# SQA Final Project Report and Screenshots

Team: CWTJABDBT
Members: cwt0013, dbt0013, jab0217

https://github.com/cwtillery/CWTJABDBT-SQA2023-AUBURN

## Report

- Alex Barras:
  - My responsibility in this project was to perform software forensics via logging. Specifically, editing/introducing logging to five different python methods in the repository. All of the methods I edited are in the file linked below and the generated log is also linked. I learned a good bit about performing software forensics and how to implement logging. Also, I learned about the different types of attacks that performing these duties can help detect / prevent such as poisoning attacks and model tricking attacks.

- Carson Tillery:
  - My responsibility in this project was to add a static analysis security git hook. I achieved this using Bandit, as we used in class. The git hook runs Bandit on all files in the repository when committing and outputs the results into a file called “results.csv”, which is automatically added to the commit. Through doing this, I learned a lot more about how Github works in general, and more deeply about how git hooks work. I feel in the future I will be able to utilize this knowledge to create git hooks to suit my needs, beyond the purposes we used git hooks for in class. Further, I feel I learned more about how security is monitored and upheld using continuous integration.

- Brown Teague:
  - My responsibility was to perform fuzzing on 5 python methods of my choice. I chose 5 methods, which you can see specifically in the fuzz.py file), from graphtaint.py. The fuzzing output is in the file fuzzingOutput.txt. I learned about the importance of identifying and addressing software vulnerabilities. Fuzzing involves testing a software system by feeding it with invalid, unexpected, or random input to see how it behaves. This technique helped

me understand that correct input is necessary for software to run. Looking at the variety of errors in the output, it is clear fuzzing is an important practice for software development and developers should make sure to have a failsafe in case these inputs occur.

## Security Analysis Git Hook:

- Generated results file (.csv)
    - https://github.com/cwtillery/CWTJABDBT-SQA2023-AUBURN/blob/main/results.csv

## Fuzzing:

- https://github.com/cwtillery/CWTJABDBT-SQA2023-AUBURN/blob/main/fuzzingOutput.txt

## Software Forensics:

- Generated report log:
    - https://github.com/cwtillery/CWTJABDBT-SQA2023-AUBURN/blob/main/SIMPLE-LOGGER.log
- Edited python file:
    -https://github.com/cwtillery/CWTJABDBT-SQA2023-AUBURN/blob/main/KubeSec-master/scanner.py

## (Extra)
## Codacy Static Analysis:

https://github.com/cwtillery/CWTJABDBT-SQA2023-AUBURN/actions/runs/4633442244/jobs/8198668921