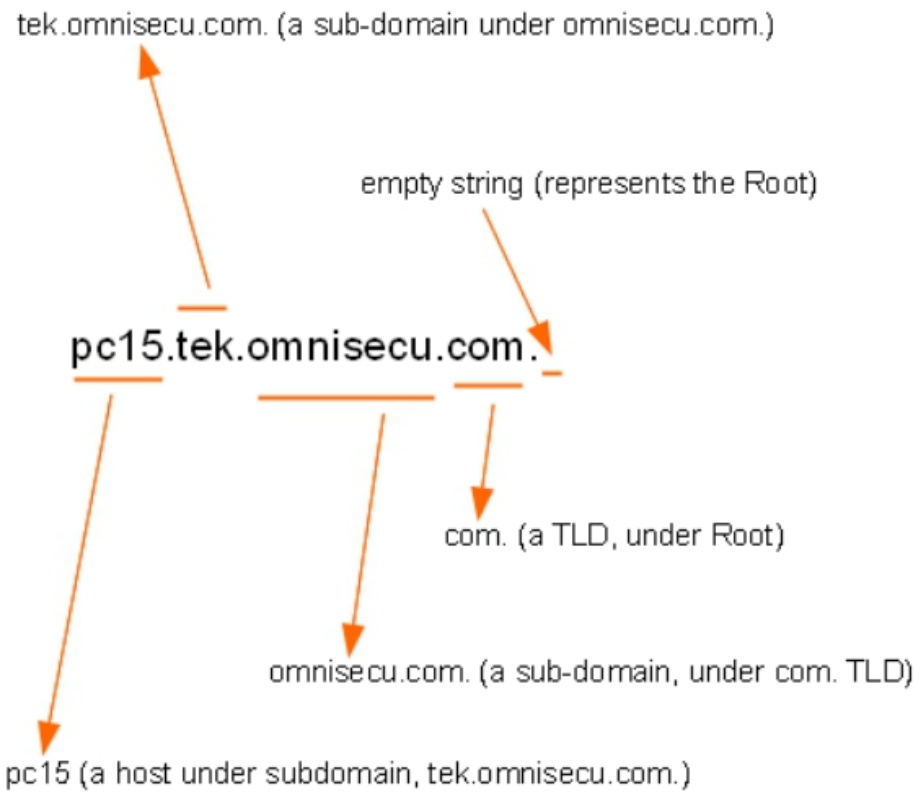
- Each node in the tree has a label, which is a string with a maximum of 63 characters.
- The root label is a null string (empty string).
- DNS requires that children of a node (nodes that branch from the same node) have different labels, which guarantees the uniqueness of the domain names.

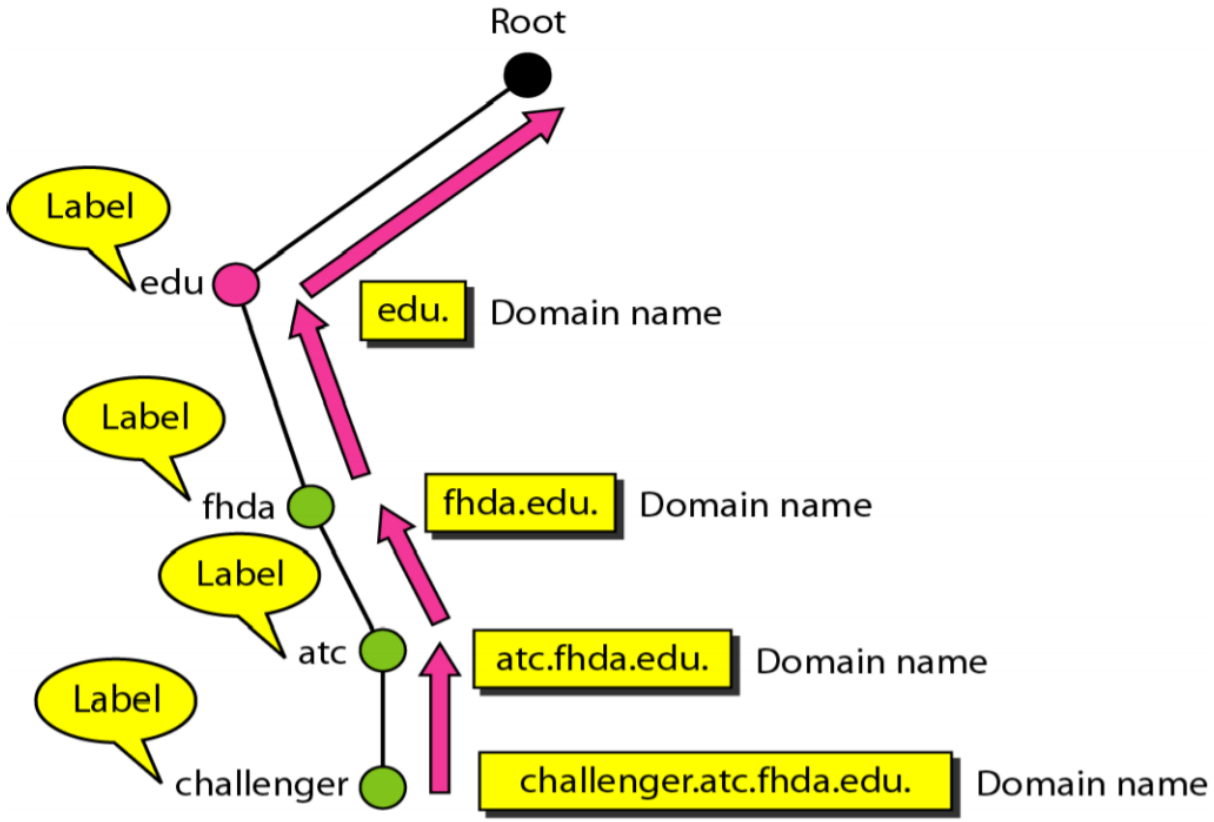


Domain Name

- Domain name is the address of your website that people type in the browser URL bar to visit your website.
- Domain name is a way to identify and locate computers connected to internet (IP address is difficult to remember).
- Each node in the tree has a domain name. A full domain name is a sequence of labels separated by dots (.). The domain names are always read from the node up to the root. The last label is the label of the root (null). This means that a full domain name always ends in a null label, which means the last character is a dot because the null string is nothing
- EXAMPLE: www.yahoo.co.in, www.facebook.com etc.

Domain Names and the Labels





Fully Qualified Domain Name (FQDN)

- A FQDN has the format:
  [hostname].[domain].[tld]. or
  [hostname].[subdomain].[domain].[tld].
- It is a domain name that specifies the **exact location** of a host (full name of a host) within the tree hierarchy of the Domain Name System (DNS).
- If a label is **terminated by a null/empty string** (which represents the Root), it is called a FQDN.
- It consists of all the labels **from the node, up to the root** of the namespace, separated by periods (".").
- FQDNs are also sometimes called ***absolute* domain names**.

Partially Qualified Domain Name (PQDN)

- If a label is not terminated by a null string, it is called a PQDN.
- PQDN is used to specify a portion of a domain name, normally the host portion of it. A PQDN starts with a host name, but it may not reach up to the root.
- It is used when the name to be resolved belongs to the same site as the client.
- It does not give the full path to the domain.

FQDN

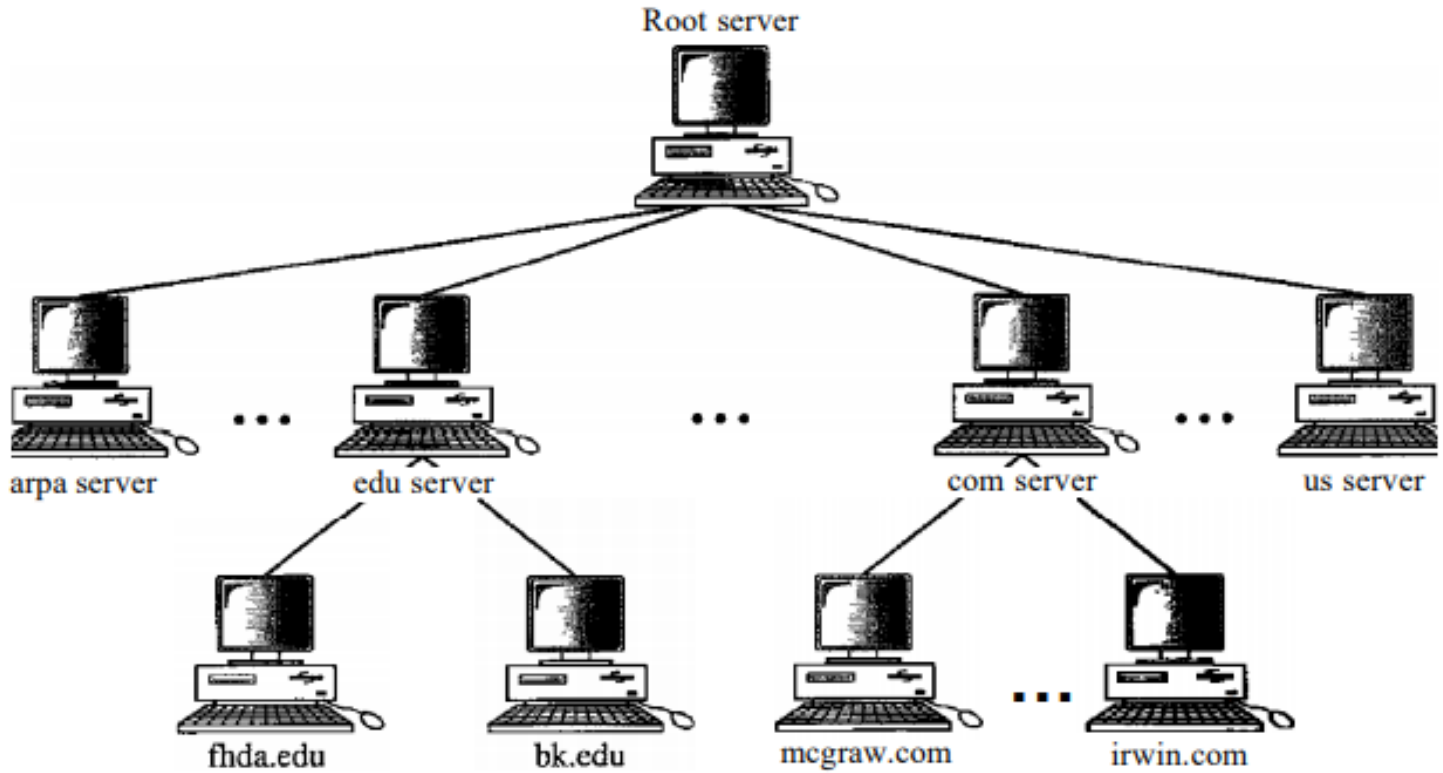
challenger.atc.fhda.edu.
cs.hmme.com.
www.funny.int.

