DNS SERVER

COURSE TEACHER: Mr. Monthasir Delwar Afnan
COURSE: Operating Systems Lab
COURSE CODE: CSE-324
SECTION: 61_S
Department of CSE
Daffodil International University

MEET THE TEAM

- Amit Kumar Ghosh (221-15-4650)
- Al Amin (221-15-4779)
- Mehedi Hasan (221-15-4680)
- Sadiqul Haque Sadib (221-15-4863)

TABLE OF CONTENTS

- OJECTIVE
- WHY CHOOSE US
- INTRODUCTION
- HOW DNS WORKS
- TYPES OF DNS SERVER
- OUR DNS SERVER SETUP
- PSEUDO CODE
- FLOWCHART
- DNS CACHING
- OUR DNS SERVER SECURITY
- ADVANTAGES AND DISADVANTAGES OF DNS
- CONCLUSION

OBJECTIVE

Our main objective was to dive into the concept of Domain Name System (DNS), learn how it works, why we need it, and finally build a DNS server ourselves. We firmly believe that the most effective way to learn is through hands-on experience.

WHY CHOOSE US

- Lightweight
- Highly reliable
- DNS Caching
- Security measure
- Huge potential

INTRODUCTION TO SIMPLE DNS SERVER

UNDERSTANDING THE BASICS OF DOMAIN NAME SYSTEM (DNS)

Definition:
DNS (Domain Name System) is an internet service that translates human-readable domain names (like www.example.com) into machine-readable IP addresses (like 127.1.0.1). This allows users to access websites without needing to memorize numerical IP addresses.

Purpose of Our DNS:
DNS acts as the "phonebook" of the internet, enabling quick and easy access to websites, email services, and other online resources by linking domain names to their corresponding IP addresses.

HOW DNS WORKS

STEPS INVOLVED IN DNS RESOLUTION

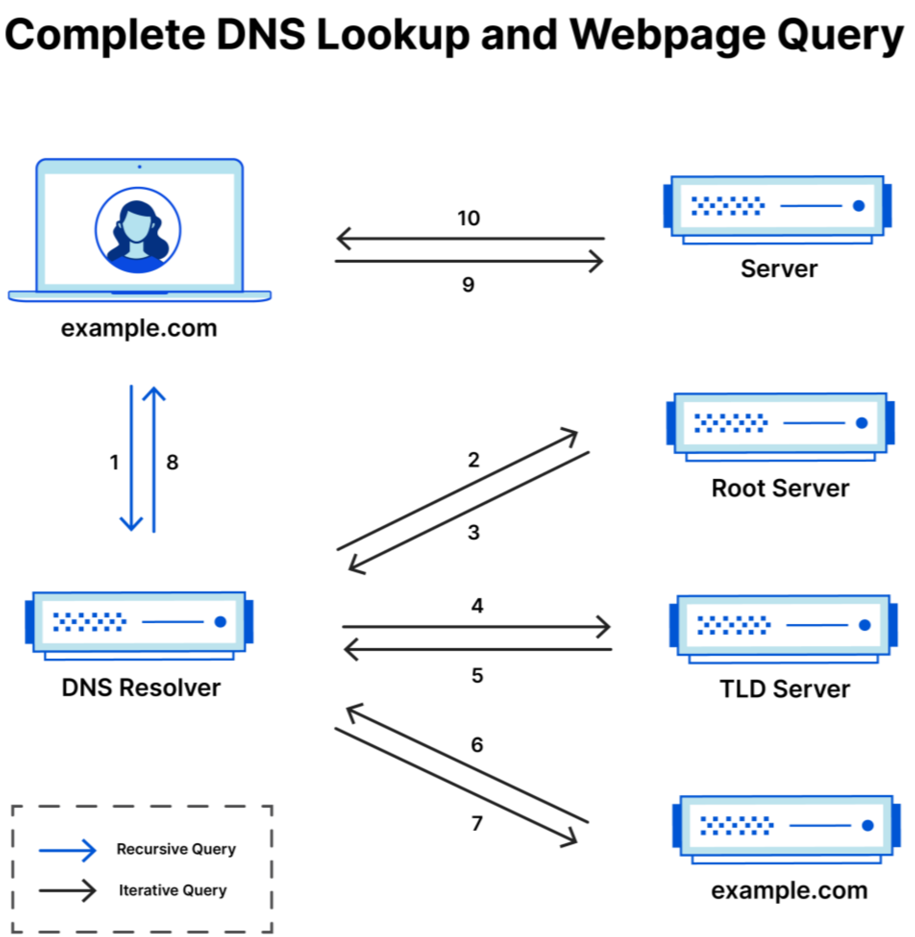
1. **User Request:**
2. **DNS Resolver:**
3. **Root Server:**
4. **TLD Server:**
5. **Authoritative DNS Server**
6. **Response to the User :**

Main Components of DNS:

1. DNS Client
2. DNS Resolver
3. DNS Server

HOW DNS WORKS

UNDERSTANDING WORKING FLOW OF DOMAIN NAME SYSTEM (DNS)



FLOW OF DNS SERVER

1. **Root DNS Server:**

Role: Root Server directs the query to the appropriate Domain Server based on the domain suffix (e.g., .com, .org, .net).

