Contents

Seasats – Senior Network & Infrastructure Engineer Take-Home Test (48 Hours) 1
Task 1 — Public REST API 1
Task 2 — Secure Endpoint via VPN 2
Task 3 — Public Webpage Plotting Hit Frequency (Status + Secure-Status) 2
Submission Requirements 2

Seasats – Senior Network & Infrastructure Engineer Take-Home Test (48 Hours)

Below is a 48-hour take-home practical test. The goal is to assess your ability to build, deploy, secure, and document small production-like services with minimal tooling, this is a microcosm of real Seasats tasks and is simplified for ease of interviews and to focus on execution speed.

Please include a short README with all deliverables.

Task 1 — Public REST API

Stand up a **publicly accessible REST API** using any language or framework.

Requirements:

- Implement an endpoint: **/status**
    - Returns JSON containing:
        - An incrementing integer (total number of times /status has been hit).
        - A random number between **1 and 10**.
- The API must be reachable from a **public IP or domain**.

Task 2 — Secure Endpoint via VPN

Add a second endpoint: /secure-status

Requirements:

- Must only be reachable when connected to a VPN server you deploy
- /secure-status must return the same JSON structure as /status.
- Deploy a VPN server of your choice
- Provide:
    - A working client configuration file
    - The required client certificate so that we can reach the secure-status

Task 3 — Public Webpage Plotting Hit Frequency (Status + Secure-Status)

Create a **publicly accessible webpage** that visualizes how often each endpoint has been hit over time.

Requirements:

- The webpage may be simple/minimal.
- It must:
    - Plot the hit count for **/status** over time.
    - Plot the hit count for **/secure-status** over time **when the viewer is on the VPN**.
- All/any off the shelf tools or libraries are allowed

Submission Requirements

Provide the following:

1. **Public URLs** for:
    - The API
    - The webpage
2. **VPN client config & certificate**
3. A short **README** containing:
    - Tech stack used
    - Setup + testing instructions
    - Short list of assumptions made
    - What you would improve with more time