

CN: Assignment 2

- CN: Assignment 2
 - Task: DNS Query Resolution
 - **Setup**
 - **A. Simulate the below given topology in Mininet and demonstrate successful connectivity among all nodes. (20 Points)**
 - **B. In the simulated topology, use the default host resolver to resolve the URLs specified in each host's respective PCAP file, and record for each host the average lookup latency, average throughput, number of successfully resolved queries, and number of failed resolutions. (10 Points)**
 - **C. Modify the DNS configuration of all Mininet hosts to use your custom resolver as the primary DNS server instead of the default system resolver. Show all the steps along with relevant screenshots. (10 Points)**
 - **D. Repeat DNS resolution for the given PCAPs as done in part B, this time using your custom DNS resolver (10.0.0.5). Compare the results with those obtained in part B. Additionally, log the following in the custom DNS resolver:**
 - Plots for H1 (first 10 URLs)
 - **F. Implement caching within your custom DNS resolver. Store recently resolved domain-to-IP mappings and serve repeated queries directly from cache without contacting external servers. Record the average lookup latency, average throughput, number of successfully resolved queries, % of queries resolved from cache and number of failed resolutions. (2.5 Points)**

Team Details:

| Name | Roll Number |
|---------------|-------------|
| Romit Mohane | 23110279 |
| Bhavya Parmar | 23110059 |

Github Repository Link: [Reckadon/mininet-assignment](https://github.com/Reckadon/mininet-assignment)

Task: DNS Query Resolution

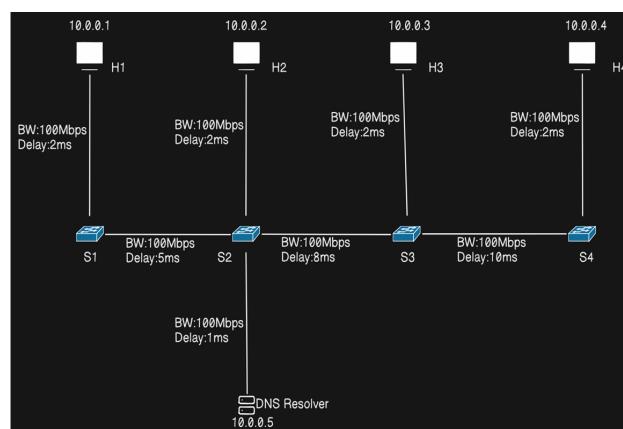
Setup

To Setup the assignment, we have used the **Mininet-VM** on Oracle VirtualBox. The setup was done by following instructions from [this](#) page. Then, we SSHed into the VM from our host system to write code and run commands.

Remember to run the following command to bring up Internet connection in the VM.

```
sudo dhclient eth1
```

A. Simulate the below given topology in Mininet and demonstrate successful connectivity among all nodes. (20 Points)



The following script ([dns_topo.py](#)) was used to create and check the connections for the given topology.

```
from mininet.topo import Topo
from mininet.net import Mininet
from mininet.node import OVSCController
from mininet.link import TCLink
from mininet.cli import CLI
import csv

class DNSTopo(Topo):
    def build(self):
```

```

# Hosts
h1 = self.addHost('h1', ip='10.0.0.1/24')
h2 = self.addHost('h2', ip='10.0.0.2/24')
h3 = self.addHost('h3', ip='10.0.0.3/24')
h4 = self.addHost('h4', ip='10.0.0.4/24')
dns = self.addHost('dns', ip='10.0.0.5/24')

# Switches
s1 = self.addSwitch('s1')
s2 = self.addSwitch('s2')
s3 = self.addSwitch('s3')
s4 = self.addSwitch('s4')

# Host-Switch links
self.addLink(h1, s1, bw=100, delay='2ms')
self.addLink(h2, s2, bw=100, delay='2ms')
self.addLink(h3, s3, bw=100, delay='2ms')
self.addLink(h4, s4, bw=100, delay='2ms')

# Switch-Switch links (core network)
self.addLink(s1, s2, bw=100, delay='5ms')
self.addLink(s2, s3, bw=100, delay='8ms')
self.addLink(s3, s4, bw=100, delay='10ms')

# DNS resolver link
self.addLink(s2, dns, bw=100, delay='1ms')

def log_latencies(net):
    """Ping all host pairs and save results to CSV."""
    hosts = net.hosts
    csv_file = "latency_matrix.csv"

    with open(csv_file, 'w', newline='') as f:
        writer = csv.writer(f)
        # Header row
        header = ["Source/Destination"] + [h.name for h in hosts]
        writer.writerow(header)

        for src in hosts:
            row = [src.name]
            for dst in hosts:
                if src == dst:
                    row.append("-") # no self-latency
                    continue
                # Ping once and extract average latency
                latency = src.cmd(f'ping -c 1 {dst.IP()} | tail -1 | cut -d"/" -f5').strip()
                row.append(latency if latency else "timeout")
            writer.writerow(row)

    print(f"\n☑ Latency matrix saved to {csv_file}")

if __name__ == '__main__':
    net = Mininet(topo=DNSTopo(), controller=OVSCController, link=TCLink)
    # nat = net.addNAT(name='nat0', connect='s2', ip='10.0.0.254/24').configDefault()
    net.start()
    print("**** Network started")
    print("**** Testing connectivity:")
    net.pingFull()
    log_latencies(net)
    print("**** Dropping into CLI: test manually if needed")
    CLI(net)
    net.stop()

