

Subject: Take Home Test from Censys
From: Jaymie Anderson <jaymie@censys.com>
Date: 10/14/25, 12:42 PM
To: katie@krupczak.org

Hi Kathryn,

Thank you for taking the time to speak with Advaith. He would love to move you forward to the next step for the **Software Engineer** opportunity at Censys. As part of your interview, you are asked to complete the following assessment:

Project Description

In security and network monitoring, understanding change over time is critical. A port that was closed yesterday might be open today; a new vulnerability might appear on a service that was previously considered safe.

Your task is to build a minimal **Host Diff Tool**:

- **Ingest snapshots of a host at different points in time.**
- **Compare any two snapshots of the same host** to identify what changed.
- **Provide a simple UI** for uploading snapshots, viewing history, and running comparisons.
- Generate a structured **diff report** that highlights meaningful changes (e.g. ports, services, vulnerabilities, or version info).

This exercise tests how you structure a project, handle data, and present results clearly — while staying within a realistic 4-hour window.

Requirements

- The project should demonstrate both **frontend and backend components**.
- The system should allow a user to:
 - Upload a host snapshot JSON file.
 - See a history of snapshots for a given host.
 - Select two snapshots and view what has changed between them.
- The application should **store snapshots** so that they can be retrieved later.
- The application should **handle errors gracefully** (e.g. bad input, duplicates, or system failures) without crashing.
- Provide a **simple web interface** to interact with the system. The design doesn't need to be polished, just functional.
- Use **Golang** (or your language of choice) for the backend implementation. The rest is up to you.

Deliverables

1. **Code Submission:** All source code should be uploaded to a publicly accessible GitHub or Bitbucket account.
2. **README File:** A comprehensive README.md file should be included in the repository with the following:
 - Instructions on how to run the project.
 - Any assumptions made during development.
 - Simple testing instructions (manual or automated).
 - A brief description of the implemented AI techniques.

3. **Future Enhancements List:** A list detailing what you would do if given more time to work on this project.
4. **Containerization:** The entire application (backend, frontend, any chosen data store or other resources) must be runnable with a single ``docker compose up`` command. Please include the necessary ``Dockerfile(s)`` and a ``docker-compose.yml`` file.

Evaluation Criteria

We will be assessing the following:

- **Core Functionality:** Does the application meet the requirements (upload, store, list, and compare snapshots with a usable UI)?
- **Correctness & Reliability:** Does the system behave as expected, including handling bad inputs or errors gracefully?
- **Design Decisions:** Are the choices around data storage, APIs, and UI sensible and well-reasoned for the problem?
- **Code Quality:** Is the code organized, readable, and maintainable?
- **Testing:** Are there minimal tests in place (especially for critical logic like comparisons)?
- **Documentation:** Does the README clearly explain how to run the project, what assumptions were made, and what could be improved?
- **User Experience:** Is the UI simple and functional, making it easy to complete the required tasks?
- **Extensibility:** Is the project structured in a way that new features could be added without major rework?

Suggested AI Tools

- Cursor
- CoPilot
- ChatGPT
- Claude
- Any other AI driven development tool you'd like to use.

Please note: *Even though we encourage AI assisted tools, we expect that the you understand and can speak to everything in your submitted solution.*

Time Commitment

We expect this project to take approximately 4 hours to complete. We understand that you may have other commitments, so if you anticipate needing more than 5 business days to complete the project, please inform us of your expected submission date.

Resources

Censys Interview data set: Attached

The attached zip file contains several host files. These files contain real Censys host data, but any sensitive information has been replaced with fictional references. You can use these snapshots for testing your solution.

Each file name is formatted as such: ``host_<ip>_<timestamp>.json``

IP is the host's address (e.g., 203.0.113.10).

Timestamp indicates when the snapshot was taken (ISO-8601, with colons replaced by dashes for filenames).

Example: host_125.199.235.74_2025-09-10T03-00-00Z.json

There are **three different hosts**, each with multiple snapshots taken at different times. Together they illustrate:

- Ports being opened or closed.
- Services being added, removed, or updated.
- Vulnerabilities (CVEs) appearing or being remediated.
- Software versions changing.

You can use these files to test your solution by uploading them, browsing snapshot history, and generating diffs.

Censys Data definitions to learn more about hosts and their structure: <https://docs.censys.com/docs/platform-host-dataset#/>

Clarification and Support

Please do not hesitate to ask any questions for clarification. We are here to support you throughout this process.

Best,

Jaymie Anderson

Please submit here:

https://app2.greenhouse.io/tests/f64c7581ee2794df3ce53de8989f8d42?utm_medium=email&utm_source=TakeHomeTest

— Attachments: —

host_snapshots-20250919T132548Z-1-001_(1).zip

5.9 KB