UNIVERSITY OF DHAKA

Department of Computer Science and Engineering

CSE-3111 : Computer Networking Lab

Lab Report 4: Distributed Database Management and Implementation of Iterative and Recursive Queries of DNS Records.

Submitted By:

Name : Sudipto Das Sukanto

Roll No : 29

Name: Ahanaf Ahmed Shawn

Roll No : 47

Submitted On :

Feb 08, 2024

Submitted To :

Dr. Md. Abdur Razzaque

Dr. Md. Mamun Or Rashid

Dr. Muhammad Ibrahim

Mr. Md. Redwan Ahmed Rizvee

1 Introduction

A database management system that is dispersed among several computers and linked by a network is known as a distributed database management system (DDDBS). Multiple users can access and control the same data with this kind of database system, which also offers greater data availability and data redundancy.

It's critical to comprehend how the Domain Name System (DNS) functions while implementing recursive and iterative DNS record requests. Domain names can be resolved into IP addresses that computers can use to communicate with one another thanks to DNS, a hierarchical, distributed database that maps domain names to IP addresses.

In DNS, iterative searches entail making a succession of requests to various DNS servers until the required data is obtained. Recursive inquiries, on the other hand, require a single DNS server to handle the complete request, including any necessary follow-up requests.

Complex programming may be needed to implement repetitive and recursive searches in a DDDBS since doing so calls for coordinating efforts between several servers and managing numerous requests and responses. However, efficient and dependable DNS record resolution in a distributed context can be achieved by leveraging a well-designed database architecture.

1.1 File Transfer with Socket Programming and HTTP:

Socket programming provides the framework for low-level communication between clients and servers, making it ideal for building unique file transfer protocols. HTTP, on the other hand, is a standardized, high-level protocol used for web-based communication, including file transfer.

In practice, socket programming can be used to create proprietary file transfer mechanisms with more control over communication specifics, whereas HTTP-based file transfer uses existing HTTP servers and clients to facilitate implementation and interoperability.

1.2 Objectives

The following are the major goals of these lab experiments:

- Create distributed database management system: To provide a centralized and efficient way to manage and access data that is stored on multiple computers or nodes within a network.

- Implementing iterative and recursive queries of DNS records : To provide a fast and reliable way to resolve domain names into IP addresses.

2 Theory

A system called the Domain Name System (DNS) maps IP addresses (like 192.0.2.1) to domain names (like example.com) and vice versa. In order to convert human-readable domain names into the numerical IP addresses that computers use to communicate with one another over the Internet, it acts as a decentralised, hierarchical database.

When a user clicks on a hyperlink or enters a domain name into a web browser, a DNS query is initiated. A DNS resolver, often offered by the user's Internet service provider (ISP), receives a request from the user's computer. After then, the resolver sends the request to a number of DNS servers, working its way down the hierarchy from the root DNS servers to the server that is authoritative for the domain name in issue. The resolver sends the IP address associated with the domain name back to the user's computer after receiving it from the authoritative server.

Our computer goes through a number of processes when we visit a domain like google.com in order to translate the human-readable web address into an IP address that is readable by machines. The following is the IP address retrieval procedure:

- 1. Search in DNS cache: Our computer searches in its local DNS cache first when we ask it to resolve a hostname. Our computer must run a DNS query to obtain the answer if it isn't already aware of it.

- 2. Request to ISP DNS servers: In the event that the data is not locally cached, our computer makes a DNS server query to our ISP.

- 3. Request to root nameservers: These servers will make a query to the root nameservers if they are unable to provide the solution. A nameserver is a machine that is always on and answers questions regarding IP addresses and domain names.

- 4. The initial portion of our request, which is read from right to left by the root nameservers, will be examined and our inquiry will be forwarded to the Top-Level Domain (TLD) nameservers for.com (google.com). Every Top Level Domain (TLD), including.com,.org, and.bd, has its own collection of nameservers that function as each TLD's front desk.