```

```
mininet@mininet-vm:~/assignments/mininet-assignment$ sudo python dns_topo.py
*** Network started
*** Testing connectivity:
*** Ping: testing ping reachability
dns -> h1 h2 h3 h4
h1 -> dns h2 h3 h4
h2 -> dns h1 h3 h4
h3 -> dns h1 h2 h4
h4 -> dns h1 h2 h3
*** Results:
dns->h1: 1/1, rtt min/avg/max/mdev 40.201/40.201/40.201/0.000 ms
dns->h2: 1/1, rtt min/avg/max/mdev 16.967/16.967/16.967/0.000 ms
dns->h3: 1/1, rtt min/avg/max/mdev 52.192/52.192/52.192/0.000 ms
dns->h4: 1/1, rtt min/avg/max/mdev 95.173/95.173/95.173/0.000 ms
h1->dns: 1/1, rtt min/avg/max/mdev 21.064/21.064/21.064/0.000 ms
h1->h2: 1/1, rtt min/avg/max/mdev 43.401/43.401/43.401/0.000 ms
h1->h3: 1/1, rtt min/avg/max/mdev 79.936/79.936/79.936/0.000 ms
h1->h4: 1/1, rtt min/avg/max/mdev 120.339/120.339/120.339/0.000 ms
h2->dns: 1/1, rtt min/avg/max/mdev 9.501/9.501/9.501/0.000 ms
h2->h1: 1/1, rtt min/avg/max/mdev 22.458/22.458/22.458/0.000 ms
h2->h3: 1/1, rtt min/avg/max/mdev 55.792/55.792/55.792/0.000 ms
h2->h4: 1/1, rtt min/avg/max/mdev 103.186/103.186/103.186/0.000 ms
h3->dns: 1/1, rtt min/avg/max/mdev 26.126/26.126/26.126/0.000 ms
h3->h1: 1/1, rtt min/avg/max/mdev 39.787/39.787/39.787/0.000 ms
h3->h2: 1/1, rtt min/avg/max/mdev 28.891/28.891/28.891/0.000 ms
h3->h4: 1/1, rtt min/avg/max/mdev 64.978/64.978/64.978/0.000 ms
h4->dns: 1/1, rtt min/avg/max/mdev 47.675/47.675/47.675/0.000 ms
h4->h1: 1/1, rtt min/avg/max/mdev 60.840/60.840/60.840/0.000 ms
h4->h2: 1/1, rtt min/avg/max/mdev 48.343/48.343/48.343/0.000 ms
h4->h3: 1/1, rtt min/avg/max/mdev 31.652/31.652/31.652/0.000 ms

✓ Latency matrix saved to latency_matrix.csv
*** Dropping into CLI: test manually if needed
mininet> 
```

This uses `pingFull` to display all the connection metrics as well. By running it with `sudo` we get:

We also get access to the mininet CLI from here, which can be used to check individual connections and run scripts on specific hosts. You can type `exit` to close the topology and CLI.

The latency matrix is as follows:

| Source/Destination | dns | h1 | h2 | h3 | h4 |
|--------------------|--------|--------|--------|--------|--------|
| dns | — | 17.653 | 6.992 | 23.770 | 46.273 |
| h1 | 17.848 | — | 20.470 | 38.208 | 62.031 |
| h2 | 7.590 | 19.783 | — | 29.870 | 48.204 |
| h3 | 24.067 | 36.649 | 26.144 | — | 29.150 |
| h4 | 46.017 | 56.523 | 50.763 | 30.902 | — |

To run commands from a specific node of the network, we can use the following syntax:

```
<node to execute command from> <command>
```

example:

```
h1 ping -c 2 dns
```

This pings the DNS node from H1 node for 2 packets.

B. In the simulated topology, use the default host resolver to resolve the URLs specified in each host's respective PCAP file, and record for each host the average lookup latency, average throughput, number of successfully resolved queries, and number of failed resolutions. (10 Points)

We first extracted the DNS queries from the respective PCAP files (stored in /pcap) to get the URLs to be resolved, using the command:

```
tshark -r pcap/PCAP_3_H3.pcap -Y "dns && udp.port == 53" -T fields -e dns.qry.name > pcap/h3_domains.txt
```

Breakdown of the command:

| Part | Description |
|------|-------------|
|------|-------------|

| Part | Description |
|----------------------------|--|
| tshark | Command-line version of Wireshark used for analyzing packet capture (PCAP) files. |
| -r pcap/PCAP_3_H3.pcap | Reads packets from the specified PCAP file corresponding to Host H3. |
| -Y "dns && udp.port == 53" | Display filter ensuring that only DNS packets (port 53) are processed, filtering out non-DNS traffic like LLMNR or mDNS. |
| -T fields | Specifies that only selected fields (not full packet details) should be printed. |
| -e dns.qry.name | Extracts the domain name field from each DNS query packet. |
| > pcap/h3_domains.txt | Redirects the extracted domain names into a text file for further processing. |

This was repeated for all the pcap files, to get `h1_domains.txt`, `h2_domains.txt`, `h3_domains.txt` and `h4_domains.txt`.

