M2 - Assurance

For general practices, our team worked on the code base through peer programming (aided by Visual Studio Code Live Share - https://visualstudio.microsoft.com/services/live-share/). We reviewed each other's code changes through a shared Git repository (https://github.com/MWRose/Accord).

We intend to do unit tests and regression tests in the future. For now, we confirm our features that we are delivering in this prototype through manual testing and running of the application.

We also logged the messages coming through the server to ensure that it was unreadable. We checked to make sure that our signing functions were functioning correctly by passing in the incorrect values to check against.

The following were tools we used to test:

- We use the automated bug-finding tool Pylint (https://www.pylint.org) to scan for low level bugs. Pylint rated our code and showed us which lines we could improve for readability too.
- We also used (http://mypy-lang.org) to do static type checking on our cryptographic functions. This allowed us to ensure that the types coming into and out of our functions were correct and allowed us to fix some incorrect data type errors before they became a problem
- We also catch errors through try-excepts. This is done for connections and to verify the signatures and tags are correct and not allow the program to go any further
- We also wrote a script that spawns up three different terminal windows for two instances of Client script and one instance of Server script in order to speed up manual testing