COMP260 – Distributed Programming Worksheet 11

Cryptography

Introduction

The goals of this worksheet are two-fold, firstly to explore the frameworks for cryptography presented in today's lecture and, secondly, to integrate cryptograph as part of assignment 2.

Cryptography sandbox

Like the other features we have looked at in this half of the module, a good place to start with encryption is to build a simple Python sandbox that will let you acquire and experiment with the cryptography and cryptodome libraries. Remember, once you get these working within you PyCharm environments on the PC, you will need to pip them into your Ubuntu hosted server environment.

A good starting point for this activity is to take the code examples on fernet and cipher from the lecture and build a command-line application that will allow you to generate keys and crypt and decrypt messages.

Users and Cryptography CRUD sandbox

As we've seen from the lectures on user accounts and encryption, keys or salts form the core of encryption and user password security. Therefore, your user-based security architecture needs to use salts and keys that will work for both your users (as salts) and your cryptography (as keys).

The simple cryptography sandbox can now be extended with the user sandbox from workshop 9 to create a framework where user accounts can be created with salts that are compatible with your cryptography solutions and data can be encrypted and decrypted with them

Next Steps: MUD cryptography

Now apply all of this to your MUD application.