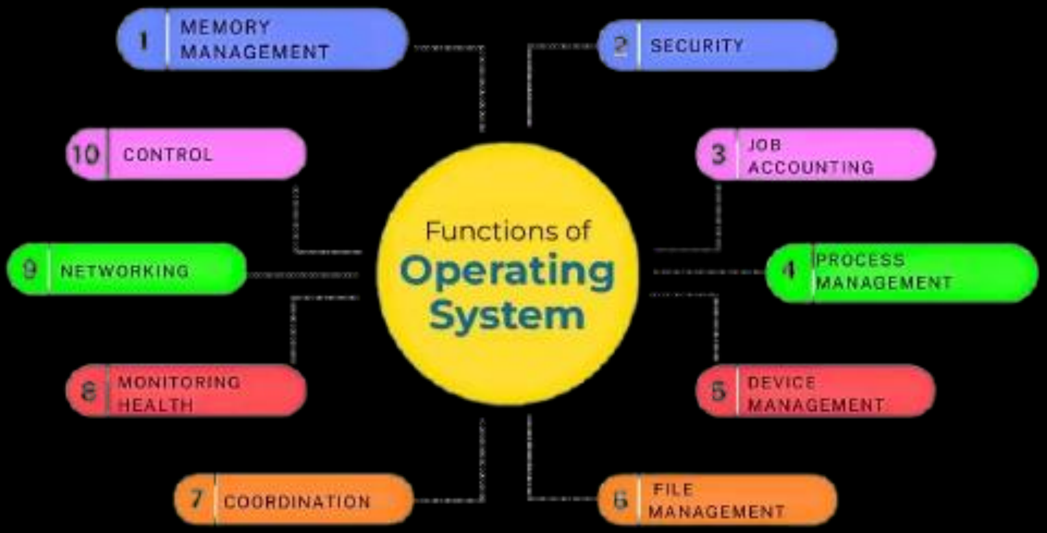
Key Point: There are only 13 sets of Root DNS Servers globally.

2. TLD (Top-Level Domain) Server:

Role: The TLD Server manages entire domains and sends to its authoritative server such as .com, .org, .edu, etc and send to Authoritative.

3. Authoritative DNS Server:

Role: This DNS Server contains the actual DNS records for a specific domain.
Type, Address, Class, Domain name, IP.