PQDN

challenger.atc.fhda.edu
cs.hmme
www

Distribution of Name space

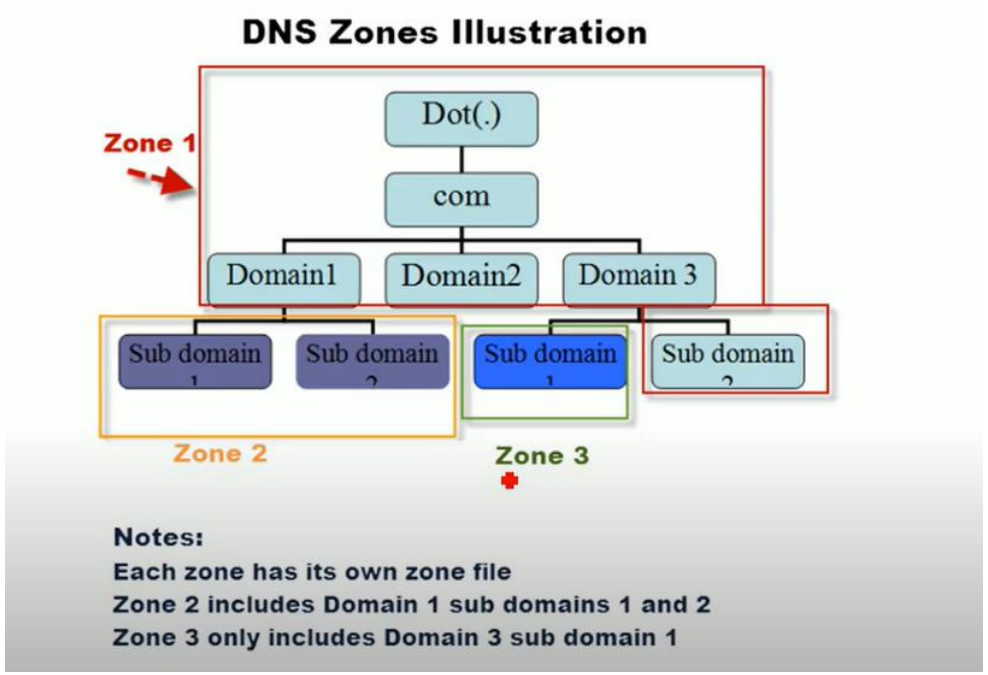
- The information contained in the domain name space must be stored.
- However, it is very inefficient and also unreliable to have just one computer store such a huge amount of information.
- It is inefficient because responding to requests from all over the world places a heavy load on the system.
- **DNS Server**: The solution is to the above problem is to distribute this information among several computers across world. These computer are called DNS Server.
- Each server is responsible for storing a domain or a subdomain.

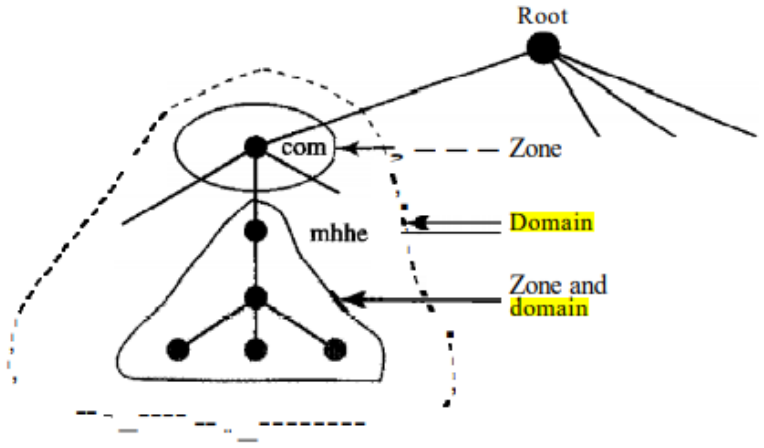

Root server
arpa server
edu server
com server
us server
fhda.edu
bk.edu
mcgraw.com
irwin.com

Zone

- A server is responsible for or has authority over called a zone.
- Zone is a contiguous part of the tree.
- The server makes a database called a **zone file** and keeps all the information for every node under that domain.
- A DNS zone is a **part of the DNS namespace that is administered by a specific organization or person**.
- If a server accepts responsibility for a domain and does not divide the domain into smaller domains, the domain and the zone refer to the same thing.
- However, if a server divides its domain into subdomains and delegates part of its authority to other servers, domain and zone refer to different things.

- The information about the nodes in the subdomains is stored in the servers at the lower levels, with the original server keeping some sort of reference to these lower-level servers.
- Types of DNS Zone:
  - DNS root zone
  - Forward DNS zone and Reverse DNS zone
  - Primary DNS zone and secondary DNS zone





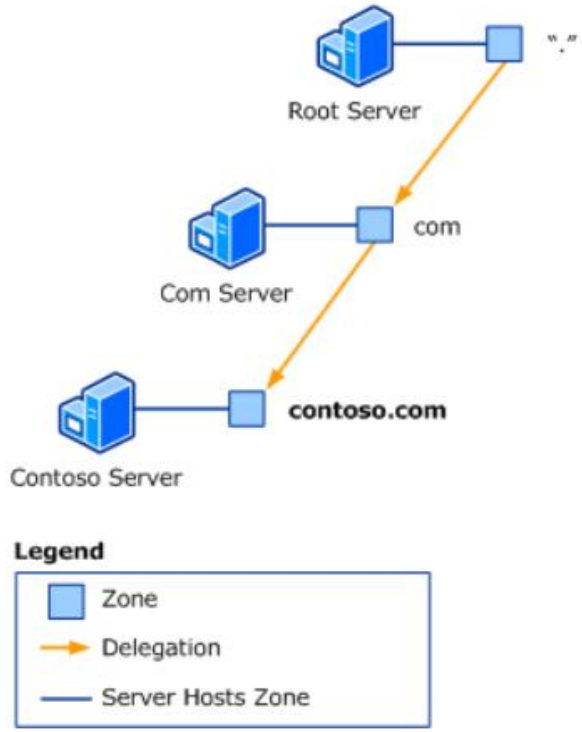
Types of Servers

- **Root Server:** A root server is a server whose zone consists of the whole tree. It usually does not store any information about domains but delegates its authority to other servers, keeping references to those servers.
- **Primary Servers:** It is a server that stores a file about the zone for which it is an authority. It is responsible for creating, maintaining, and updating the zone file. It stores the zone file on a local disk.
- **Secondary server:** It is a server that transfers the complete information about a zone from another server (primary or secondary) and stores the file on its local disk. It neither creates nor updates the zone files.
  If updating is required, it must be done by the primary server, which sends the updated version to the secondary.

