CSE Technical Project

Thank you for taking the time to speak with us. We have enjoyed the interviews so far!

Our technical homework is intended to evaluate your hands-on capabilities, give you a taste for the work we do, and show us how you communicate complex subjects in writing.

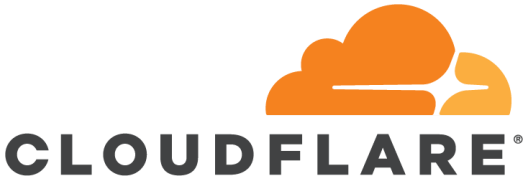
First steps:

1. Please add a domain (new or existing) to Cloudflare. Cloudflare Free plan is sufficient.
2. Activate on Cloudflare by following the steps to change the nameservers at your registrar.

Please keep in mind that while we can help point you in the right direction, this assignment is meant to evaluate your ability to learn new concepts quickly through effective research & self-discovery. The Cloudflare offering is vast, and it's crucial that you are comfortable researching answers on topics you may not be familiar with.

Additional considerations:

- Keep the audience in mind: your reviewers are technical experts on Cloudflare, but also want to understand how you can communicate complex topics to non-technical stakeholders.
- Technical deliverables are largely mutually exclusive. If you have trouble with any of the specific tasks this should not deter or prevent you from completing the remainder of the tasks.
- You will need to be ready to discuss your homework in the next round of interviews.



Steps:

1. Create an origin web server on a platform of your choosing. This could be in AWS, Google Cloud, DigitalOcean, your Raspberry Pi, etc.

   This web server must run an endpoint that returns all HTTP request headers in the body of the HTTP response.

   The web server can be something that you have written yourself (e.g. in JavaScript, Python, etc) or by using a 3rd party application such as HTTPBin. Proxy traffic to this server through Cloudflare.

2. Secure the communication between Cloudflare and your Origin Server with a non-Cloudflare provisioned TLS certificate using at least the Full-Strict mode on Cloudflare.

3. Install and configure Cloudflare Tunnel on your origin server using a subdomain called "tunnel", e.g. tunnel.yourwebsite.com. Make connections proxied to your server protected using this tunnel.

4. Write an API call that outputs all of your DNS records, using an API scoped token. Include the token permission scope, API call and its output.

5. Create a Cloudflare Worker. The HTTP response body should be "This is your ${CLIENT_IP} and you are accessing this site from ${COUNTRY) | ${ASN}.

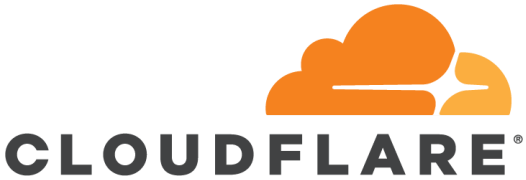
   a. This response should be visible on the browser as HTML. Run this worker script on the /geo path. For example: www.yourwebsite.com/geo.

   b. Use Workers to create a logic if a user who is not from Singapore to be redirected to https://1.1.1.1/.

   c. Create this worker using the Wrangler CLI, upload your Workers code to a public Git repository for your implementation.

6. Lock down access for a particular path for your Cloudflare Tunnel subdomain (e.g. tunnel.yourwebsite.com/secure) and only allow access for a particular user or a group of users using **Cloudflare Zero Trust**.

   a. Ensure nobody can bypass Cloudflare and access your server's IP directly.



Deliverables:

1. A working application with instructions needed to access any of the above.

2. A report that addresses the following:
    - A report detailing how you implemented the technical requirements step by step.
    - What use cases can you see different products being useful for?
    - How did you fill the gaps (if any) in your knowledge during the process?
    - How do you imagine that a target customer will find this experience?

Please let us know if you have any questions with any of the above!