Interview: Full-Stack Python Developer on Linux

This project is an interview for your chance to join a top employer on Upwork. We have a number of on-going projects that need developers. Specifically, this project is for Python Developers capable of handling the full Linux stack, including devops work and frontend frameworks. For this project, you will build a simple web service that uses GPG to decrypt a message. Your successful completion of this project will start you on a trial period with our team of up to 45 days.

Priority will be given to applicants with experience in any of the following: React, React Native, Angular, Django, .NET Core, Elasticsearch.

NOTE: This project description is a lot to read and we don't want to waste anyone's time. We employ developers from all corners of the Earth and at a variety of rates. Many of our developers start at an hourly rate in the $10 to $14 range. If you are requesting a rate above this range, you can send a message stating that you would like to determine your hourly rate first so that we don't waste your time.

It is imperative that you READ THIS ENTIRE POST before sending any messages or placing any bids.

The entirety of this agreement is in the attached PythonWebService.pdf file. The first 3,000 characters are included below for reference.

Before you do any work on this project, send us a proposal with your lowest acceptable hourly rate. We will have a few questions for you. If things look good, we'll ask you to complete the interview project.

Your Task:

- Build a RESTful API with a single endpoint using the Python framework or packages of your choice. The endpoint only accepts POST requests and is responsible for decrypting a message from the request's payload.
- Use Docker to bundle your app, include a `Dockerfile` that will build your app as a Docker image. We would appreciate if your Dockerfile is based on one of the Python official docker images: https://hub.docker.com/_/python.
- Your app must be served via a wsgi handler. Like gunicorn, uWSGI, Apache (mod_wsgi), etc unless the chosen framework has its own event loop or built-in production server (like Tornado). Do NOT serve the app using the framework's development server (like Django's `python manage.py runserver` or Flask's `flask run`).

- Include a `run_tests.sh` file that will be responsible for running tests from within the Docker image itself and/or can be invoked with `docker run YOUR_IMAGE run_tests.sh`.

Deliverable:

1. Your deliverable must be a single file named `app.tar.gz` without any parent/root directory within (subfolders are fine).

2. Our automated build system will extract your deliverable file and build your Docker image with `docker build -t IMAGE_NAME .`. Make sure your `Dockerfile` is at the tarball's root.

3. Our automated build system will run your docker image with `docker run -d -p 80:80 IMAGE_NAME`.

4. Your deliverable is only correct if the following is true after our automated build system runs:

a. A web service is available via the server's public IP (e.g. http://51.11.132.197/decryptMessage).

i. The web service is built using Python. You may use any packages or frameworks you prefer.

ii. The web service accepts a JSON payload with the following parameters:

1. `passphrase` : The passphrase to use to decrypt the message.

2. `message` : The GPG encrypted message.

iii. The web service can be executed using an HTTP POST only.

iv. The web service returns either an HTTP error (if bad input parameters are given) or a single response parameter (if good input parameters are given):

1. The response must be JSON in the form of `{"DecryptedMessage": "The given message, decrypted using GPG and the given passphrase"}`.

b. The web service has unit tests that you believe are sufficient.

i. All available unit tests can be executed by running `docker run -it --rm IMAGE_NAME run_tests.sh`.

ii. After running, details about the tests are output (pass, fail, specific failure messages).

5. Your deliverable must be your original work.

6. Your deliverable will be tested with an encrypted message similar to the following (the passphrase for this sample message is "topsecret"):

-----BEGIN PGP MESSAGE-----
Version: GnuPG v2

jA0ECQMCVady3RUyJw3X0kcBF+zdkfZOMhISoYBRwR3uk3vNv+TEg+rJnp4/yYIS
pEoI2S82cDiCNBIVAYWB8WKPtH2R2YSussKhpSJ4mFgqyOA01uwroA==
=KvJQ
-----END PGP MESSAGE-----

We will use a curl command similar to this:

```
curl -vX POST http://SOME_IP/decryptMessage -d @payload.json --header
"Content-Type: application/json"
```

Rules:

1. Only bid on this project if you have 20 or more hours available a week. We have significant amounts of work and need developers who are doing this as more than a hobby.

2. Only bid on this project if you are an individual working by yourself. We are not interested in bids from companies. We need to know the work we award in the future will be performed by the same person who completes this project.

3. Only bid on this project if you have never bid on any other project for this employer.

4. Only bid on this project if you can successfully complete this site's identity verification process. We are a U.S. company and therefore our government does not allow us to conduct business with, and we will not accept, workers who reside in or perform work from within the following countries: Cuba, Iran, North Korea, Sudan, Syria, or Burma.

5. On future projects, you will be required to install time tracking software from Hubstaff.com (similar to this site's time tracking). This software will take screenshots while you are "on the clock." You will only be paid for hours tracked through Hubstaff.com.

6. We will use the message board for the contract bidding process to facilitate the interview project. We will not accept the bid because this is an unpaid interview project. If you are successful, we will start a new project contract specifically for you.

7. Before starting the project, you may ask any questions you like and we will respond as best we can. We will not answer questions regarding the deliverable or how to go about your work. All the information you need to successfully complete the project is in this description.

8. The project length is a maximum of five days. This is your due date. Your correctly working deliverable must be uploaded no later than the due date.

How To Start:

1. Your bid must include a message with this exact phrase as the first line, "I have read PythonWebService.pdf in its entirety and agree to its terms."

2. This is an unpaid interview project. We believe this is best for all parties. Successfully completing this project requires perfect execution and only ~10% of applicants meet that expectation. For the 90% who are not successful, we don't want to leave negative feedback, which would be required of us if it were paid. Further, typical in-office interviews can take an entire day, sometimes multiple days. This test takes successful candidates 4-6 hours, which we believe is comparable to an in-office interview. We're not asking for more time than you would devote to any other serious interview offer so we think an unpaid interview project is fair.

We look forward to your work!