# **DOMAIN NAME SYSTEM(DNS)**

**TABLE OF CONTENTS:**

**1.OVERVIEW.**

**2.DNS MESSAGE FORMAT (QUESTION & ANSWER).**

**3.DNS HIERARCHY.**

**4.DNS ZONE**

**5.DNS RESOULTION.**

**6.DNS RECORDS.**

**7.DNS REVERSE LOOKUP PROCESS.**

**8.DDNS.**

**9.DNS LINUX AND WIRESHARK CAPTURES.**

# 1.OVERVIEW:

* **DNS (Domain Name System)** is a hierarchical naming system that translates domain names into IP (Internet Protocol) addresses, which are used to locate computers and resources on the internet or a private network.
* It provides a more user-friendly way of accessing resources on the internet or a network than using IP addresses directly.
* DNS (Domain Name System) is like a phone book for the internet. When you type in a website address, like [www.google.com](http://www.google.com), your computer uses DNS to look up the IP address associated with that domain name. The IP address is like the phone number for that website, which allows your computer to connect to it.
* DNS is made up of many different servers, each with its own job. The servers work together to make sure your computer can find the IP address you need.
* DNS records are like entries in the phone book. They contain information about domain names and the IP addresses or other data associated with them, like mail servers or service providers. Several types of records have different purposes.
* Overall, DNS is an important part of how we navigate the internet and access the resources we need.

# **2.DNS MESSAGE FORMAT:**

## ***DNS QUERY:***

|  |
| --- |
| HEADER |
| QUESTION SECTION |

## ***DNS RESPONSE:***

|  |
| --- |
| HEADER |
| QUESTION SECTION |
| ANSWER SECTION |
| AUTHORATIVE SECTION |
| ADDITONAL SECTION |

### **HEADER:**

|  |  |
| --- | --- |
| IDENTIFICATION (16 BITS) | FLAGS (16 BITS) |
| NUMBER OF QUESTION RECORDS (16 BITS*)* | NUMBER OF ANSWER RECORDS (ALL 0’S IN QUERY) (16 BITS*)* |
| NUMBER OF AUTHORATIVE RECORDS (ALL 0’S IN QUERY) (16 BITS) | NUMBER OF ADDITONAL RECORDS (ALL 0’S IN QUERY) (16 BITS) |

### **IDENTIFICATION (16 BITS):**

* Identification field contains a number to identify and should map in response with same as Query.

### **FLAGS (16 BITS):**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Q/R (1) | OP CODE (4) | AA (1) | TC (1) | RD (1) | RA (1) | 0 | 0 | 0 | RCODE |

* **Q/R**:

0 -> QUERY

1 -> RESPONSE

* **OP CODE**:

Standard Query --> 0000

Reverse Query --> 0001

Service Status --> 0010

### **Authoritative Answer (AA):**

1 --> Name Server is Authoritative server.

* In Response message this AA is observed.

### **Truncate (TC):**

1 --> Response was more than 512 bytes and truncated to 512.

* Used when DNS uses services of UDP (User Datagram Protocol).

### **Recursion Desire (RD):**

1 --> When Client desires recursive.

### **Recursion Available (RA):**

1 --> Recursion is available.

* **Reserved**: 3 bit --> 0 0 0.
* **Rcode**: RCODE stands for "Response Code" and is a 4-bit field in the DNS message header that indicates the status of the DNS query response.
* The most used RCODE values include:
* 0 (NOERROR): The query was successful, and the response contains the requested information.
* 1 (FORMERR): The DNS message was malformed or improperly formatted.
* 2 (SERVFAIL): The server encountered an error while processing the query and was unable to complete it.
* 3 (NXDOMAIN): The requested domain name does not exist.
* 4 (NOTIMP): The server does not support the requested DNS query type.
* 5 (REFUSED): The server refused to respond to the query.
* 6 -15: Reserved.