Then, we used this script (`resolve_default.py`) to resolve the URLs using socket's `gethostbyname()`. The VM must be connected to the Internet through the Host, by using a NAT Adapter. We set the nameserver to Google's DNS (8.8.8.8) using `h1 python -c "open('/etc/resolv.conf', 'w').write('nameserver 8.8.8.8\\n')"`.

We used Google DNS since we couldn't figure out a way to connect to the VM's default host resolver. The Mininet VM used `systemd-resolved` with a local stub (127.0.0.53) and upstream DNS (10.0.2.3). However, since 10.0.2.3 is part of the VM's internal NAT network and unreachable from the Mininet virtual network (10.0.0.0/24), it could not be used directly by the Mininet hosts. Instead, a public DNS (8.8.8.8) was configured in each Mininet host's `/etc/resolv.conf` to ensure external DNS resolution via the NAT gateway.

The script can be run on each host as such:

```
h1 python resolve_default.py H1
```

where you can replace h1/H1 with h2/H2, h3/H3 and h4/H4 respectively.

The detailed records for each Host can be found in `results/H1_default_results.csv`, `results/H2_default_results.csv`, `results/H3_default_results.csv` and `results/H4_default_results.csv` respectively. The summary of all the records can be found in `results/default_summary.csv`. Also given below is the summary:

| Host | Total Queries | Success | Failed | Avg Latency (ms) | Throughput (qps) |
|------|---------------|---------|--------|------------------|------------------|
| H1 | 100 | 71 | 29 | 166.42 | 6.01 |
| H2 | 100 | 68 | 32 | 110.20 | 9.07 |
| H3 | 100 | 72 | 28 | 131.48 | 7.6 |
| H4 | 100 | 73 | 27 | 120.46 | 8.3 |

It should be noted that the latencies and throughput depend on that particular instant's system performance (memory usage etc), since the VM's performance is impacted heavily sometimes.

C. Modify the DNS configuration of all Mininet hosts to use your custom resolver as the primary DNS server instead of the default system resolver. Show all the steps along with relevant screenshots. (10 Points)

To modify the DNS configuration of all hosts from `h1 → default system resolver → external DNS servers` to `h1 → 10.0.0.5 (custom resolver) → external servers (Google/Root/etc.)`:

1. Start the topology:

```
sudo python dns_topo.py
```

2. Inside the mininet CLI, we can verify the nodes and their IP addresses using: `<node> ifconfig`.

3. We start running the custom DNS resolver we built in the previous assignment, using

```
dns python custom_dns.py &
```

(The '&' runs it in the background. To check, you can run `dns ps`)

4. Now, we will configure the hosts to use our custom DNS server:

```
h1 python -c "open('/etc/resolv.conf', 'w').write('nameserver 10.0.0.5\\n')"
```

This will set the nameserver to `10.0.0.5`, which all hosts can now use.

We can check for the changes using `cat` command to open the file.

```
mininet> h4 python -c "open('/etc/resolv.conf', 'w').write('nameserver 10.0.0.5\\n')"
mininet> h1 cat /etc/resolv.conf
nameserver 10.0.0.5
```

D. Repeat DNS resolution for the given PCAPs as done in part B, this time using your custom DNS resolver (10.0.0.5). Compare the results with those obtained in part B. Additionally, log the following in the custom DNS resolver:

- **Timestamp**
- **Domain name queried**
- **Resolution mode**
- **DNS server IP contacted**
- **Step of resolution (Root / TLD / Authoritative / Cache (if caching implemented))**
- **Response or referral received**
- **Round-trip time to that server**
- **Total time to resolution**
- **Cache status (HIT / MISS) (if caching implemented)**

For PCAP_1_H1, present graphical plots for the first 10 URLs, showing the total number of DNS servers visited and latency per query.

The Pure Iterative DNS Resolver can be found in `custom_dns.py`

To let the Mininet nodes access the internet: Enable IP forwarding on the VM

```
sudo sysctl -w net.ipv4.ip_forward=1
```

And, uncomment the line `nat = net.addNAT(name='nat0', connect='s2', ip='10.0.0.254/24').configDefault()` in `dns_topo.py`. This connects `nat0` to the Mininet topology (via `s2`) and to the VM's external interface (`eth1`), providing Internet access.