- 5. Request to authoritative DNS servers: After examining the following section of our request, the TLD nameservers forward our inquiry to the nameservers in charge of this particular domain. The DNS records provide all the information that these authoritative nameservers need to know about a particular domain.

3 Methodology

- 1. For every DNS server, we launch a thread.
- 2. We establish a root node in a tree of DNS servers.
- 3. A reference to the parent server and child servers is given to each server.
- 4. The server is waiting for a client request.
- 5. The server searches for the specified domain and any children (if any) in its current location after receiving the request. If the request cannot be located, it will be forwarded to the parent.
- 6. The parent conducts a comparable search and notifies its parents if the request is not discovered.
- 7. The client will receive an error message if none of the DNS servers are able to locate the domain.

4 Experimental Result

4.1 Part 1: Setting up the DNS server

4.1.1 Server:

The client submits a request to the server.The header and the IP address of the requested domain name are sent by the client if it wishes to know the IP address of a domain name.There is a comparable Header format for both kinds of messages. Up to five distinct DNS message parts can contain the information. The header and question records are the two sections that make up the query message.

The response message consists of five sections:

- 1. Header

- 2. Question
- 3. Records
- 4. Answer records
- 5. Authoritative records
- 6. Additional records

DNS message format is 12 bytes.

4.1.2 Clint

The client can request for IP address for any domain name.After accepting the request by server the header and IP will show on the client's screen.

4.2 Part 2: Iterative DNS resolution

DNS clients employ iterative DNS resolution, sometimes referred to as iterative querying or iterative lookup, to ask DNS servers questions about a domain name.

A DNS server receives a query from a client and returns the best response it can in an iterative DNS resolution process. The information requested by the client is returned by the server in the response if it is available. A recommendation to another DNS server that might have the information is returned by the server if it does not contain the information.

After that, the client queries the subsequent DNS server, and so on, until either the needed data is located or there are no more referrals. The reason this procedure is named iterative is that the client asks the same question repeatedly and receives the same answers until it finds the solution.

4.2.1 Server:

A DNS server that operates by querying multiple servers until the proper IP address of a domain name is found is known as an iterative DNS resolution server. In order to reach the top-level domain (TLD) DNS server, the server first sends the query to a root DNS server. The authoritative DNS server for the disputed domain receives a referral from the TLD DNS server and replies with the IP address.

- Root Server: The local DNS resolver receives DNS queries sent by clients seeking to resolve domain names. The local resolver forwards the query to the root server if it does not contain the requested data in its cache. The top-level domain (TLD) server in charge of the TLD linked to the domain name being resolved is referred to by the root server in response to the query.

- TLD Server: The top-level domain (TLD) server is an essential component of an iterative DNS resolution process. Its primary purpose is to direct the user to the domain name's authoritative name server.When a client sends a DNS query to resolve a domain name, the query first reaches the local DNS resolver. If the local resolver does not have the requested information in its cache, it sends the query to the root server. The root server then provides a referral to the TLD server responsible for the TLD associated with the domain name being resolved.

- Authoritative Server: In the iterative DNS resolution process, the authoritative server is in charge of responding to a DNS query with the most precise and final response. This server is the source of truth for all information pertaining to a certain domain and contains the authoritative data for that domain.A client computer executing a DNS query may initially check its local cache or local DNS resolver, which may thereafter ask many DNS servers to ascertain the IP address linked to a domain name. A root DNS server will forward a request to a top-level domain (TLD) server if the local resolver does not have the necessary information stored. The resolver will then be directed by the TLD server to the authoritative server for the concerned domain.

4.2.2 Client:

The local DNS resolver, which serves as the initial step in the resolution process, receives the DNS query packet from the client.In addition, the client has the responsibility of receiving the root server's answer, which directs the user to the relevant TLD server, and interpreting it to ascertain the subsequent course of action in the resolution procedure.

4.3 Part 3: Recursive DNS resolution

DNS servers employ a sort of DNS resolution procedure called recursive DNS resolution to translate domain names into IP addresses. Recursive resolution involves the DNS server iterating through a sequence of queries and responses until it discovers the answer in order to fully resolve the query on behalf of the client.