## **Question Section (QUERY):**

* It is in the DNS message header and consists of the following fields:
* **QNAME:** The domain name being queried, expressed as a sequence of labels separated by dots. For example, "[www.google.com](http://www.google.com)" would be expressed as three labels: "www", "google", and "com".
* **QTYPE**: The type of DNS record being requested, such as A, AAAA, MX, CNAME, or others.
* **QCLASS**: The class of the DNS record being requested, usually IN (Internet) for queries over the public internet.
* The question section is used by the DNS client to specify the domain name and record type it wants to retrieve from the DNS server. When the DNS server receives the query, it checks its database for the requested information and responds with the appropriate DNS records in the answer section of the DNS message.

## **Answer Section:**

The answer section is in the DNS message header and consists of the following fields:

* **NAME**: The domain name that was queried, expressed as a sequence of labels separated by dots.
* **TYPE:** The type of DNS record that is being returned, such as A, AAAA, MX, CNAME, or others.
* **CLASS:** The class of the DNS record being returned, usually IN (Internet) for queries over the public internet.
* **TTL (time to live)**: The time-to-live value of the DNS record, which specifies how long the record can be cached before it needs to be refreshed from the authoritative source.
* **RDLENGTH**: The length of the RDATA field, which contains the actual data associated with the DNS record.
* **RDATA:** The actual data associated with the DNS record, such as an IP address or other information.

The answer section is used by the DNS server to provide the requested information to the DNS client. When the DNS client receives the DNS message with the answer section, it can use the information to connect to the requested resource, such as a web server or email server.

## **Authoritative section:**

The **authoritative section** is a part of the DNS message that contains information about the DNS server that is the ultimate source of information for the queried domain name. The authoritative section is in the DNS message header and consists of the following fields:

* **NAME:** The domain name that was queried, expressed as a sequence of labels separated by dots.
* **TYPE:** The type of DNS record that is being returned, usually NS (Name Server).
* **CLASS:** The class of the DNS record being returned, usually IN (Internet) for queries over the public internet.
* **TTL:** The time-to-live value of the DNS record, which specifies how long the record can be cached before it needs to be refreshed from the authoritative source.
* **RDLENGTH:** The length of the RDATA field, which contains the actual data associated with the DNS record.
* **RDATA:** The actual data associated with the DNS record, which is typically the domain name of the authoritative DNS server for the queried domain.

The authoritative section is used by the DNS client to determine the DNS server that is the ultimate source of information for the queried domain name. This is important because the DNS server that provides the authoritative answer is the one that is responsible for managing the domain name's DNS records and providing accurate information about them.

## **Additional section:**

* The additional section is a part of the DNS message that contains additional information that may be useful to the DNS client but is not strictly necessary for answering the query specified in the question section or the answer provided in the answer section. The added section is in the DNS message header and typically consists of one or more of the following types of records:
* **Additionally, A or AAAA records**: These records provide IP addresses for other DNS servers that are associated with the domain name being queried. This information can be useful for recursive DNS servers that need to continue the resolution process for more queries.
* **Additional MX records**: These records provide information about mail servers that are associated with the domain name being queried. This information can be useful for clients that need to send email to the domain.
* **Additional NS records**: These records provide the names of other DNS servers that are associated with the domain name being queried. This information can be useful for clients that need to continue the resolution process for added queries.
* **Additional CNAME records**: These records provide aliases for domain names that are associated with the domain name being queried. This information can be useful for clients that need to follow redirects or aliases.
* The added section is used by the DNS client to obtain more information that may be needed to complete a query or to perfect future queries. Although the information in the added section is not strictly necessary for answering the query, it can be useful for clients and can help to reduce the number of queries that need to be sent to the DNS server.

# **3.DNS HIERARCHY:**

* DNS (Domain Name System) hierarchy is a system used to organize and manage domain names in a hierarchical manner. The DNS hierarchy is based on a tree-like structure that starts with the root domain at the top, followed by top-level domains (TLDs), second-level domains, and so on.
* At the top of the hierarchy is the root domain, represented by a dot (.). Below the root domain are the TLDs, which are represented by strings such as .com,.org,. Edu, and so on. Each TLD (Top Level Domains) can have multiple second-level domains, such as example.com or example.org. Each second-level domain can have multiple subdomains, such as mail.example.com or ftp.example.com.
* The DNS hierarchy is important because it allows domain names to be organized in a logical and efficient manner, and it makes it possible for DNS servers to resolve domain names quickly and accurately to their associated IP addresses. When a user enters a domain name into a web browser, for example, the browser uses DNS to look up the IP address associated with that domain name. The DNS server starts at the root domain and works its way down the hierarchy until it finds the IP address associated with the domain name.