To test DNS resolution:

```
h1 dig example.com
```

The resolver logs can be found in `results_custom` folder in `steps` and `summary` log files. (`H1_steps.csv`, `H1_summary.csv`, etc.)

```
resolver_detailed_steps.csv
1 timestamp,domain,resolution_mode,dns_server_ip,step,response_type,rtt_ms,total_time_ms,cache_status
2 2025-10-28 09:11:35,tuhafhaberler.com,iterative,198.41.0.4,ROOT,REFERRAL,172.14,-,N/A
3 2025-10-28 09:11:35,tuhafhaberler.com,iterative,192.41.162.30,TLD/AUTHORITATIVE,REFERRAL,161.69,-,N/A
4 2025-10-28 09:11:35,tuhafhaberler.com,iterative,192.48.79.30,TLD/AUTHORITATIVE,REFERRAL,155.67,-,N/A
5 2025-10-28 09:11:35,tuhafhaberler.com,iterative,192.54.112.30,TLD/AUTHORITATIVE,NO_RESPONSE,timeout,-,N/A
6 2025-10-28 09:11:35,tuhafhaberler.com,iterative,192.31.80.30,TLD/AUTHORITATIVE,NO_RESPONSE,timeout,-,N/A
7 2025-10-28 09:11:35,tuhafhaberler.com,iterative,192.33.14.30,TLD/AUTHORITATIVE,REFERRAL,270.17,-,N/A
8 2025-10-28 09:11:35,tuhafhaberler.com,iterative,192.35.51.30,TLD/AUTHORITATIVE,REFERRAL,169.46,-,N/A
9 2025-10-28 09:11:35,tuhafhaberler.com,iterative,170.247.170.2,ROOT,REFERRAL,268.79,-,N/A
10 2025-10-28 09:11:35,tuhafhaberler.com,iterative,192.5.6.30,TLD/AUTHORITATIVE,REFERRAL,49.93,-,N/A
11 2025-10-28 09:11:35,tuhafhaberler.com,iterative,192.26.92.30,TLD/AUTHORITATIVE,REFERRAL,173.21,-,N/A
```

We can also have some

random/leaky DNS queries from internal applications, as shown below:

```
resolver_summary.csv
1 timestamp,client,domain,result_ip,total_time_ms
2 2025-10-28 08:39:48,10.0.0.1,example.com,23.192.228.80,1228.38
3 2025-10-28 08:40:17,10.0.0.254,default.exp-tas.com,FAIL,21374.71
4 2025-10-28 08:40:39,10.0.0.254,westus-0.in.applicationinsights.azure.com,FAIL,21856.45
5
```

Now, you can use the following command to resolve the URLs from the PCAPs (Repeat Part B):

```
h1 python resolve_custom.py H1
```

where you can replace `h1/H1` with `h2/H2`, `h3/H3` and `h4/H4` respectively.

Then, you can see the logs live in the [CSV](#) files mentioned above.