Note : Zone transfer *is the process of copying the contents of the zone file on a primary DNS server to a secondary DNS server.*

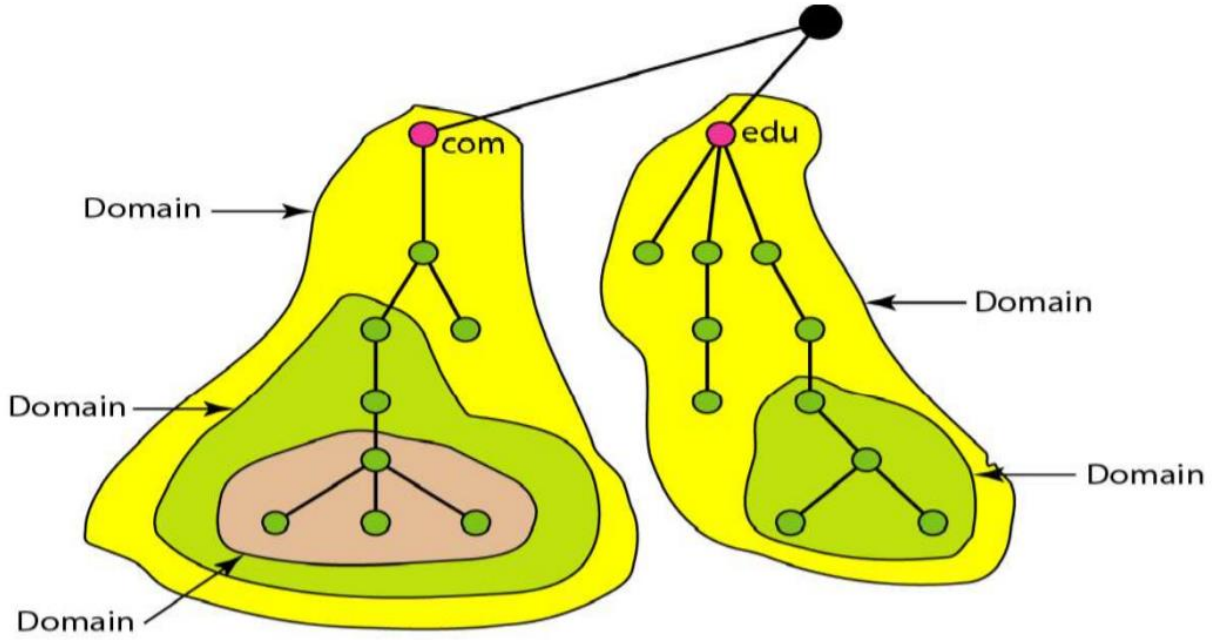
DNS Delegation

- For a DNS server to answer queries about any name, it must have a **direct or indirect path to every zone in the namespace**. These paths are created by means of delegation.
- A delegation is a record in a parent zone that lists a name server that is authoritative for the zone in the next level of the hierarchy.
  Delegations makes it possible for servers in one zone to refer clients to servers in other zones.



Domain

- A domain is a subtree of the domain name space. The name of the domain is the domain name of the node at the top of the subtree.
- Note that a domain may itself be divided into domains (or **subdomains** as they are sometimes called).



Top Level Domain (TLD)

A **top-level domain** (**TLD**) is one of the **domains** at the **highest level** in the hierarchical **Domain** Name System of the Internet. The **top-level domain** names are installed in the root zone of the name space

Top level domains are classified into 3 categories:

- Organizational or generic domains
- Geographical or country domains
- Inverse domains

Organizational or Generic Domains

Indicates primary function of the organization or their generic behavior.

Label	Description
aero	Airlines and aerospace companies
biz	Businesses or firms
com	Commercial Organizations
coop	Cooperative business Organizations
edu	Educational institutions
gov	Government institutions
info	Information service providers
int	International Organizations
mil	Military groups
museum	Museum & other nonprofit organizations
name	Personal names
net	Network Support centers
org	Nonprofit Organizations
pro	Professional individual Organizations

Geographical or Country Domains

- It consists of two characters which represents different countries/regions all around the world
- These codes have been standardized by International Standard Organization (ISO)

EXAMPLE:

- .in India
- .jp Japan
- .us United States
- .fr France
- .it Italy
- .cn China
- .au Australia

1.	.de	(Germany)	6.	.ru	(Russian Federation)
2.	.uk	(United Kingdom)	7.	.br	(Brazil)
3.	.nl	(Netherlands)	8.	.ar	(Argentina)
4.	.eu	(European Union)	9.	.it	(Italy)
5.	.cn	(China)	10.	.pl	(Poland)

Inverse Domains

- The inverse domain is used to map an address to a name.
- E.g. If a server receives a request from a client and the server has only the IP addresses of the clients in its list then the server asks its resolver(DNS Client) to query to the DNS server to map the IP address to name to verify if the client is authorized.
- This type of query is called an **inverse or pointer (PTR) query**.

Sub Domain

A subdomain is an additional part to your main domain name. Subdomains are created to organize and navigate to different sections of your website. You can create multiple subdomains or child domains on your main domain.

For example:

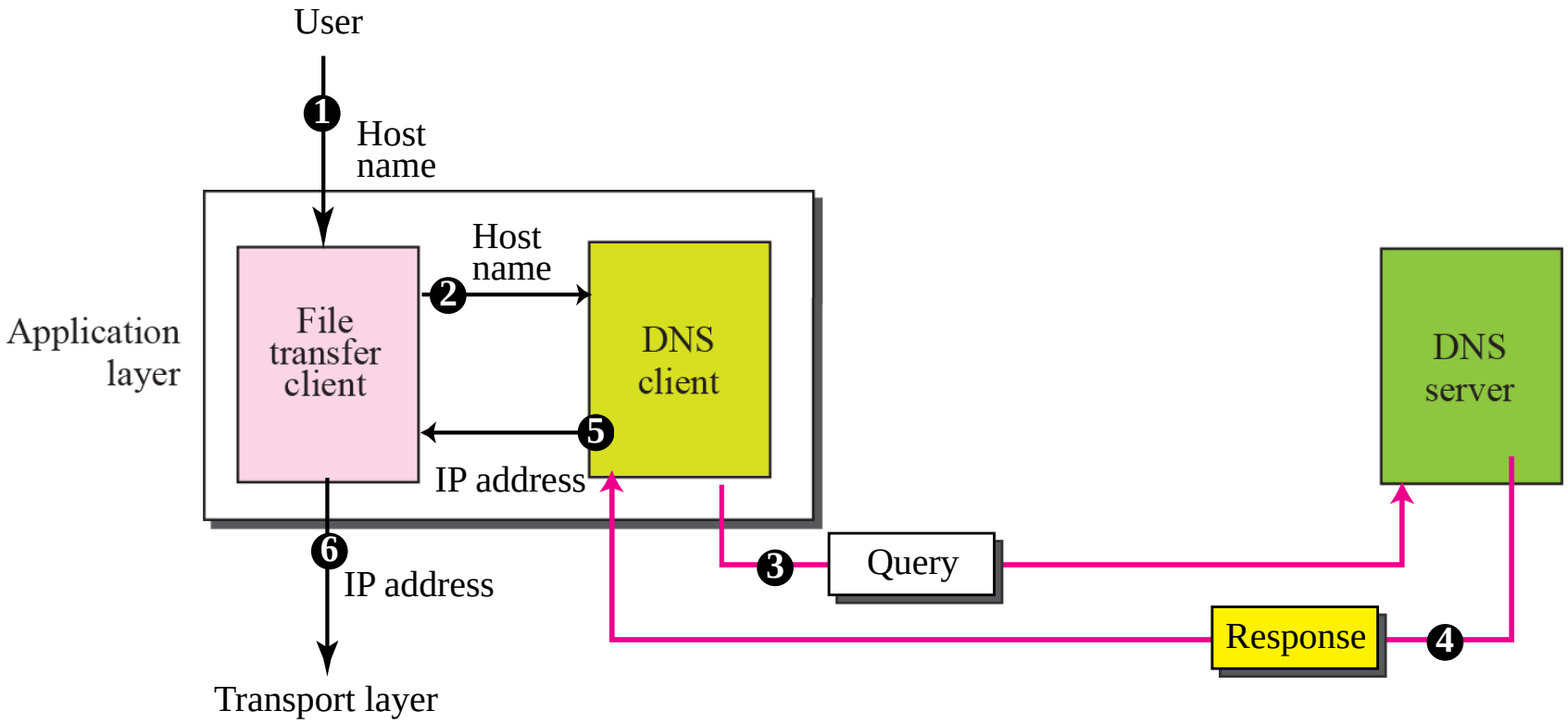
store.yourwebsite.com

In this example, ‘store’ is the subdomain, ‘yourwebsite’ is the primary domain and ‘.com’ is the top level domain (TLD).

Domain Name System (DNS)

- **Domain Name System** lets you connect to websites by matching human-readable **domain names** (like wpbeginner.com) with the unique ID of the server where a website is stored.
- The Domain Name System (DNS) is the **phonebook** of the Internet. Humans access information online through domain names, like nytimes.com or espn.com.
- Web browsers interact through Internet Protocol (IP) addresses. DNS translates domain names to IP addresses so browsers can load Internet resources.

Purpose of DNS



DNS Components

- **DOMAIN NAME SPACE and RESOURCE RECORDS(RR)** - Specifications for a tree structured name space and data associated with the names. RR defines the domain names within the Domain Name Space.
- **CLIENT or RESOLVER -** The client in the DNS contains functions or software routines, which request information from the Domain Name Space on behalf of an application. The clients in the DNS are also known as resolvers because they group functions to a resolver library. Programs that extract information from name servers in response of client requests.
- **Server/ Name Server -** Server programs which hold information about the domain tree's structure. These servers manage portions of the Domain Name Space and assist clients in finding information within the DNS tree. In addition, servers can be used as a delegation point to identify other name servers that have authority over subdomains within a given domain.

DNS Servers

- A server is a device or program dedicated to providing services to other programs, referred to as ‘clients’.
- Four types:

Recursive Resolvers, Root Name servers, TLD Name server, Authoritative servers.

- The **DNS stub resolver** is a component of the DNS that is accessed by application programs when using the DNS for e.g. resolving domain names to IP addresses. The stub resolver simply serves as an intermediary between the application requiring DNS resolution, and a recursive DNS resolver.

Types of DNS servers

1. **Recursive Resolver -** A Recursive resolver(DNS Recursor), is designed to receive DNS queries, which include a human-readable hostname such as “www.example.com”, and is responsible for tracking the IP address for that hostname. First stop in a DNS query. Acts as a middleman between Client and DNS nameserver.
2. **Root Name Server -** The root server is the first step in the journey from hostname to IP address. The DNS Root Server extracts the Top Level Domain (TLD) from the user’s query
   — for example, **www.example.com**
   — It provides details for the .com TLD Name Server.

