Software Quality Assurance Final Project

Maritza Spott

# 4A. Git Hook

The CSV file created is labeled ‘issues\_found.csv’. It is currently in .gitignore with pre-commit, but a copy of the file from one upload is in the main folder (SQAWARRIOR-SQA2023-AUBURN). A screenshot is also below.

Graphical user interface, application, table, Excel

Description automatically generated

# 4B. Fuzzing

I created a fuzzing method that selects seven random phrases from blns.json file to run through five functions in scanner.py. The five functions are `isValidUserName`, `isValidPasswordName`, `isValidKey`, `checkIfValidKeyValue`, and `checkIfValidSecret`. Screenshots of output are below where some of the methods allow emojis to be valid.

Graphical user interface, text, application, Word

Description automatically generatedGraphical user interface, text, application, Word

Description automatically generatedGraphical user interface, text, application, Word

Description automatically generatedGraphical user interface, application, Word

Description automatically generated

# 4B. Forensics: Logging

For 4B, I used logging to identify vulnerabilities from poisoning attacks and model tricking.

The five functions scanned were `getYAMLfiles`, `scanForHTTP`, `scanForMissingSecurityContext`, and `getItemFromSecret` in scanner.py and `getYAMLfiles` in graphtaint.py.

# Lessons Learned

1. Bandit is very picky about white space and does not like blsn.json at all because it is not formatted according to its standards.
2. I had to push files independently to make sure they were up to Bandit’s standards.
3. Some of the checking functions in scanner.py worked better than others, as they wouldn’t allow many fraudulent inputs go through as valid.