| | resolver_detailed_steps.csv |
|-----|---|
| 620 | 2025-10-28 10:10:17,probigbets.com,iterative,192.45.172.30,TLD/AUTHORITATIVE,REFERRAL,1/1.20,-,N/A |
| 621 | 2025-10-28 10:10:17,probigbets.com,iterative,199.7.91.13,ROOT,REFERRAL,44.79,-,N/A |
| 622 | 2025-10-28 10:10:17,probigbets.com,iterative,192.203.230.10,ROOT,REFERRAL,45.04,-,N/A |
| 623 | 2025-10-28 10:10:17,probigbets.com,iterative,192.5.5.241,ROOT,NO_RESPONSE,timeout,-,N/A |
| 624 | 2025-10-28 10:10:17,probigbets.com,iterative,192.112.36.4,ROOT,REFERRAL,162.09,-,N/A |
| 625 | 2025-10-28 10:10:17,probigbets.com,iterative,198.97.190.53,ROOT,REFERRAL,297.16,-,N/A |
| 626 | 2025-10-28 10:10:17,probigbets.com,iterative,192.36.148.17,ROOT,REFERRAL,31.51,-,N/A |
| 627 | 2025-10-28 10:10:17,probigbets.com,iterative,192.58.128.30,ROOT,NO_RESPONSE,timeout,-,N/A |
| 628 | 2025-10-28 10:10:17,probigbets.com,iterative,193.0.14.129,ROOT,REFERRAL,53.06,-,N/A |
| 629 | 2025-10-28 10:10:17,probigbets.com,iterative,199.7.83.42,ROOT,NO_RESPONSE,timeout,-,N/A |
| 630 | 2025-10-28 10:10:17,probigbets.com,iterative,202.12.27.33,ROOT,REFERRAL,173.31,-,N/A |
| 631 | 2025-10-28 10:10:39,bikesandtransit.com,iterative,198.41.0.4,ROOT,REFERRAL,154.28,-,N/A |
| 632 | 2025-10-28 10:10:39,bikesandtransit.com,iterative,192.41.162.30,TLD/AUTHORITATIVE,REFERRAL,152.45,-,N/A |
| 633 | 2025-10-28 10:10:39,bikesandtransit.com,iterative,198.181.116.9,TLD/AUTHORITATIVE,ANSWER,43.86,351.24,N/A |
| 634 | 2025-10-28 10:10:39,aqvr.com,iterative,198.41.0.4,ROOT,REFERRAL,165.45,-,N/A |
| 635 | 2025-10-28 10:10:39,aqvr.com,iterative,192.41.162.30,TLD/AUTHORITATIVE,REFERRAL,150.78,-,N/A |
| 636 | 2025-10-28 10:10:39,aqvr.com,iterative,89.117.23.47,TLD/AUTHORITATIVE,ANSWER,294.89,611.66,N/A |
| 637 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,198.41.0.4,ROOT,REFERRAL,168.57,-,N/A |
| 638 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,193.232.128.6,TLD/AUTHORITATIVE,REFERRAL,218.01,-,N/A |
| 639 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,194.190.124.17,TLD/AUTHORITATIVE,REFERRAL,272.21,-,N/A |
| 640 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,193.232.156.17,TLD/AUTHORITATIVE,REFERRAL,309.24,-,N/A |
| 641 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,194.85.252.62,TLD/AUTHORITATIVE,REFERRAL,219.47,-,N/A |
| 642 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,193.232.142.17,TLD/AUTHORITATIVE,REFERRAL,185.06,-,N/A |
| 643 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,170.247.170.2,ROOT,REFERRAL,266.84,-,N/A |
| 644 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,192.33.4.12,ROOT,REFERRAL,97.71,-,N/A |
| 645 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,199.7.91.13,ROOT,REFERRAL,42.59,-,N/A |
| 646 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,192.203.230.10,ROOT,REFERRAL,53.48,-,N/A |
| 647 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,192.5.5.241,ROOT,NO_RESPONSE,timeout,-,N/A |
| 648 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,192.112.36.4,ROOT,REFERRAL,148.43,-,N/A |
| 649 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,198.97.190.53,ROOT,REFERRAL,297.88,-,N/A |
| 650 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,192.36.148.17,ROOT,REFERRAL,36.53,-,N/A |
| 651 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,192.58.128.30,ROOT,NO_RESPONSE,timeout,-,N/A |
| 652 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,193.0.14.129,ROOT,REFERRAL,75.63,-,N/A |
| 653 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,199.7.83.42,ROOT,NO_RESPONSE,timeout,-,N/A |
| 654 | 2025-10-28 10:10:40,niceinternetmoney.ru,iterative,202.12.27.33,ROOT,REFERRAL,169.70,-,N/A |
| 655 | 2025-10-28 10:10:51,chonvn.com,iterative,198.41.0.4,ROOT,REFERRAL,157.61,-,N/A |
| 656 | 2025-10-28 10:10:51,chonvn.com,iterative,192.41.162.30,TLD/AUTHORITATIVE,REFERRAL,151.41,-,N/A |
| 657 | 2025-10-28 10:10:51,chonvn.com,iterative,192.48.79.30,TLD/AUTHORITATIVE,REFERRAL,147.92,-,N/A |
| 658 | 2025-10-28 10:10:51,chonvn.com,iterative,192.54.112.30,TLD/AUTHORITATIVE,NO_RESPONSE,timeout,-,N/A |
| 659 | 2025-10-28 10:10:51,chonvn.com,iterative,192.31.80.30,TLD/AUTHORITATIVE,NO_RESPONSE,timeout,-,N/A |
| 660 | 2025-10-28 10:10:51,chonvn.com,iterative,192.33.14.30,TLD/AUTHORITATIVE,REFERRAL,272.96,-,N/A |
| 661 | 2025-10-28 10:10:51,chonvn.com,iterative,192.35.51.30,TLD/AUTHORITATIVE,REFERRAL,175.31,-,N/A |
| 662 | 2025-10-28 10:10:51,chonvn.com,iterative,170.247.170.2,ROOT,REFERRAL,268.77,-,N/A |
| 663 | 2025-10-28 10:10:51,chonvn.com,iterative,192.5.6.30,TLD/AUTHORITATIVE,REFERRAL,136.26,-,N/A |
| 664 | 2025-10-28 10:10:51,chonvn.com,iterative,192.26.92.30,TLD/AUTHORITATIVE,REFERRAL,173.24,-,N/A |
| 665 | 2025-10-28 10:10:51,chonvn.com,iterative,192.12.94.30,TLD/AUTHORITATIVE,REFERRAL,175.82,-,N/A |
| 666 | 2025-10-28 10:10:51,chonvn.com,iterative,192.42.93.30,TLD/AUTHORITATIVE,NO_RESPONSE,timeout,-,N/A |
| 667 | 2025-10-28 10:10:51,chonvn.com,iterative,192.33.4.12,ROOT,REFERRAL,87.82,-,N/A |
| 668 | 2025-10-28 10:10:51,chonvn.com,iterative,192.55.83.30,TLD/AUTHORITATIVE,REFERRAL,149.10,-,N/A |
| 669 | 2025-10-28 10:10:51,chonvn.com,iterative,192.52.178.30,TLD/AUTHORITATIVE,REFERRAL,168.35,-,N/A |
| 670 | 2025-10-28 10:10:51,chonvn.com,iterative,192.43.172.30,TLD/AUTHORITATIVE,REFERRAL,174.89,-,N/A |
| 671 | 2025-10-28 10:10:51,chonvn.com,iterative,199.7.91.13,ROOT,REFERRAL,42.48,-,N/A |
| 672 | 2025-10-28 10:10:51,chonvn.com,iterative,192.203.230.10,ROOT,REFERRAL,47.35,-,N/A |
| 673 | 2025-10-28 10:10:51,chonvn.com,iterative,192.5.5.241,ROOT,NO_RESPONSE,timeout,-,N/A |
| 674 | 2025-10-28 10:10:51,chonvn.com,iterative,192.112.36.4,ROOT,REFERRAL,152.33,-,N/A |
| 675 | 2025-10-28 10:10:51,chonvn.com,iterative,198.97.190.53,ROOT,REFERRAL,317.84,-,N/A |
| 676 | 2025-10-28 10:10:51,chonvn.com,iterative,192.36.148.17,ROOT,REFERRAL,47.82,-,N/A |
| 677 | 2025-10-28 10:10:51,chonvn.com,iterative,192.58.128.30,ROOT,NO_RESPONSE,timeout,-,N/A |