3. TLD Name servers- Servers that maintain information for all domain names that share a common domain extention such as .com, .net.

4. Authoritative Name Server - Higher level servers in the DNS hierarchy define which DNS server is the “authoritative” name server for a specific hostname(holds the up-to-date information for that hostname.)
The last stop in the name server query.
It takes the hostname and returns the correct IP address to the DNS Resolver (or if it cannot find the domain, returns the message NXDOMAIN).

Note: *An NXDOMAIN error message means that the domain does not exist.(Received by the DNS resolver when a request to resolve a domain is sent to the DNS and cannot be resolved to an IP address.)*

Services Offered By a DNS Server

- Provides a mechanism for resolving host names into Internet Protocol (IP) addresses. This process is known as **forward lookup**.
- Provides a mechanism to find host names from IP addresses. This is known as a **reverse lookup**.
- Provides Internet directory-like lookup capabilities to retrieve information concerning other DNS Name Servers, Canonical Names, Mail Exchangers, and so on.
- Lead to efficient searching.
- Organizes named machines into efficient hierarchies.
- Is distributed by nature, which increases its robustness and reliability.

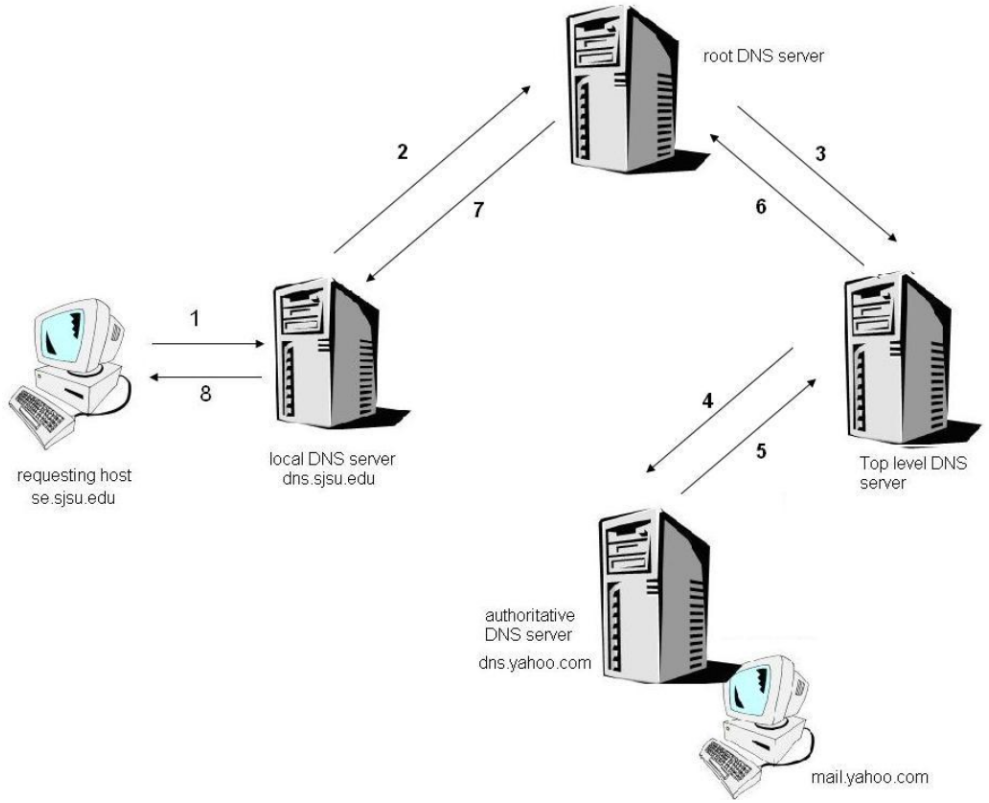
DNS Queries

- DNS queries are the computer code that tells the DNS servers what kind of query it is and what information it wants back. A DNS query (also known as a DNS request) is a demand for information sent from a user's computer (DNS client) to a DNS server.

1. Recursive query
2. Iterative query
3. Non-recursive query: A query in which the DNS Resolver already knows the answer. It either immediately returns a DNS record because it already stores it in local cache, or queries a DNS Name Server which is authoritative for the record, meaning it definitely holds the correct IP for that hostname.

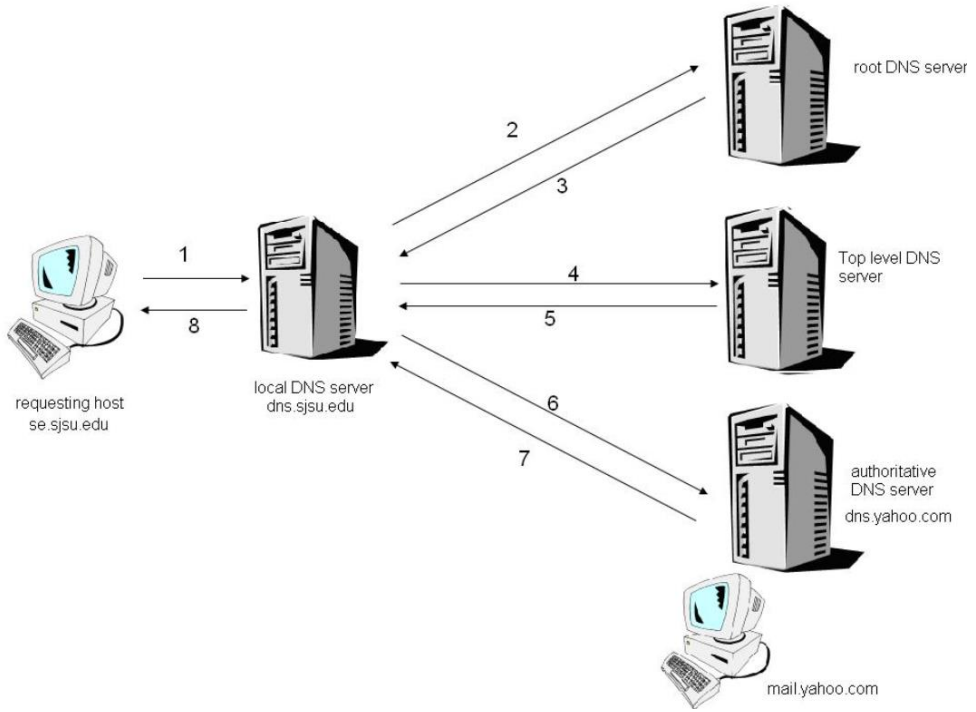
Recursive Queries

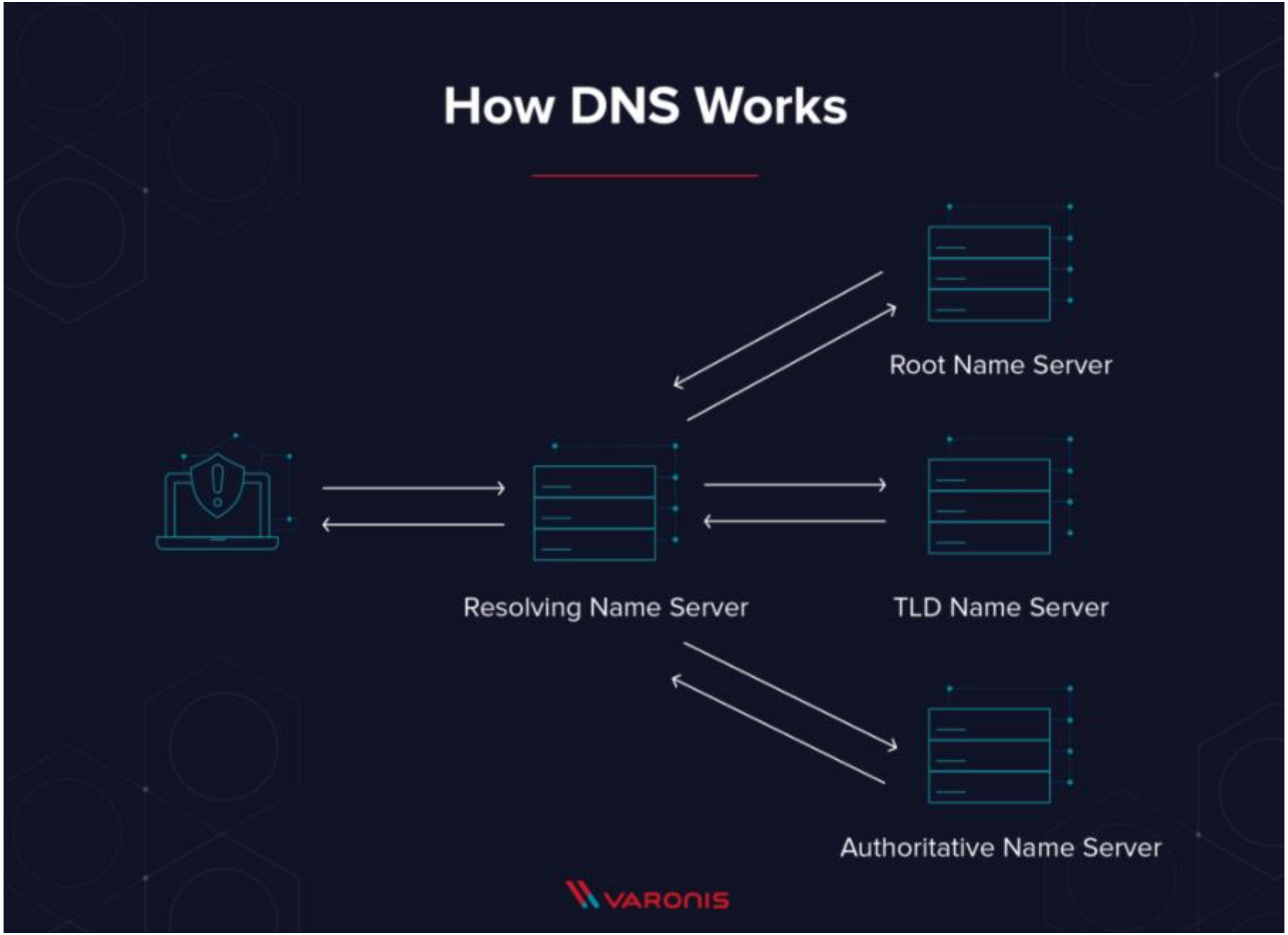
The computer requests an IP address or the confirmation that the DNS server doesn't know that IP address. When a DNS client directly gets the IP address of a domain, by asking the name server system to perform the complete translation.



