📋 Recruitment Technical Test – Cloud Platform Engineer

Overview

Hello there! This challenge is a chance for you to showcase your skills and, more importantly, how you think about building robust solutions in the cloud. We're excited to see your approach.

This isn't about getting a "right" answer, but about seeing how you tackle problems. We'd love to see how you go about:

- Building a serverless function and automating its deployment.
- Writing clean, secure, and effective Terraform code.
- Designing a modern, production-ready web hosting solution on AWS.
- Communicate complex technical ideas with clarity, tailored to the audience.

A quick note on time: We suggest setting aside around 3 hours for this. Please don’t feel pressured to produce a flawless production implementation — what we value is your reasoning, trade-offs, and creative approach. Think of it as a normal peer-review, not a pitch to investors.

Submission

Please ensure you submit your test answers back to us **at least 1 working day prior** to your interview.

Part 1: Lambda + Pipeline

Let's get hands-on! Your first task is to build a small serverless function and hook it up to a CI/CD pipeline.

The Goal:

1. **Write a simple AWS Lambda function** (Python or TypeScript, your choice!) that uses an AWS API to grab some information.
2. Have the function do something meaningful with that data, such as persisting it or making it queryable.
3. Create a user-friendly endpoint so the function can be triggered by anyone, even without direct AWS access.
4. Define all the necessary AWS infrastructure using **Terraform**.

To get your creativity flowing, here are a few ideas. Feel free to pick one of these or come up with your own unique solution!

- **EC2 Inspector:** List all running EC2 instances and save the list to a DynamoDB table.

- **S3 Counter:** Count the objects in a specific S3 bucket and send that count as a custom CloudWatch Metric.
- **CloudWatch Detective:** Query recent error logs and create a neat summary file.
- **IAM Auditor:** List all IAM users and their MFA status, then pop that info into DynamoDB.

Automate It All:

Next, please set up a **CI/CD pipeline** (we would prefer GitHub Actions as that's what we use here) that automatically:

- Runs linters and tests
- Deploys your Lambda function and all its Terraform-managed resources.

Heads-up: We don't expect you to deploy this for us — but we do expect code we can review together in your interview, and that you can talk us through your design choices.

Part 2: AWS Web Hosting Design & Presentation

Now, let's zoom out and think about the big picture. For this part, we'd like you to prepare a short design presentation outlining how you'd host a modern, production-ready web application on AWS.

Imagine you're designing the infrastructure from scratch. We'd love to hear your thoughts on the following areas. You don't need to be an expert in everything, but showing your general understanding of these concepts is what counts:

- **Compute:** What's your tool of choice? ECS, EKS, Lambda, EC2? Why?
- **Networking:** How would you handle DNS, networking, and SSL/TLS certificates?
- **Speed & Caching:** What's your game plan for using a CDN like CloudFront?
- **Scaling:** How will your app handle traffic spikes gracefully?
- **Observability:** How will you keep an eye on everything with logging and monitoring?
- **The Essentials:** And of course, how do you factor in cost, resilience, and security from day one?

Crafting Your Presentation:

In your presentation, we'd like to see:

- An **Executive Summary**
- An **Architecture Overview**
- **Challenges & Risks:** A brief, honest view of technical difficulties and risks for delivery.

👉 **What are we looking for?**

- Executive summary that is suitable for senior, non-technical stakeholders.
- Technical content is clear, structured, and accurate.
- Ability to balance business and technical perspectives.

- Awareness of trade-offs, not just a “happy path” design.

We look forward to seeing how you approach these challenges. During the interview, we’ll explore your technical reasoning and the choices you made — much like a design review session.