resolver_summary.csv

```

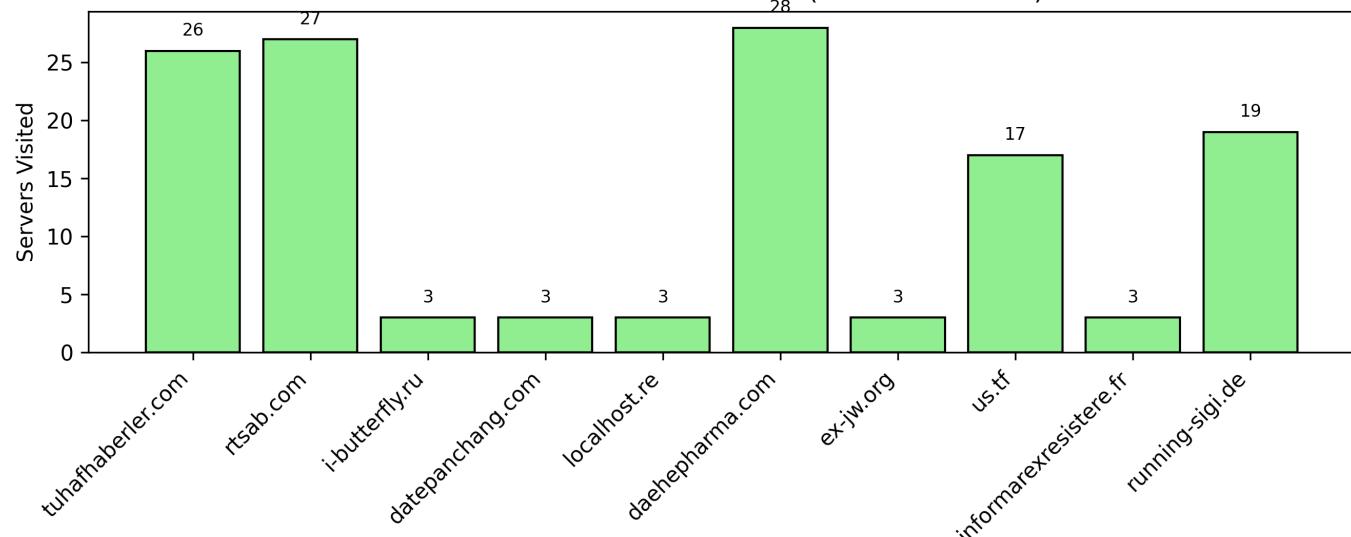
17 2025-10-28 10:05:24,10.0.0.1,ex-jw.org,185.230.63.107,3628.42
18 2025-10-28 10:05:28,10.0.0.1,us.tf,FAIL,16758.91
19 2025-10-28 10:05:45,10.0.0.1,informarexresistere.fr,178.32.143.165,982.75
20 2025-10-28 10:05:46,10.0.0.1,running-sigi.de,FAIL,11481.85
21 2025-10-28 10:05:57,10.0.0.1,buynowfromusa.com,35.223.118.1,442.46
22 2025-10-28 10:05:58,10.0.0.1,pbprofile.com,FAIL,23950.77
23 2025-10-28 10:06:22,10.0.0.1,fini.net,198.57.192.137,602.66
24 2025-10-28 10:06:22,10.0.0.1,globalpoliticalspectrum.com,FAIL,21056.41
25 2025-10-28 10:06:43,10.0.0.1,41latitude.com,FAIL,21196.87
26 2025-10-28 10:07:04,10.0.0.1,zzu.cn,FAIL,15935.90
27 2025-10-28 10:07:20,10.0.0.1,energybulbs.co.uk,23.227.38.32,1002.38
28 2025-10-28 10:07:21,10.0.0.1,o-ov.com,FAIL,24339.12
29 2025-10-28 10:07:46,10.0.0.1,lozo.com,107.21.144.47,352.33
30 2025-10-28 10:07:46,10.0.0.1,tottenhamhk.com,FAIL,21075.97
31 2025-10-28 10:08:07,10.0.0.1,newstetic.com,172.67.131.89,348.12
32 2025-10-28 10:08:08,10.0.0.1,triggerfish.se,162.159.135.42,520.60
33 2025-10-28 10:08:08,10.0.0.1,reposystems.ru,172.67.167.102,943.13
34 2025-10-28 10:08:09,10.0.0.1,interfloracentral.co.uk,217.69.32.44,994.56
35 2025-10-28 10:08:10,10.0.0.1,afairjudgement.com,172.67.178.132,354.11
36 2025-10-28 10:08:10,10.0.0.1,owlcreek.com,205.178.189.129,360.78
37 2025-10-28 10:08:11,10.0.0.1,radioterminal.ru,188.225.40.161,575.96
38 2025-10-28 10:08:11,10.0.0.1,junio.com,209.196.144.25,636.57
39 2025-10-28 10:08:12,10.0.0.1,ditrblog.com,FAIL,21098.57
40 2025-10-28 10:08:33,10.0.0.1,alchemedialtd.com,68.66.224.29,346.28
41 2025-10-28 10:08:33,10.0.0.1,portclydegeneralstore.com,198.185.159.144,496.40
