# Clojure Homework Questions

## Build a RESTful Microservice

| File | Modified |
|------|----------|
| indicators.json | yesterday at 8:30 PM by Jared Holmberg |

Drag and drop to upload or **browse for files**

## Requirements

Your objective is to build a simple JSON based REST Microservice. The service will provide search capabilities on an open source intelligence feed provided by AlienVault OTX. REST and microservices are fairly common, the goal here is to show how this might be done in a Clojure application using it's unique functional capabilities. Since most production level applications require maintaining state in one way or another, the component framework is a common library used to help manage state in a systematic way.

1. Create a simple REST JSON MicroService using Pedestal and Component
   a. http://pedestal.io/
   b. https://github.com/stuartsierra/component
2. The project should be built in either lein or tools.build
3. Create a REST endpoint component that returns a document by its ID.
   a. GET /indicators/:id
4. Add endpoints to return all documents or to find a document by its type.
   a. GET /indicators
   b. GET /indicators?type=IPv4
5. There should be a component that does a lookup on a JSON file (array of documents).
   a. Treat the JSON file like a database with many records (array of documents). Attached is a JSON file you can use which contains 100 Indicators of Compromise (IOCs) from AlienVault OTX.
6. Tests are important. In fact, writing tests first is recommended.
   a. We are continually working to improve our Continuous Integration and Continuous Delivery pipelines to ensure high code quality.
   b. Show us what you think would be good unit level and integration level tests.
7. The application should be able to compile and be packaged up in a Docker container. Provide a valid Docker file in the project that could be used to build the image.
8. Plagiarism will disqualify you.
   a. We're not saying you can't use libraries. Please do! It shows us you know how to integrate systems together. You don't need to re-implement hash maps, regular expresses, or array. But don't take someone else's web service implementation and call it yours.
   b. If you read any articles or take any tips from any site, tutorial, or blog, **Cite them**. Put them in a Readme.md or references.txt file to show what research you did.
9. You can either send us an archive of the project or share it via Github. You are welcome to email us with any questions you have.

**Bonus Points**

1. Use a monorepo design with https://polylith.gitbook.io/polylith/ . Note, if you choose polylith, it's not required to use pedestal. Standard ring handlers is fine.
2. Add an endpoint that takes fields to search by that exist in the attached indicators.json file.
   a. POST /indicators/search
      i. The POST endpoint should be able to take search criteria as a JSON document in the body of the HTTP request. For example, how could the endpoint search for a specific indicator or author?

## Assumptions

1. This service is not expected to be ready to scale to petabytes of information. It should be able to handle the attached data set without any issues with system memory or JVM heap.

2. Assume we won't spend copious amounts of time trying to figure out how to use your code. If you've used lein, provide a shell script that can build the jar file and put it in the right place so the Dockerfile can produce an image. Same goes for tools.build.

The rest is up to you and your creative mind! This project is open ended on purpose. Every applicant has different time capacities and experiences. We want this to be attainable based on what you know and have time to learn. Want to use GraphQL instead? Great. Want to import this into a database? Awesome! But make sure it's baked into a system that is easy enough for us to use with either intuition or good instructions. It's up to you!

Engineering is a job where there are always a thousand different areas to spend your time in but you have to make sacrificial choices on which areas will provide the most value for the outcome desired.

Your outcome: ***Display your talents and expertise in a way that we want you to join our team.***

All of the grading criteria is below is important. But how will you balance your time to showcase your skills and display what you think is the most valuable is the key.

Here is how we will grade your submission:

1. Where do we perceive the engineer spent most of their time and energy? Scale (Least)1 - - - 5(Most)
2. What is the quality of that time spent in each of the following areas? Scale (Bad)1 - - - 5(Good)

| Application Aspect | Perceived Time | Quality |
| --- | --- | --- |
| Feature Rich | | |
| Optimizations | | |
| Algorithm Design | | |
| Code Clarity and Simplicity | | |
| Documentation | | |
| Tests | | |
| Usability | | |

You should aim to have no 1's but don't stress yourself out to have all 5's.

Feel free to ask any questions, design, clarifying requirements, asking about features, etc.

jholmberg@centripetalnetworks.com

pmallett@centripetalnetworks.com

tcamechis@centripetalnetworks.com

twendell@centripetalnetworks.com