

**Technical Explanation:** The DNS structure that I created had a header and a question, and additionally the header had 6 different variables and the question had 3 different variables under it. This was chosen because during my research on how a DNS packet should look like, I used multiple resources such as GeeksForGeeks, MisLove, and DNS-Wizardzines to decide on the structure of my packet as well as the format. The header of the DNS packet contained identification, flags, number of questions, answers, authorities, and additionals, but since this was a query, the latter 3 were set to 0 and ignored, same as the flags. While the question part of the DNS packet contained the domain name, the DNS type and the DNS class. That's how the DNS packet is structured in my case. The main reason why it was constructed this way was because the query of a DNS didn't need flags, answers, authorities, and additionals, I chose to set them to 0 and since the number of questions were always going to be 1, I hardcoded the value to 1 as well. Furthermore, another reason why my packet was constructed this way is because I knew that the query class was always going to be an 'A', meaning name to address, so I also hardcoded the type variable in the question header to 1 while the others would be variable in the question part of the DNS packet. These are my hows and whys for the way the DNS packet is structured.

Each response was parsed using the "dnslib" external python library. Then, I looked for the IP addresses under the additional section from the DNS record and used those IP addresses to iterate through each DNS server until I reached "wikipedia.org". This external python library was only used for parsing and not in the construction of the DNS packet itself.

For the HTTP request, it was created via a TCP socket which was established through the three way handshake. After that, we sent the request to the wikipedia DNS IP address as well as the HTTP port. Then the response was a 301 maintenance code, saying that the site has been

migrated/moved. I then proceeded to try it with Google.com as well, the source code for google will be listed under “part1” labeled as test.py. But it too popped up with the 301 maintenance, which makes sense for both cases since HTTPS is the main request that is used by the modern internet thus seeing the HTTP request message telling us that the sites are under maintenance or have migrated checks out.

**External Resources Cited [ACM Rules]:** The sources that I used were [https://implement-dns.wizardzines.com/book/part\\_1](https://implement-dns.wizardzines.com/book/part_1), <https://mislove.org/teaching/cs4700/spring11/handouts/project1-primer.pdf>, <https://www.geeksforgeeks.org/dns-message-format/>, <https://realpython.com/python-sockets>, and <https://www.internalpointers.com/post/making-http-requests-sockets-python>.

**Results Explanation:** The RTTs from the DNS servers were calculated using the datetime module from Python and then each request and response to and from the DNS servers were recorded. Meaning each individual request/response had a RTT, then those were all added up in total to get the total RTT that it took to request/response to and from the DNS servers. The RTT for wikipedia.org used a very similar method from above with the DNS server, where we recorded the start time for the request portion and recorded the end time for the response portion. Subtracting the end time by the start time yields the RTT for the HTTP request. Additionally, the RTT is included for the HTTP request for google.com as well.

**Output Screenshot:**

```
PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE Python + - [icon] ... ^ X
```

```
PS C:\EEC 173A Projects> & C:/Users/herob/AppData/Local/Microsoft/WindowsApps/python3.13.exe "c:/EEC 173A Projects/Project #02/part1/DNS_client_[Tyson-Tien Nguyen].py"
DNS Query Built: b'D\xcb\x00\x00\x01\x00\x00\x00\x00\x00\wikipedia\x03org\x00\x00\x01\x00\x01'.
```

```
DNS Query sent to DNS Public Resolver at ('198.41.0.4', 53).
Received response from DNS Public Resolver ('198.41.0.4', 53).
RTT: 0:00:00.024033 in seconds.
Unpacking response from DNS Public Resolver ('198.41.0.4', 53).
Extracting DNS Records from DNS Public Resolver ('198.41.0.4', 53).
Parsed DNS Records, TLD DNS server found. TLD DNS server is at ('199.249.112.1', 53) and DNS type is A.
```

```
Sending out DNS Query to TLD DNS server ('199.249.112.1', 53).
Received response from TLD DNS server ('199.249.112.1', 53).
RTT: 0:00:00.025760 in seconds.
Unpacking response from TLD DNS server ('199.249.112.1', 53).
Extracting DNS Records from TLD DNS server ('199.249.112.1', 53).
Parsed DNS Records, Authoritative DNS server found. Authoritative DNS server is at ('208.80.154.238', 53) and DNS type is A.
```

```
Sending out DNS Query to Authoritative DNS server ('208.80.154.238', 53).
Received response from Authoritative DNS Server ('208.80.154.238', 53).
RTT: 0:00:00.085984 in seconds.
Unpacking response from Authoritative DNS server ('208.80.154.238', 53).
Extracting DNS Records from Authoritative DNS server ('208.80.154.238', 53).
Parsed DNS Records, "wikipedia.org" server found. "wikipedia.org" server is at ('198.35.26.96', 53) and DNS type is A.
```

```
Total RTT: 0:00:00.135777 in seconds.
```

```
RTT: 0:00:00.023912 in seconds.
HTTP/1.1 301 Moved Permanently
content-length: 0
location: https://198.35.26.96/
server: HAProxy
x-cache: cp4044 int
x-cache-status: int-tls
connection: close
```

```
PS C:\EEC 173A Projects>
```