42 2025-10-28 10:08:34,10.0.0.1,alenpuaca.com,67.205.28.253,614.07
43 2025-10-28 10:08:34,10.0.0.1,dj-producer-j-mbargo.com,FAIL,21103.39
44 2025-10-28 10:08:56,10.0.0.1,iag-inc.com,103.224.182.253,10155.68
45 2025-10-28 10:09:06,10.0.0.1,advertiso.ru,45.92.176.75,1360.55
46 2025-10-28 10:09:07,10.0.0.1,kh.ua,195.69.185.211,421.50
47 2025-10-28 10:09:08,10.0.0.1,kennethehartman.com,192.0.78.25,392.64
48 2025-10-28 10:09:08,10.0.0.1,hemcolubricants.com,194.9.94.85,1010.26
49 2025-10-28 10:09:09,10.0.0.1,forpsi.net,85.255.6.237,496.67
50 2025-10-28 10:09:09,10.0.0.1,memeblender.com,104.21.7.158,396.94
51 2025-10-28 10:09:10,10.0.0.1,somode.com,FAIL,21151.60
52 2025-10-28 10:09:31,10.0.0.1,gamemob.com,3.33.130.190,491.90
53 2025-10-28 10:09:31,10.0.0.1,monsterhomesecuritystore.com,FAIL,21300.61
54 2025-10-28 10:09:53,10.0.0.1,door2games.com,160.153.0.43,877.21
55 2025-10-28 10:09:54,10.0.0.1,ondemandappliancerepair.com,74.208.236.102,617.81
56 2025-10-28 10:09:54,10.0.0.1,gensee.com,211.150.89.25,633.95
57 2025-10-28 10:09:55,10.0.0.1,caricsports.com,104.21.51.12,447.28
58 2025-10-28 10:09:55,10.0.0.1,auqzzx.com,FAIL,21248.38
59 2025-10-28 10:10:17,10.0.0.1,toaskfuture.com,119.18.54.195,365.80
60 2025-10-28 10:10:17,10.0.0.1,cetsi.fr,91.151.64.24,505.34
61 2025-10-28 10:10:17,10.0.0.1,probigbets.com,FAIL,21119.44
62 2025-10-28 10:10:39,10.0.0.1,bikesandtransit.com,192.0.78.25,351.24
63 2025-10-28 10:10:39,10.0.0.1,aqvr.com,89.117.19.57,611.66
64 2025-10-28 10:10:40,10.0.0.1,niceinternetmoney.ru,FAIL,11574.82
65 2025-10-28 10:10:51,10.0.0.1,chonvn.com,FAIL,21147.91
66 2025-10-28 10:11:12,10.0.0.1,eugenbotnar.com,FAIL,21328.91
67 2025-10-28 10:11:34,10.0.0.1,phi-pe.com,FAIL,21422.88
68

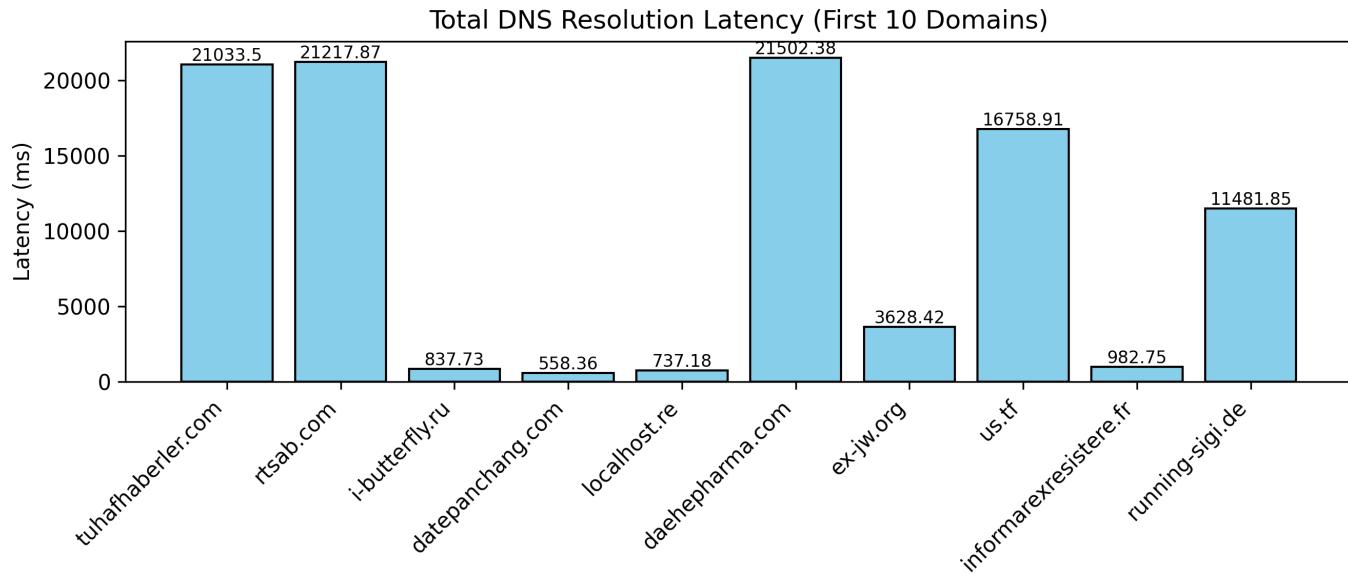
```

