Flask REST API using JWT

or

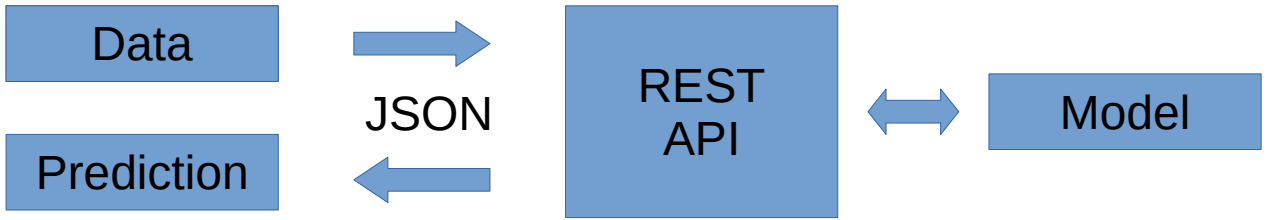
Securing an API

Why?

- worldwide IP addresses are scanned for open ports
- https only secures the transport, but API can be still abused
- some form of authentication necessary
- JWT = JSON Web Token
  - proposed standard for creating data with optional signature and/or optional encryption whose payload holds JSON that asserts some number of claims
  - tokens are designed to be compact, URL-safe, and usable especially in a web-browser single-sign-on (SSO) context
  - source: https://en.wikipedia.org/wiki/JSON_Web_Token

What?

- Developed a great ML model
- Instead of handing out model, provide API



How?

- model
  - scikit-learn: https://scikit-learn.org/
- API
  - flask: https://flask.palletsprojects.com/
  - flask-jwt: https://pythonhosted.org/Flask-JWT/
- Deployment in AWS:
  - https://medium.com/analytics-vidhya/deploy-restful-api-with-flask-on-aws-12575c808b4
  - https://linuxroutes.com/default-usernames-aws-for-logging-on-to-different-linux-distros-ec2-instance/
- secure transport: uWSGI + nginx

  outside of scope, instead see tutorial:

  https://www.digitalocean.com/community/tutorials/how-to-serve-flask-applications-with-uswgi-and-nginx-on-ubuntu-18-04

Let’s see some code

What else?

- flask-jwt-extended
  - being able to refresh tokens

    https://github.com/vimalloc/flask-jwt-extended
- flask-sqlalchemy
  - for recording usage of model
  - charge by usage??

    https://flask-sqlalchemy.palletsprojects.com/