Iterative Queries

The requester asks a DNS server for the best answer it has. If the DNS server doesn't have the IP address, it will return the authoritative name server or TLD name server (referral). The requester will continue this iterative process until it finds an answer or times out.(when a DNS client contacts the name servers, one by one, until it finds the server, containing the needed information)




How DNS Works
Root Name Server
Resolving Name Server
TLD Name Server
Authoritative Name Server
VARONIS

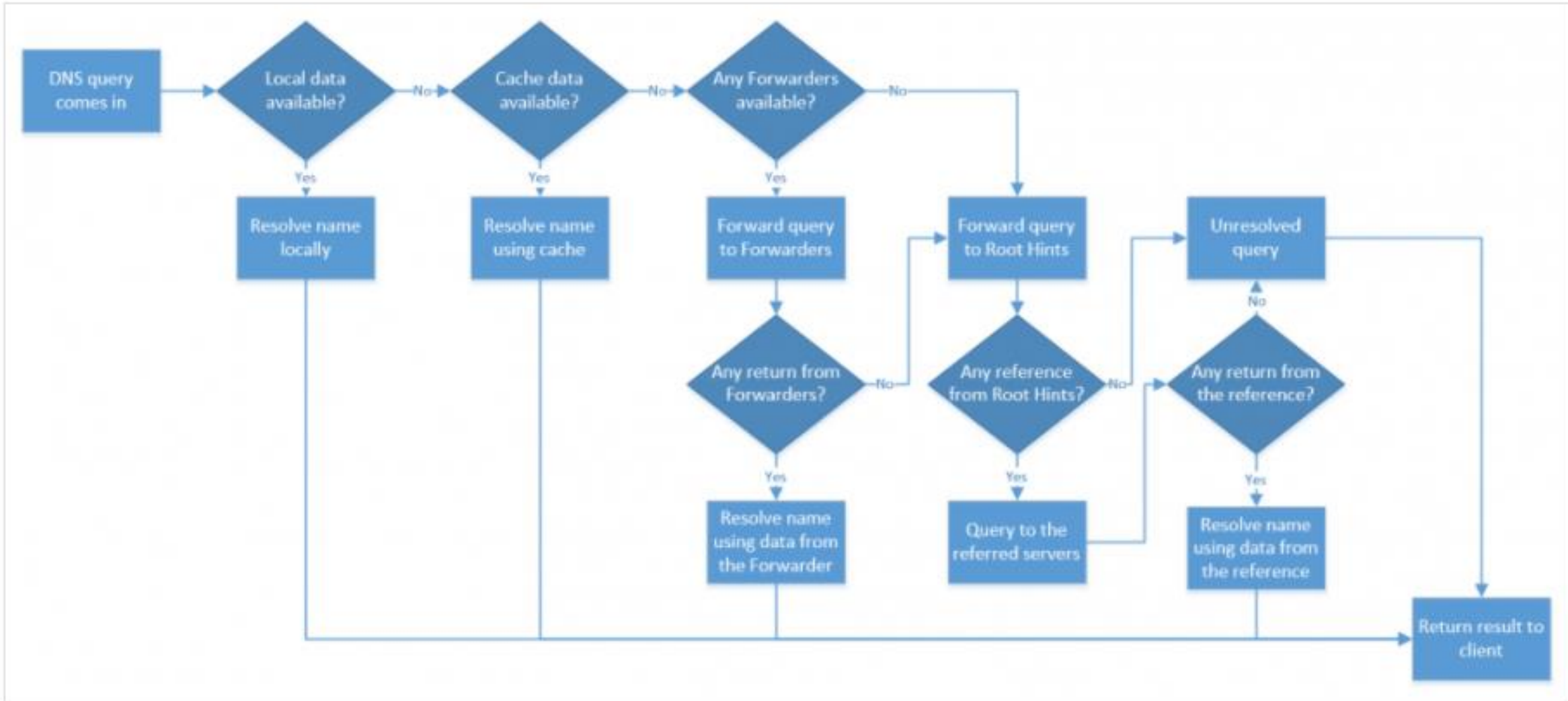
RESOLUTION

- Mapping a name to an address or an address to a name is called **name-address resolution**.
- **Resolver:**
  - DNS is designed as a client/server application.
  - A host that needs to map an address to a name or a name to an address calls a DNS client called a **resolver.**
  - It accesses the closest DNS server with a mapping request. If the server has the information, it satisfies the resolver; otherwise, it either refers the resolver to other servers or asks other servers to provide the information.
  - After the resolver receives the mapping, it interprets the response to see if it is a real resolution or an error, and finally delivers the result to the process that requested it.

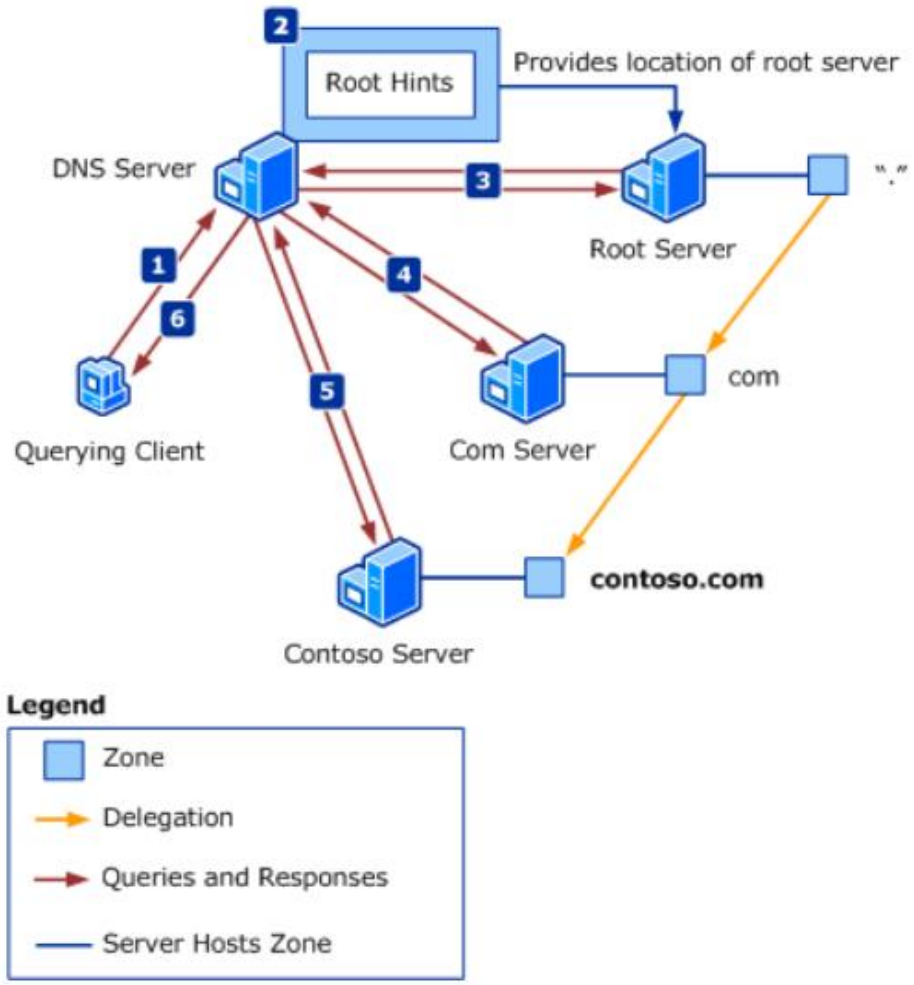
DNS Forwarders and Root Hints

- Two ways to direct DNS queries out of your organization**:**
  Root hints and DNS forwarders.

  - **Root hints** : **List of authoritative name servers for the root DNS names in the internet**. When a **DNS** server cannot resolve a name query by using its local data, it uses its **root hints** to send the query to a **DNS** server .(pointers to servers).(Iterative Queries)
  - A **forwarder :** A feature in DNS server that is used to forward DNS queries for external DNS names to DNS servers outside of that network(cannot resolve locally to that DNS server).
    It is an **another server in the local network or external network**.