## **DNS HIERACHY DIAGRAM:**

Diagram

Description automatically generated

# **4.DNS ZONE:**

* **A zone** refers to a part of the DNS namespace that is managed by a particular authoritative name server. A DNS zone is typically associated with a specific domain name, and has information about the domain name's DNS records, such as A records, MX records, and NS records.
* A DNS zone is defined by a DNS zone file, which is a text file that has DNS records for the zone. The zone file is stored on the authoritative name server that handles managing the DNS zone and is used to answer DNS queries for the zone.
* There are two types of DNS zones: primary zones and secondary zones. A primary zone is the authoritative zone for a domain name and has the original copy of the zone file. The primary zone can be managed by a DNS administrator, and any changes made to the zone file are propagated to the secondary zones.
* A secondary zone is a read-only copy of the primary zone and is used to supply backup and redundancy for the DNS zone. The secondary zone is periodically synchronized with the primary zone to ensure that it has the most up-to-date DNS records.

### **DNS ZONE TRANSFER:**

* DNS zone transfer is a mechanism for transferring a complete copy of a DNS zone file from a primary DNS server to one or more secondary DNS servers.
* This enables the secondary DNS servers to answer DNS queries for the zone and supply redundancy in case the primary server is unavailable.
* Zone transfer is typically done over a TCP (Transmission Control Protocol) connection and ensures that all servers have an up-to-date copy of the zone file.

# **5.DNS RESOULTION:**

* DNS (Domain Name System) resolution is the process of converting domain names into IP addresses that can be used by computers to communicate with each other over the internet.
* When you enter a domain name into a web browser, for example, the browser uses DNS resolution to find the IP address associated with that domain name so it can connect to the server hosting the website.
* DNS resolution can be done in one of two ways: iterative resolution and recursive resolution.
* **Iterative resolution*:*** In iterative resolution, the DNS client sends a query to a DNS server and the DNS server responds with the best answer it has available. If the DNS server does not have a definitive answer, it supplies a referral to another DNS server that may have more information. The DNS client then sends a query to the new server and repeats the process until it either receives a definitive answer or reaches the end of the DNS hierarchy.
* **Recursive resolution*:*** In recursive resolution, the DNS client sends a query to a DNS server and the DNS server responds with the best answer it has available. If the DNS server does not have a definitive answer, it takes responsibility for finding the answer by sending queries to other DNS servers on behalf of the client. The recursive DNS server continues to send queries until it either receives a definitive answer or decides that the query cannot be resolved.

## **RECURSIVE AND ITERATIVE QUERY:**

***Diagram

Description automatically generated***

# **6.DNS RECORDS:**

* DNS (Domain Name System) records are a key part of the DNS system and are used to store several types of information about domain names. When a client requests information about a domain name, the DNS server responds with one or more DNS records that have the requested information. Here are some of the most common types of DNS records:
* **A record**: This is the most common DNS record type and is used to map a domain name to an IP address. When a client requests the IP address for a domain name, the DNS server responds with the A record that has the IP address.
* **CNAME record**: A CNAME (Canonical Name) record is used to create an alias for a domain name. For example, if you have a website hosted at example.com but also want people to access it using [www.example.com](http://www.example.com), you can create a CNAME record that maps [www.example.com](http://www.example.com) to example.com.
* **MX record:** An MX (Mail Exchange) record is used to specify the mail server that should receive email messages for a domain. When someone sends an email to an address at your domain, the DNS server looks up the MX record to figure out which server should receive the message.
* **NS record:** An NS (Name Server) record is used to specify the DNS server that is authoritative for a particular domain. When a client requests information about a domain, the DNS server responds with the NS record that has the address of the authoritative DNS server for that domain.
* **PTR record:** A PTR (Pointer) record is used to map an IP address to a domain name. This is the reverse of an A record, which maps a domain name to an IP address.
* **TXT record**: A TXT (Text) record is used to store text information about a domain. This can be used for various purposes, such as supplying more information about a domain or verifying domain ownership for diverse services.