**Figure 1:** Wikipedia.org

```

PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE
PS C:\EEC 173A Projects> & C:\Users\herob\AppData\Local\Microsoft\WindowsApps\python3.13.exe "c:/EEC 173A Projects/Project #02/part1/test.py"
DNS Query Built: b'D\xcb\x00\x00\x00\x01\x00\x00\x00\x00\x06google\x03com\x00\x00\x01\x00\x01'

DNS Query sent to DNS Public Resolver at ('198.41.0.4', 53).
Received response from DNS Public Resolver ('198.41.0.4', 53).
RTT: 0:00:00.027103 in seconds.
Unpacking response from DNS Public Resolver ('198.41.0.4', 53).
Extracting DNS Records from DNS Public Resolver ('198.41.0.4', 53).
Parsed DNS Records, TLD DNS server found. TLD DNS server is at ('192.41.162.30', 53) and DNS type is A.

Sending out DNS Query to TLD DNS server ('192.41.162.30', 53).
Received response from TLD DNS server ('192.41.162.30', 53).
RTT: 0:00:00.028864 in seconds.
Unpacking response from TLD DNS server ('192.41.162.30', 53).
Extracting DNS Records from TLD DNS server ('192.41.162.30', 53).
Parsed DNS Records, Authoritative DNS server found. Authoritative DNS server is at ('216.239.34.10', 53) and DNS type is A.

Sending out DNS Query to Authoritative DNS server ('216.239.34.10', 53).
Received response from Authoritative DNS Server ('216.239.34.10', 53).
RTT: 0:00:00.035441 in seconds.
Unpacking response from Authoritative DNS server ('216.239.34.10', 53).
Extracting DNS Records from Authoritative DNS server ('216.239.34.10', 53).
Parsed DNS Records, "google.com" server found. "google.com" server is at ('142.251.214.142', 53) and DNS type is A.

Total RTT: 0:00:00.091408 in seconds.

RTT: 0:00:00.034743 in seconds.

HTTP/1.1 301 Moved Permanently
Location: http://www.google.com/
Content-type: text/html; charset=UTF-8
Content-Security-Policy: Report-Only: object-src 'none';base-uri 'self';script-src 'nonce-6MENS3FS6bl_DI_qIawXg' 'strict-dynamic' 'report-sample' 'unsafe-eval' 'unsafe-inline' https://http://report-uri https://csp.withgoogle.com/csp/gcs/other-hp
Date: Mon, 17 Mar 2025 00:04:33 GMT
Expires: Wed, 16 Apr 2025 00:04:33 GMT
Cache-Control: public, max-age=2592000
Server: gws
Content-Length: 219
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN

```

**Figure 2:** Google.com Part 1

```
X-Frame-Options: SAMEORIGIN

<HTML><HEAD><meta http-equiv="content-type" content="text/html; charset=utf-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="http://www.google.com/">here</A>.
</BODY></HTML>

PS C:\EEC 173A Projects>
```

Ln 112, Col 55 Spaces: 4 UTF-8 CRLF Python 3.13.2 64-bit (Microsoft Store)

**Figure 3:** Google.com Part 2