DNS query comes in
Local data available?
No
Cache data available?
No
Any Forwarders available?
No
Yes
Resolve name locally
Yes
Resolve name using cache
Yes
Forward query to Forwarders
Forward query to Root Hints
Unresolved query
No
Any return from Forwarders?
No
Any reference from Root Hints?
No
Any return from the reference?
Yes
Resolve name using data from the Forwarder
Yes
Query to the referred servers
Yes
Resolve name using data from the reference
Return result to client

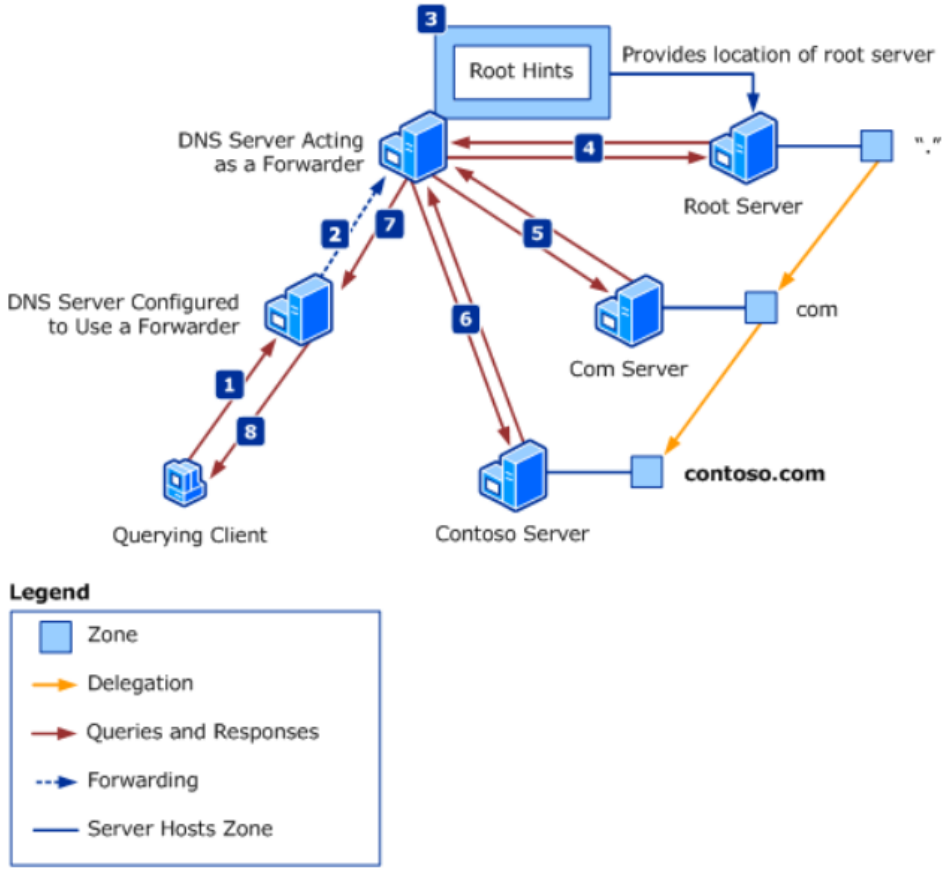
Resolving names by using root hints (Iteration)



Steps….

- A client sends a recursive query to a DNS server to request the IP address that corresponds to the name **ftp.contoso.com.** The response to the recursive query must be a valid address or a message indicating that the address cannot be found.
- If the DNS server is not authoritative for the name and does not have the answer in its cache, the DNS server uses **root hints** to find the IP address of the **DNS root server**.
- The DNS server uses an iterative query to ask the DNS root server to resolve the name ftp.contoso.com. An iterative query indicates that the server will accept a referral to another server in place of a definitive answer to the query. **DNS root server** returns a referral to the **Com server** that hosts the com zone.
- The DNS server uses an iterative query to ask the Com server to resolve the name ftp.contoso.com. Because the name ftp.contoso.com ends with the name contoso.com, the Com server returns a referral to the **Contoso server** that hosts the contoso.com zone.
- The DNS server uses an iterative query to ask the Contoso server to resolve the name ftp.contoso.com. The Contoso server finds the answer in its zone data and then returns the answer to the server.
- The server then returns the result to the client.

Resolving names by using forwarding (Recursion)



Steps…

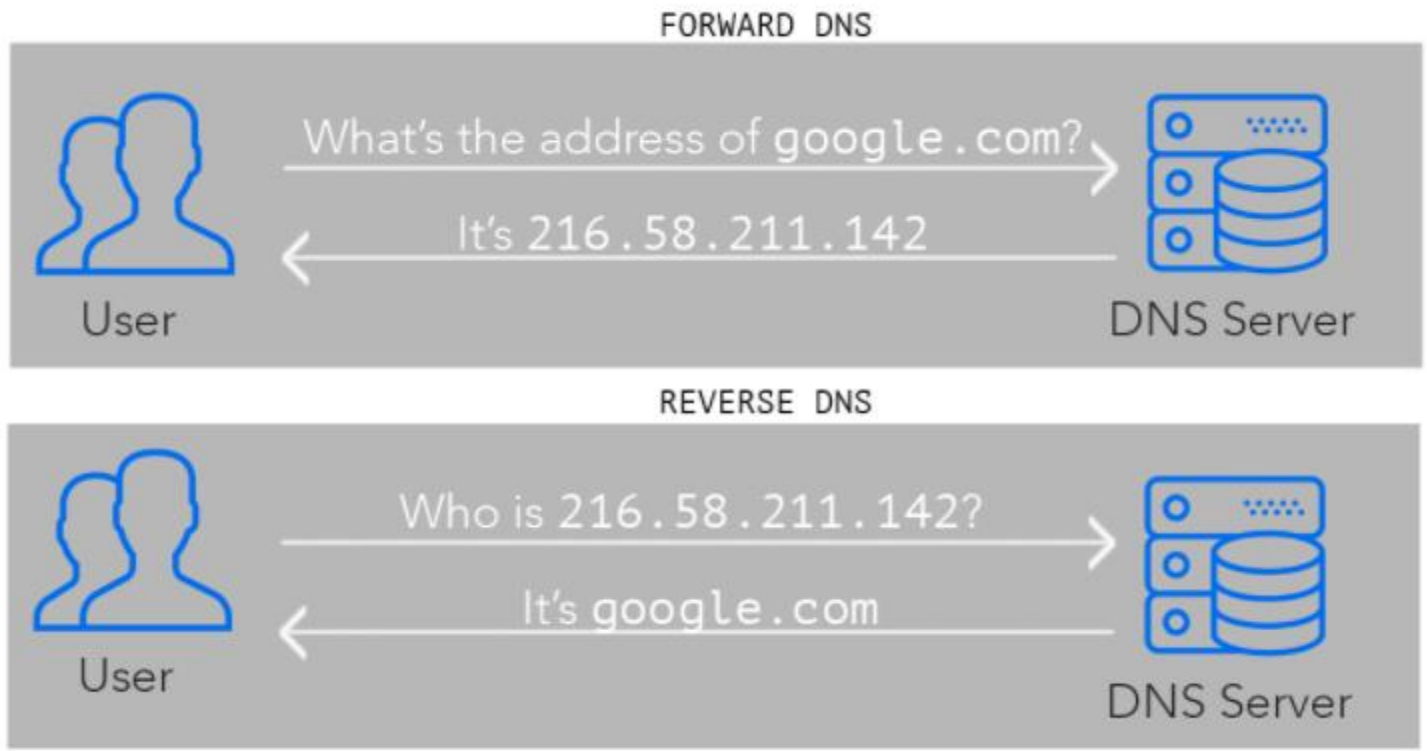
- A client queries a DNS server for the name ftp.contoso.com.
- The DNS server forwards the query to another DNS server, known as a **forwarder**.
- Because the forwarder is not authoritative for the name and does not have the answer in its cache, it uses **root hints** to find the IP address of the DNS root server.
- The **forwarder** uses an iterative query to ask the **DNS root server** to resolve the name ftp.contoso.com. Because the name ftp.contoso.com ends with the name com, the DNS root server returns a referral to the Com server that hosts the com zone.
- The **forwarder** uses an iterative query to ask the **Com server** to resolve the name ftp.contoso.com. Because the name ftp.contoso.com ends with the name contoso.com, the Com server returns a referral to the Contoso server that hosts the contoso.com zone.
- The **forwarder** uses an iterative query to ask the **Contoso server** to resolve the name ftp.contoso.com. The Contoso server finds the answer in its zone files, and then returns the answer to the server.
- The forwarder then returns the result to the original DNS server.
- The original DNS server then returns the result to the client.

DNS Lookup

- A DNS lookup is the process of querying technique of the Domain Name System (DNS) .
- This is like looking up a phone number in a phone book - that is why it is referred to as a "lookup".
- Two types:

    Forward DNS Lookup

    Reverse DNS Lookup


FORWARD DNS
What's the address of google.com?
It's 216.58.211.142
User
DNS Server
REVERSE DNS
Who is 216.58.211.142?
It's google.com
User
DNS Server

Forward DNS Lookup

- Forward DNS lookup is using an Internet domain name to find an Ip address.
- A DNS server is said to resolve a domain name when it returns its IP address. A forward DNS request is the opposite of a reverse DNS lookup.
- A most common lookup since most users think in terms of domain names rather than IP addresses.

Reverse DNS Lookup

- Reverse DNS lookup is using an Internet IP address to find a domain name.
- In computer networks, a **reverse DNS lookup** or **reverse DNS** resolution (**rDNS**) is the querying technique of the Domain Name System (**DNS**) to determine the domain name associated with an IP address .
- In this case DNS uses the inverse domain.

