N26 SRE Challenge

DNS to DNS-over-TLS proxy

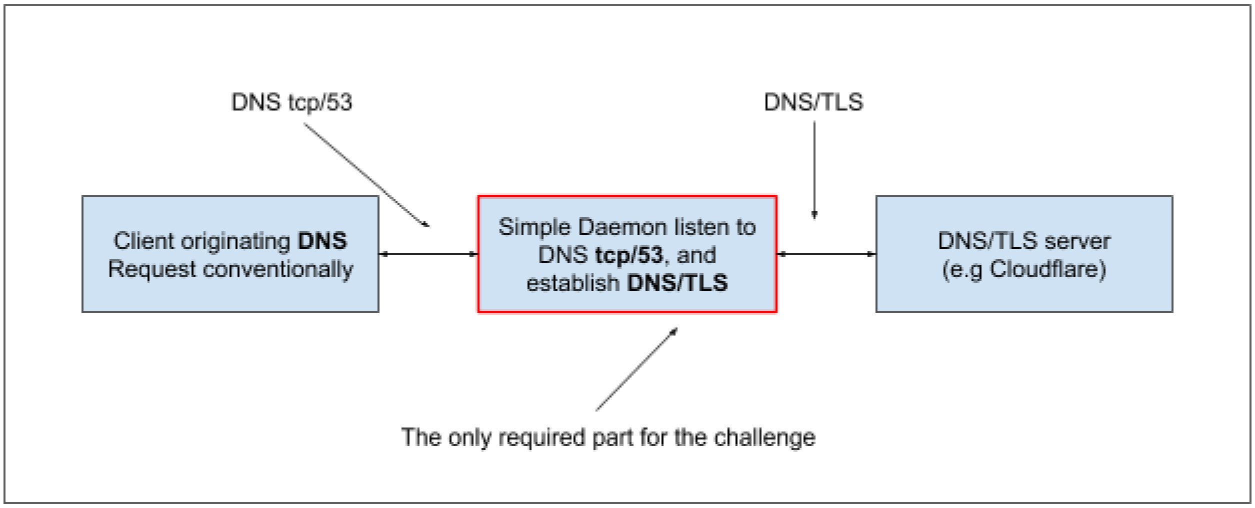
The following challenge is to write a DNS proxy that listens to conventional DNS and sends it over TLS (***detailed requirements below***).

Background

Nowadays, some providers (such as Cloudflare) provide a **DNS-over-TLS** feature that could let us enhance privacy by encrypting our DNS queries.

Our applications don't handle DNS-over-TLS by default. Your task is to design and create a simple **DNS to DNS-over-TLS proxy** that we could use to enable our application to query a DNS-over-TLS server.

Overview


DNS tcp/53
DNS/TLS
Client originating DNS Request conventionally
Simple Daemon listen to DNS tcp/53, and establish DNS/TLS
DNS/TLS server (e.g Cloudflare)
The only required part for the challenge

Requirements

From your understanding of the topic, we would expect a working example of a **DNS** to **DNS-over-TLS** proxy that can:

1. Handle **at least one DNS query**, and give a result to the client.
2. Work over **TCP** and talks to a DNS-over-TLS server that works over TCP (e.g: Cloudflare).

Deliverables

1. The **source code**.
2. A **Dockerfile**, and the different options required to run your software.
3. A **README.md** file detailing your implementation, your choices, and answering the following questions:
    - ❏ What are the security concerns for this kind of service?
    - ❏ Considering a microservice architecture; how would you see this the dns to dns-over-tls proxy used?
    - ❏ What other improvements do you think would be interesting to add to the project?

Note that there is **no requirement regarding the language** you use, but you **are** required to write the code yourself! This means that you **cannot use already written software** like this or this similar project.

Challenge time should be around **3.5 hours.**

Resources

Here is some helpful documentation:

- Cloudflare's explanation of DNS-over-TLS; you can use Cloudflare's **DNS-over-TLS** feature to test your queries.
- Feel free to look into these RFCs:
    - DNS (especially UDP vs. TCP implementations)
    - DNS-over-tls
- Use any resource available to you online, but you must **write the code yourself**.
- Use any language you see fit, and any libraries to leverage code already written by modules and packages.
- If needed, here is some links to libraries regarding implementations
    - Golang: tls, net (udp/tcp lib)
    - Python: ssl, socket (udp/tcp)

Bonus points

If you think the challenge was a little too easy for you, we'd love to see you give these features a try:

- Allow multiple incoming requests at the same time
- Also handle UDP requests, while still querying tcp on the other side
- Any other improvements you can think of!

If you have questions, feel free to contact us at: **sre-challenge@n26.com**.