Overall, DNS records are a critical part of the DNS system and are used to store several types of information about domain names. By mapping domain names to IP addresses and specifying which DNS servers are authoritative for a particular domain, DNS records make it possible for clients to resolve domain names quickly and accurately to their associated IP addresses.

# **7.DNS REVERSE LOOKUP PROCESS:**

DNS (Domain Name System) reverse lookup is the process of mapping an IP address to a domain name. This is the reverse of the more common forward lookup process, which maps domain names to IP addresses. Here is a general overview of how the DNS reverse lookup process works:

1. A client sends a DNS query to its local DNS resolver requesting the domain name associated with a particular IP address.
2. The local DNS resolver checks its cache to see if it already has the answer to the query. If the answer is in the cache, the resolver returns it to the client.
3. If the answer is not in the cache, the local DNS resolver sends a query to a root DNS server to get information about the top-level domain associated with the IP address.
4. The root DNS server responds with a referral to the proper top-level domain (such as .com, .net, or .org).
5. The local DNS resolver sends a query to the proper top-level domain server to get information about the domain associated with the IP address.
6. The top-level domain server responds with a referral to the authoritative DNS server for the domain.
7. The local DNS resolver sends a query to the authoritative DNS server for the domain to get the domain name associated with the IP address.
8. The authoritative DNS server responds with the domain name associated with the IP address.
9. The local DNS resolver stores the answer in its cache and returns it to the client.

* The DNS reverse lookup process involves a series of queries to various DNS servers to decide the domain name associated with a particular IP address.
* By using the DNS system to perform reverse lookups, clients can quickly and accurately map IP addresses to domain names, which is useful for various purposes, such as network troubleshooting, security analysis, and web analytics.

# **8.DYNAMIC DOMAIN NAME SYSTEM:**

* DDNS stands for Dynamic DNS, which is a service that allows clients with dynamic IP addresses to have a fixed hostname that can be accessed from the Internet.
* Dynamic IP addresses are typically assigned by Internet service providers (ISPs) and can change periodically. This makes it difficult to access devices or services hosted at home or in small businesses, since the IP address could change without notice, and there is no way to associate a hostname with a dynamic IP address.

With DDNS, clients can register a hostname with a DDNS provider, which associates the hostname with the client's current IP address. The client typically uses software or hardware that periodically updates the DDNS provider with its current IP address. This allows clients to have a fixed hostname that always points to their current IP address, even if it changes.

Here is an example of how DDNS works:

1. A client registers a hostname (such as myserver.example.com) with a DDNS provider and sets up a client software or hardware to periodically update the DDNS provider with the current IP address.
2. The client's ISP (Internet Service Provider) assigns a dynamic IP address to the client's device.
3. The client software or hardware detects the new IP address and updates the DDNS provider with the new IP address associated with the registered hostname.
4. Users on the Internet can access the client's device using the registered hostname (myserver.example.com), which always points to the client's current IP address, even if it changes.

Overall, DDNS is a useful service that allows clients with dynamic IP addresses to have a fixed hostname that can be accessed from the Internet. By periodically updating the DDNS provider with their current IP address, clients can ensure that their hostname always points to their current IP address, even if it changes. This makes it easier to access devices and services hosted at home or in small businesses and simplifies network setup and management.

# **9.DNS LINUX COMMANDS:**

* **dig**: a command-line tool used for querying DNS servers. It can be used to retrieve DNS records and perform several types of DNS queries, such as A, MX, NS, and TXT records.
* **nslookup**: a command-line tool used for querying DNS servers. It is like dig but supplies a simpler interface and fewer options.
* **host**: a command-line tool used to perform DNS lookups. It can be used to retrieve IP addresses for domain names, and vice versa.
* **Ipconfig/displaydns**: to display cache of all records in your pc

## **DNS Wireshark capture (query):**

***A picture containing graphical user interface

Description automatically generated***

## **DNS Wireshark capture(response):**

**Graphical user interface, application, Word

Description automatically generated**