DNS Protocol

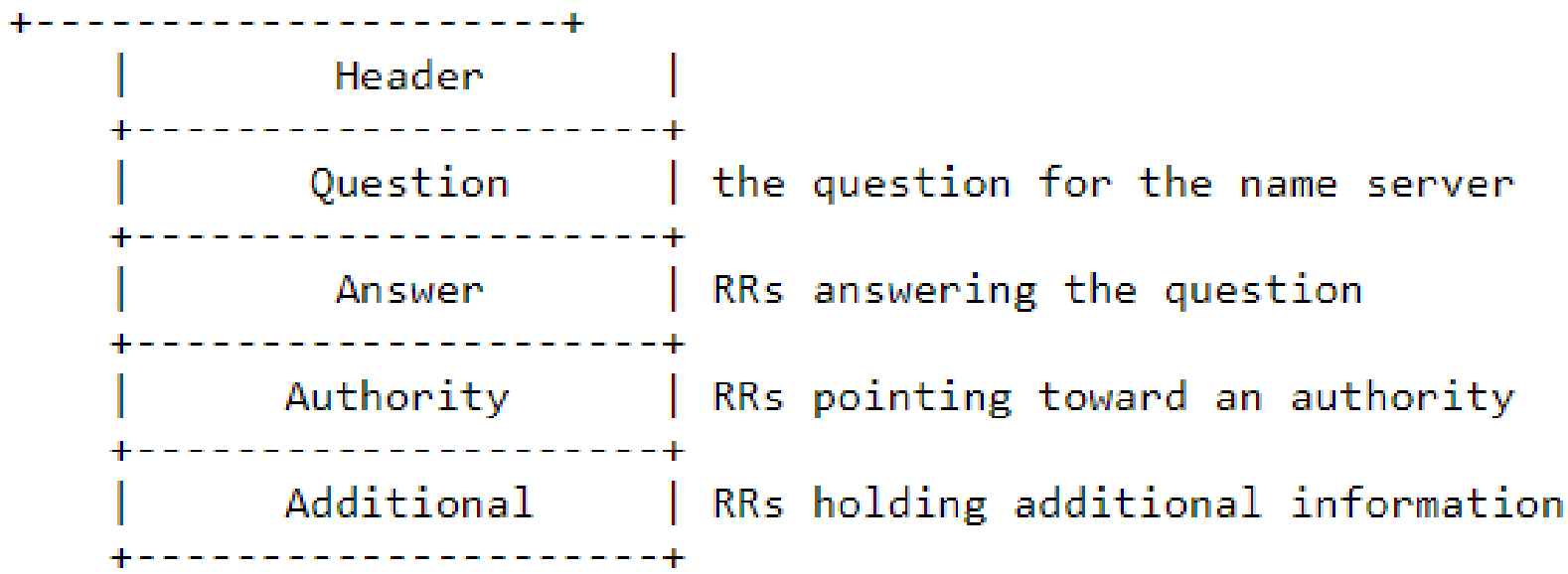
The Domain Network System (DNS) protocol helps Internet users and network devices discover websites using human-readable hostnames, instead of numeric IP addresses.

Two types of DNS messages, **queries and replies**.

DNS uses UDP for message smaller than 512 bytes (common requests and responses).
DNS uses TCP for bigger exchange (i.e. zone transfer).

- **Header section -** Information about the **type of message** and other sections that are present in the message.
- **Question section -** Information concerning the **object of the query**.
- **Answer section** - **RRs** regarding the answer.
- **Authority section - Statement of Authority** (SOA) or **name server records**.
  These name server records belong to the zone of authority for the owner name of the RR(s) in the Answer section.
- **Additional section -** Additional information for the receiver.

DNS Message structure



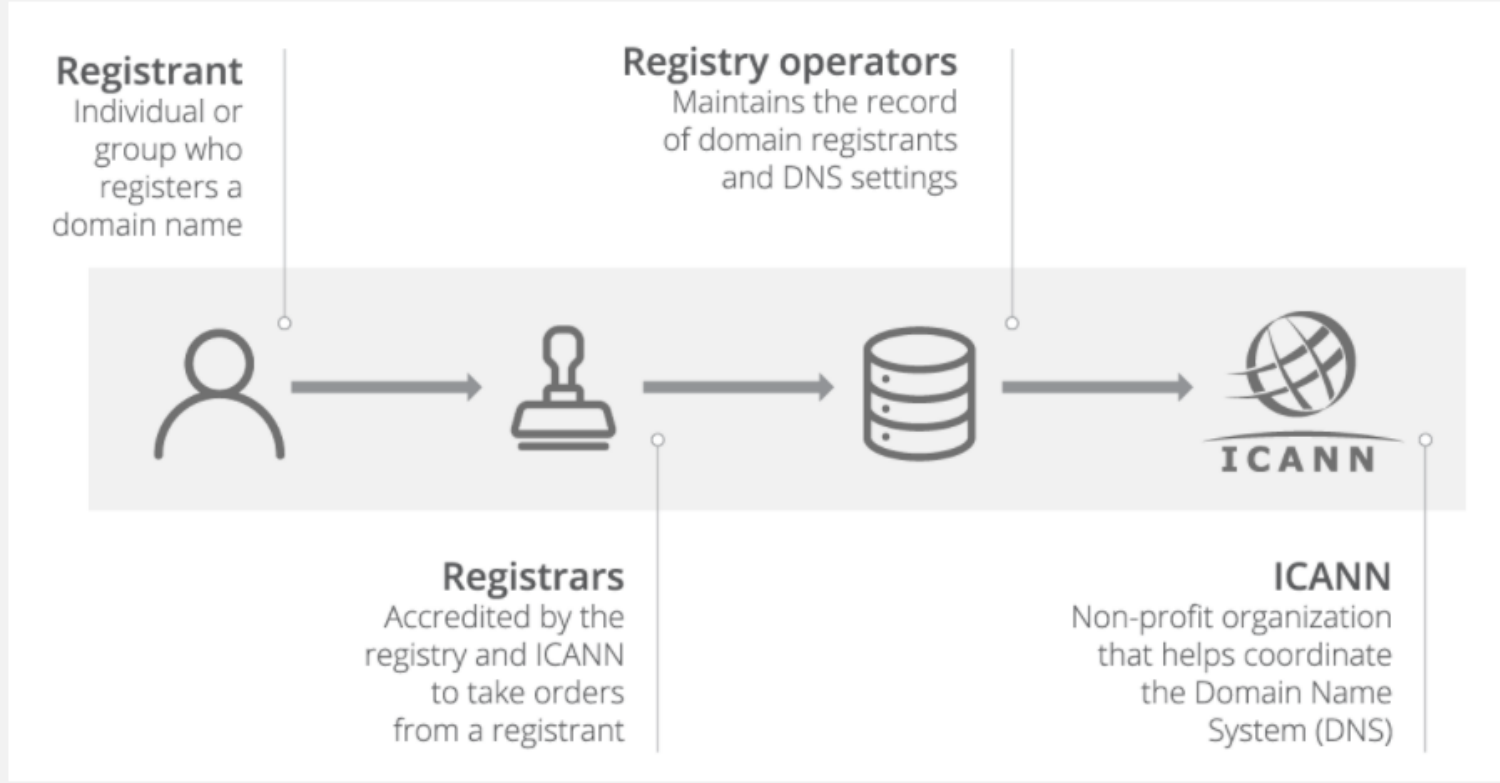
Domain name registrars

- A **domain name registrar** is a business that handles the reservation of domain names as well as the assignment of Ip addresses for those domain names.
- It should be noted that registrars don't actually manage and maintain domain names; that part is done by a domain name registry.
- **Registries** are organizations that manage top-level domains (TLDs) '.com' and '.net' — specifically by maintaining the records of which individual domains belong to which people and organizations.
- The registry information about a domain name and the designated registrar can be looked up at many services such as ICANN's WHOIS.

Note: ICANN-Internet Corporation for Assigned Names and Numbers

Cont...

- A registrar first verifies that the requested domain name is unique and then enters it into the DNS database.
- A fee is charged.
- Eg: GoDaddy
  bluehost
  HostGator
  LK Domain Registry



Domain name records

Two types of records:

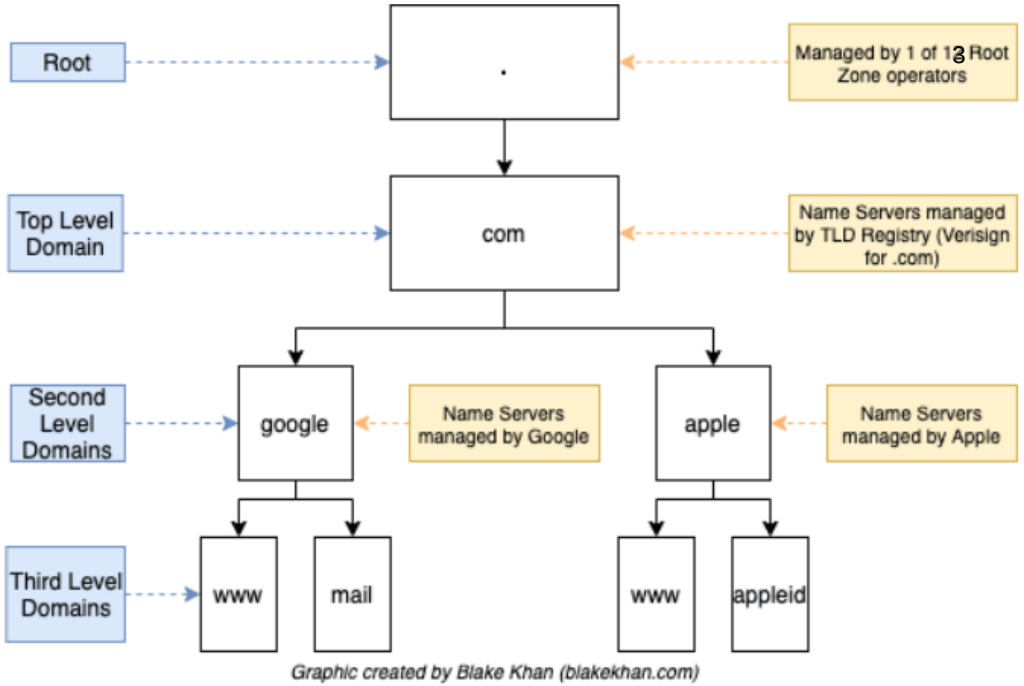
1. **Question Record** : A question record is used by the client to get information from a server. This contains the domain name. They are used in the **question section** of the query and response messages.
2. **Resource Record**: A **resource record(RR)** is the unit of information entry in **DNS** zone files;
   RRs are the basic building blocks of host-name and IP information and are used to resolve all **DNS** queries