OUR DNS SERVER SETUP

Step-1: Install Node.js

Step-2: Install DNS Packet

Step-3: Using UDP Socket to handle all user request

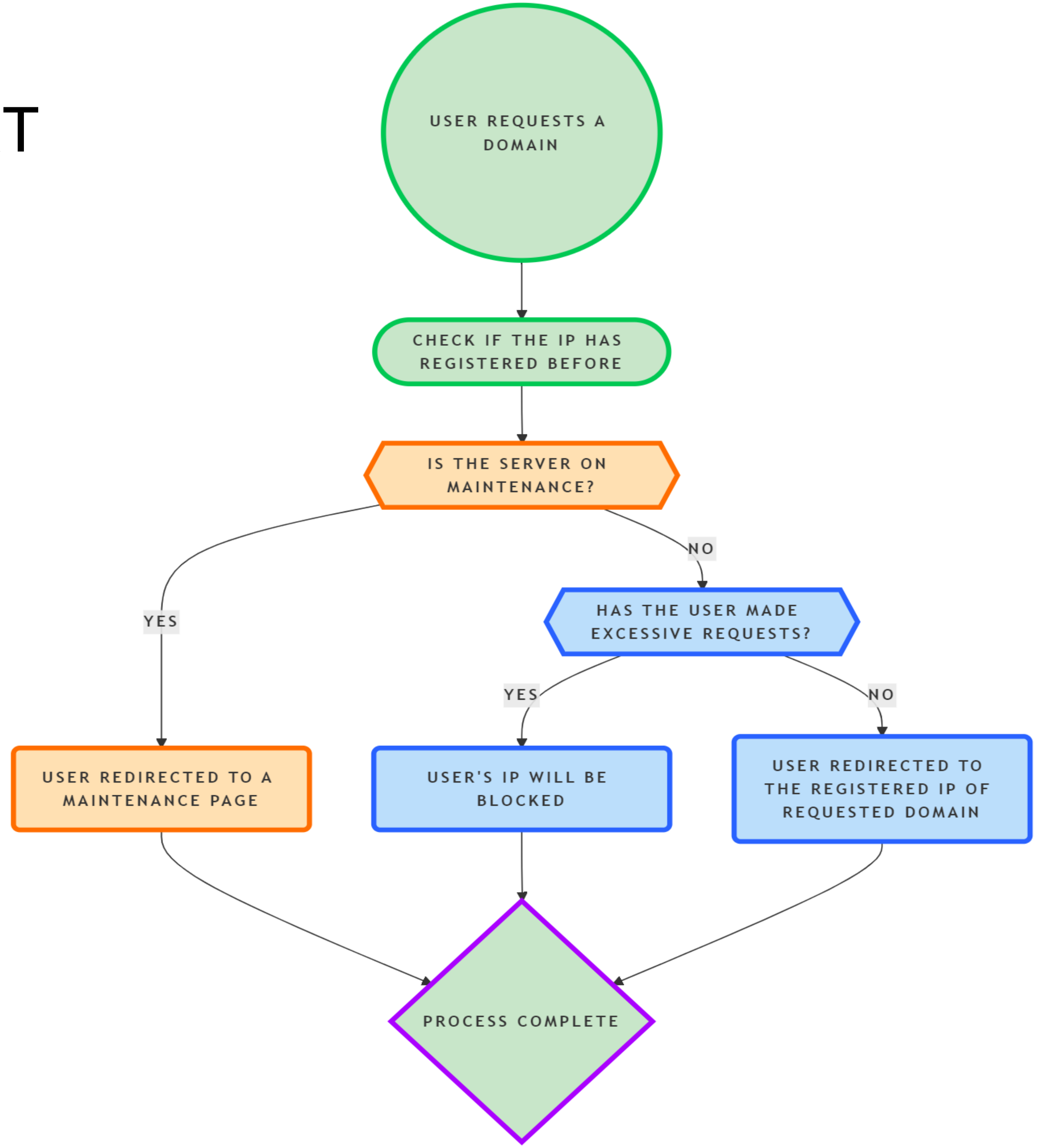
Step-4: Setup Domain & Register IP

Step-5: Request Domain from user

PSEUDO CODE

```
1: User Request a domain
2: User get checked if his IP has registered before
3: If user do excessive requests user’s IP will be blocked
4: It will check if is server on maintenance
5: After all that user will redirected to the registered IP of requested domain
```

FLOWCHART


USER REQUESTS A DOMAIN
CHECK IF THE IP HAS REGISTERED BEFORE
IS THE SERVER ON MAINTENANCE?
YES
NO
HAS THE USER MADE EXCESSIVE REQUESTS?
YES
NO
USER REDIRECTED TO A MAINTENANCE PAGE
USER'S IP WILL BE BLOCKED
USER REDIRECTED TO THE REGISTERED IP OF REQUESTED DOMAIN
PROCESS COMPLETE

DNS CACHING

What is DNS Caching ?

DNS caching speeds up loading times by storing DNS information locally. When a DNS Caching stores recently resolved DNS queries so that repeated requests for the same domain can be served faster without querying the IP database repeatedly.

Cache Storage: DNS cache is an object where each domain has a cached response along with timestamp and time to live (TTL) values.

Clear Expired Cache: It checks the current time against each entry’s timestamp and TTL. If an entry has expired, it is deleted from the cache.

Checking the Cache: When a query is received, the server first checks if the requested domain is in DNS cache and hasn’t expired. If found, it sends the cached response to the client. If not found, it fetches the IP from DB of IP, creates a response, caches it, and sends it to the client.

Setting TTL: The default TTL is set to 5 minutes (300,000 milliseconds) in our DNS server, but it can be adjusted as needed

OUR DNS SERVER SECURITY

Security Measure

Rate Limiting: Limits each IP to five requests per minute. This prevents overuse and helps protect against denial-of-service (DoS) attacks.

IP Blocking: If an IP makes too many requests, it is added to a block list to prevent further access.

Limitation

No DNSSEC:
Issue: Without DNSSEC (Domain Name System Security Extensions), this server can be tricked into providing fake responses (DNS spoofing).
Solution: Implement DNSSEC to verify that DNS responses are authentic and not altered.

No Encryption:
Issue: DNS queries and responses are unencrypted, making them easy to intercept. -
Solution: Consider using **DNS over HTTPS (DoH)** or **DNS over TLS (DoT)** to secure DNS traffic.

ADVANTAGES AND DISADVANTAGES OF DNS

Advantages

Easy to Use: Our DNS lets us use website names instead of IP addresses, making it simpler to browse the internet.

Security: We implemented high security, so a user can not request too many times in a selected time window.

Fast Access: We implemented DNS caching helps websites load faster and reduces server traffic by storing frequently used data locally.

Reliable: Our DNS server is highly reliable, faster and secure.

Disadvantages:

Browser Dependency: Our DNS server can perform poorly depending on user browser

High Traffic: due to High traffic our performance can vary.

Costly Maintenance: As the DNS server is running 24/7 it could become costly.

Privacy Concerns: DNS queries are often visible to others, which can reveal user browsing habits.

CONCLUSION

DNS is an essential component of the internet, allowing users to access websites through human-readable domain names rather than numeric IP addresses. Throughout the presentation, we have explored how DNS works, including the different types of DNS servers involved in resolving domain names, the importance of DNS caching for faster access, and the security measures like DNSSEC that protect against attacks.

While DNS offers significant advantages, such as scalability, speed, and reliability, it also faces challenges like security vulnerabilities and privacy concerns. Overall, DNS is the backbone of internet navigation, and ensuring its security and efficiency is crucial for maintaining a seamless and safe online experience.

ANY QUESTIONS?

THANK YOU

GitHub Repo