Here is how recursive DNS resolution works:

The DNS server receives a query from the client that includes the domain name that has to be resolved.The DNS server sends the response back to the client if it has the answer cached.The DNS server sends a query to another DNS server if it does not already have the answer in its cache.If the second DNS server has the answer, it verifies its cache and sends it back to the first DNS server. It queries a third DNS server if it is unable to resolve the issue.Until the solution is found, the process is repeated. The client receives the response from the DNS server, which saves it in its cache for upcoming queries.

Many DNS servers, including those from Internet service providers (ISPs) and open DNS resolvers like Google DNS and OpenDNS, use recursive DNS resolution. It is frequently compared to iterative DNS resolution, in which the client sends queries to servers and receives answers until it finds what it's seeking for.

4.3.1 Server :

A DNS server uses recursive DNS resolution to translate domain names into IP addresses by contacting other DNS servers. The recursive DNS server initiates the domain name resolution process when a client makes a request by querying the root server to get the IP address of the top-level domain server in charge of the requested domain name.

The IP address of the authoritative name server for the requested domain is then obtained by the recursive DNS server by contacting the top-level domain (TLD) server. The DNS records for the requested domain name are kept up to date by the authoritative name server. To find the IP address connected to the domain name, the recursive DNS server queries the authoritative name server.

In order to speed up subsequent requests for the same domain name, the recursive DNS server caches the IP address it receives from the authoritative name server in its local memory.The IP address is then returned to the client that made the initial request by the recursive DNS server.

- Root Server: To determine which TLD server is in charge of the domain name being questioned, the DNS server handling the recursive DNS resolution 10 begins by submitting a query to a root server. The DNS server queries the root server for the IP address of the domain name being resolved, and the root server provides a referral with the IP address of the relevant TLD server.

- TLD Server: In recursive DNS resolution, a DNS server first sends a request to one of the 13 root servers in order to determine which TLD server is in charge of the domain name that is being queried. When a domain name is queried, the root server provides a reference that includes the IP address of the TLD server in charge of that TLD.

  The DNS server queries the TLD server to obtain the IP address of the domain name being resolved after obtaining the IP address of the TLD server. The IP address of the authoritative name server for the domain being resolved is included in the referral that the TLD server sends in response. After obtaining the IP address of the domain name being resolved through a query to the authoritative name server, the DNS server replies with the IP address to the initial requester.

- Authoritative Server: Recursive DNS resolution involves the DNS server sending a request for the IP address connected to the domain name to the domain's authoritative name server. The DNS server then sends the domain name's IP address back to the original requester after receiving the response from the authoritative name server. Since authoritative name servers are in charge of keeping the most recent DNS entries for domain names, they are crucial to the DNS system's operation. DNS resolution and name-based access to webpages and other Internet services would not be feasible without authoritative name servers.

4.3.2 Client:

The device or programme that starts a DNS query to resolve a domain name into an IP address is known as the client in recursive DNS resolution. Any device or programme that has to communicate via the Internet with a server or service might be considered the client.

A recursive DNS server receives a query from a client requesting domain name resolution in order to resolve the domain name. After that, the recursive DNS server handles DNS resolution on the client's behalf by progressively contacting different DNS servers until it finds the IP address associated with the requested domain name.

4.4 Part 4: Extending the System

In order to speed up its response to repeated queries for the same domain name, DNS servers cache previously resolved domain name-to-IP address mappings in memory. A DNS server first looks in its cache to see if it has the IP address associated with the domain name when it gets a query for a domain name. The IP address is returned to the requester by the DNS server without requiring a complete DNS resolution if it already has a record of the domain name in its cache.

The time and network resources needed to answer DNS queries can be greatly decreased by DNS caching, especially for commonly visited domain names. A DNS server can respond to queries for frequently visited domain names more rapidly and with less network traffic by caching DNS records. This eliminates the need to contact other DNS servers.