Most commonly used DNS record types:

- **A record** holds the **IPv4 address** of a domain.
- **AAAA record** is essentially the same as A record but for **IPv6 addresses**.
- **PTR record** finds a domain name in a **reverse-lookup** when the IP is already known.
- **CNAME record**, or canonical name, forward a domain or sub-domain to another domain without providing an IP address. These can be used as **aliases to domains**.
- **MX record** is the mail exchange record that directs **mail to an email server**. It indicates how email should be routed to its destination.
- **TXT record** lets a domain administrator store text information for sources outside of your domain. These are commonly used to **gauge the trustworthiness and verify ownership of a domain**.
- **NS record** indicates the **authoritative name servers**. A domain often has multiple name servers, primary and secondary, to prevent outages in case of failures.

DNS is Decentralized

- DNS is decentralized in terms of not a single party is responsible for providing the nameservers at each level.(Delegate authority)



DYNAMIC DOMAIN NAME SYSTEM (DDNS)

- When the DNS was designed, no one predicted that there would be so many address changes.
- In DNS, when there is a change, such as adding a new host, removing a host, or changing an IP address, the change must be made to the DNS master file.
- These types of changes involve a lot of manual updating.
- The size of today's Internet does not allow for this kind of manual operation. The DNS master file must be updated dynamically.
- The Dynamic Domain Name System (DDNS) therefore was devised to respond to this need.
- **Dynamic DNS** (**DDNS**) is a method of automatically updating a name server in the Domain name system(DNS), often in real time, with the active DDNS configuration of its configured hostnames, addresses or other information.

Cont…

- In DDNS, when a binding between a name and an address is determined, the information is sent, usually by DHCP to a primary DNS server.
- The primary server updates the zone. The secondary servers are notified either actively or passively.
- In active notification, the primary server sends a message to the secondary servers about the change in the zone, whereas in passive notification, the secondary servers periodically check for any changes.
- In either case, after being notified about the change, the secondary requests information about the entire zone (zone transfer).
- DDNS systems are used to **update traditional DNS records without manual editing and permits lightweight and immediate updates.**
- To provide security and prevent unauthorized changes in the DNS records, DDNS can use an **authentication mechanism**.

Who is Responsible for Domain Name System?

- **Internet Corporation for Assigned Names and Numbers** (ICANN) manages the domain names system. It is a non-profit organization that creates and implements the policies for domain names.
- ICANN gives permission to companies called **Domain Name Registrars** for selling domain names. They are allowed to make changes to domain names registry on your behalf.
- **Domain name registrars** can sell domain names, manage its records, renewals, and transfers to other registrars.

Public DNS and Private DNS

- **Public DNS:** For a server to be accessible on the public internet, it needs a **public DNS record**, and its IP address needs to be reachable on the internet. It is accessible to anyone that can connect to them and don’t require authentication.
- **Private/ Internal DNS**: Computers that live **behind a firewall or on an internal network** use a private DNS record so that local computers can identify them by name. Outside users on the internet will not have direct access to those computers.

DNS Cache

- DNS cache is a **repository of domain names and IP addresses** that are stored on a computer, so it doesn’t have to ask for the IP address every time.
- Each time a server receives a query for a name that is not in its domain, it needs to search its database for a server IP address. Reduction of this search time would increase efficiency. DNS handles this with a mechanism called caching.
- It refers to the temporary storage of information about previous **DNS** lookups on a machine's OS or web browser.
- DNS clients and DNS server both use caching to speed up the **domain name lookup process** and to ease traffic on the root servers. DNS caching has two major goals:

    1. Speed up DNS requests
    2. Reduce bandwidth of DNS requests across the internet

Cont....

- However to inform the client that the response is coming from the cache memory and not from an authoritative source, the server marks the **response as unauthoritative**.
- Caching speeds up resolution, but it can also be problematic. If a server caches a mapping for a long time, it may send an outdated mapping to the client.
- To counter this, **Time To Live[TTL]** is used. [The time for which the DNS data is considered as up-to-date]
- It defines the time in seconds that the receiving server can cache the information.
- After that time, the mapping is invalid and any query must be sent again to the authoritative server.

Cont..

- If a client queries **domain server A** looking to resolve **www.mydomain.com,** and in turn **domain server A** queries **domain server B** etc then the result will be stored in a cache on

  the client (windows only)
  domain server A
  domain server B

- If another client needs to resolve the same domain name using server A then server A can respond using the **cached result**.
- You can check the DNS cache on a Windows machine with the command:

  ipconfig /displaydns

Cont…

There are a few different types of DNS caching used on the internet:

- **Browser DNS caching:** Current browsers have built in DNS caching functionality. Fast and efficient.
- **Operating System (OS) DNS caching:** Your computer is a DNS client, and there is a service on your computer that manages DNS resolution and requests. Fast and requires no bandwidth.
- **Recursive resolving DNS caching:** Each DNS recursor has a DNS cache, and it stores any IP address that it knows to use for the next request.

What Is Smart DNS

- Smart DNS is an online service that combines DNS and a proxy server.
- It substitutes your original DNS address (that of your ISP) with the one provided to you by a proxy server.
- The location of that proxy server may vary depending on your DNS provider’s infrastructure, but usually, this doesn’t limit your ability to access blocked websites.
- However, that’s only true if those websites were blocked at a DNS and not IP level.

Features of DNS

- **Global Distribution** - Data is maintained locally, but retrievable globally . No single computer has all DNS data . DNS lookups can be performed by any device
- **Scalability** - No limit to the size of the database . One server has over 20,000,000 names . No limit to the number of queries . 24,000 queries per second handled easily . Queries distributed among masters, slaves, and caches.
- **Dynamicity** - Database can be updated dynamically (Add/delete/modify) of any record .
- **Reliability** - Data is replicated . Data from master is copied to multiple slaves. Clients can query Master server and any of the copies at slave servers .Clients will typically query local caches

Advantages of DNS

- Internet Dependency
- Internet Speed.(High speed connections)
- Security
- IP Address Conversion
- Resolving names of World Wide Web (WWW) sites
- Routing messages to email servers and webmail services
- Connecting app servers, databases and middleware within a web application
- Virtual Private Networks (VPN)
- Instant messaging and online meeting services
- Communication between IoT devices, gateways and servers

DNS Weaknesses and Vulnerabilities

- **Internal DNS servers hold all the server names and IP addresses for their domains and will share them with anyone that asks.** This makes DNS a great source of information for attackers when they're trying to do internal reconnaissance.
- **DNS caches aren't "authoritative, and they can be manipulated.** If your DNS server is "poisoned" with bad records, computers can be fooled into going to bad places.
- **Registry Control** - Registry control of DNS is under ICANN. Which means that no other organizations will be able to control them. Hence, the concept of net neutrality is questioned here. ICANN is known as a non-profit organization which originates from one single country.

Cont....

- **Client Information** - DNS queries usually don’t carry any information about the clients who initiated it . This is because the server side will only see the IP address from where the query came from and which can at times be manipulated by hackers.
- **Server Breakdown** - When the DNS server gets broken down, the World Wide Web would crash too. Al though in the presence of back up servers and root servers. This is because **once the server crashes the connection to the local network will get disconnected** not allowing the clients to reach them.
- **DNS Attacks** -In here the original DNS address is replaced with a fake one so that users are redirected into fraudulent websites. From this attackers can gather sensitive information's such as bank account details.
- **Troubleshooting** -DNS issues are generally difficult to troubleshoot. This is due to its distributed nature and geographical locations.

Summary

- DNS is a client/server network communication protocol. DNS clients send requests to the. server while DNS servers send responses to the client.
- Client requests contain a name which is converted into an IP address known as a forward DNS lookups while requests containing an IP address which is converted into a name known as reverse DNS lookups.
- DNS implements a distributed database to store the name of all the hosts available on the internet.
- If a client like a web browser sends a request containing a hostname, then a piece of software such as **DNS resolver** sends a request to the DNS server to obtain the IP address of a hostname. If DNS server does not contain the IP address associated with a hostname, then it forwards the request to another DNS server. If IP address has arrived at the resolver, which in turn completes the request over the internet protocol.

Exercise 01

- **What’s the difference between a domain name and a URL?**
- **How is Domain Name Different from a Website and Web Hosting?**
- **Explain the Lookup zones in DNS.**
- **Differentiate host name and domain name.**
- **What Are A Forward And Reverse Lookup?**
- **What Are The Steps Involved In Registering The Domain?**
- **Differentiate Smart DNS and VPN**


Any Questions