Plots for H1 (first 10 URLs)

Pure Iterative Server

Number of DNS Servers Visited (First 10 Domains)





F. Implement caching within your custom DNS resolver. Store recently resolved domain-to-IP mappings and serve repeated queries directly from cache without contacting external servers. Record the average lookup latency, average throughput, number of successfully resolved queries, % of queries resolved from cache and number of failed resolutions. (2.5 Points)

The DNS Resolver with Caching can be found in [custom_dns_cache.py](#)

You can see the detailed logs and overall metrics, for Custom DNS Resolver with Cache, in the folder [results_custom_cache](#).

Steps with Cache Status

| | resolver_detailed_steps.csv |
|----|---|
| 1 | timestamp, domain, resolution_mode, dns_server_ip, step, response_type, rtt_ms, total_time_ms, cache_status |
| 2 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 198.41.0.4, ROOT, REFERRAL, 174.09, -, MISS |
| 3 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.41.162.30, TLD/AUTH, REFERRAL, 157.00, -, MISS |
| 4 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.48.79.30, TLD/AUTH, REFERRAL, 177.10, -, MISS |
| 5 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.54.112.30, TLD/AUTH, NO_RESPONSE, timeout, -, MISS |
| 6 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.31.80.30, TLD/AUTH, NO_RESPONSE, timeout, -, MISS |
| 7 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.33.14.30, TLD/AUTH, REFERRAL, 279.16, -, MISS |
| 8 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.35.51.30, TLD/AUTH, REFERRAL, 174.64, -, MISS |
| 9 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 170.247.170.2, ROOT, REFERRAL, 276.28, -, MISS |
| 10 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.5.6.30, TLD/AUTH, REFERRAL, 146.81, -, MISS |
| 11 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.26.92.30, TLD/AUTH, REFERRAL, 181.59, -, MISS |
| 12 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.12.94.30, TLD/AUTH, REFERRAL, 177.66, -, MISS |
| 13 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.42.93.30, TLD/AUTH, NO_RESPONSE, timeout, -, MISS |
| 14 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.33.4.12, ROOT, REFERRAL, 93.50, -, MISS |
| 15 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.55.83.30, TLD/AUTH, REFERRAL, 175.07, -, MISS |
| 16 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.52.178.30, TLD/AUTH, REFERRAL, 153.75, -, MISS |
| 17 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.43.172.30, TLD/AUTH, REFERRAL, 177.96, -, MISS |
| 18 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 199.7.91.13, ROOT, REFERRAL, 57.25, -, MISS |
| 19 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.203.230.10, ROOT, REFERRAL, 57.90, -, MISS |
| 20 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.5.5.241, ROOT, NO_RESPONSE, timeout, -, MISS |
| 21 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.112.36.4, ROOT, REFERRAL, 168.52, -, MISS |
| 22 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 198.97.190.53, ROOT, REFERRAL, 311.99, -, MISS |
| 23 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.36.148.17, ROOT, REFERRAL, 47.45, -, MISS |
| 24 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 192.58.128.30, ROOT, NO_RESPONSE, timeout, -, MISS |
| 25 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 193.0.14.129, ROOT, REFERRAL, 63.53, -, MISS |
| 26 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 199.7.83.42, ROOT, NO_RESPONSE, timeout, -, MISS |
| 27 | 2025-10-28 11:02:14, tuhafhaberler.com, iterative, 202.12.27.33, ROOT, REFERRAL, 173.42, -, MISS |
| 28 | 2025-10-28 11:02:35, rtsab.com, iterative, 198.41.0.4, ROOT, REFERRAL, 166.35, -, MISS |