**Installation Guide: Mceliece KEM**

This guide is for users operating in a Linux or MacOS environment. Guide for Windows is given in separate section in the last.

**Cryptography Project Setup Guide**

This guide provides step-by-step instructions to set up and run the cryptography project. This setup includes unzipping necessary files, creating a Python virtual environment, installing dependencies, and running the application.

**Prerequisites**

- Python 3.6 or higher

- pip (Python package installer)

- Access to terminal or command line interface

**Setup Instructions**

1. **Extract the McEliece Folder**

Before proceeding with the setup, ensure that you have the McEliece.zip file in your working directory.

unzip McEliece.zip

cd McEliece

2. **Create Python Virtual Environment**

It is recommended to create a virtual environment to avoid conflicts with other package versions you might have installed.

# Install virtualenv if it's not installed

pip install virtualenv

# Create a virtual environment

virtualenv venv

# Activate the virtual environment

source venv/bin/activate

3. Install Required Packages

Ensure all necessary Python packages are installed.

# Install Python and pip if they are not already installed

# This step may vary depending on your operating system.

# Install required Python packages

pip install numpy

pip install cryptography

pip install pyldpc

**Running the Application**

**Start the Server**

Open a terminal and run the following command to start the server.

python server\_version5.py

**Start the Client**

Open another terminal and run the following command to start the client.

python client\_version5.py

**Running Tests**

**Encryption Tests**

To run encryption tests, use the following command.

python encryption\_tests.py

**Efficiency Tests**

To run efficiency tests regarding the cryptographic processes, use the following command.

python Testing\_metrics.py

**Additional Notes**

* Ensure that you keep your Python environment up to date.
* If you encounter any issues during installation, verify that all commands are compatible with your operating system and Python version.

**Troubleshooting**

* If the pip commands fail, ensure that your virtual environment is activated, and pip is updated to the latest version.
* If Python packages fail to install, check for errors in the command line that might suggest what went wrong, and ensure your internet connection is stable.

**Installation guide for Windows**

This guide details the process for setting up and running the cryptography project on a Windows platform. Follow these steps to unzip necessary files, set up a Python virtual environment, install dependencies, and execute the application.

**Prerequisites**

- Python 3.6 or higher (Ensure Python and pip are added to PATH during installation)

- Access to Command Prompt or PowerShell

**Setup Instructions**

**1. Extract the McEliece Folder**

First, ensure you have downloaded the McEliece.zip file. Right-click the file in your File Explorer and select "Extract All..." to unzip the folder to a desired location.

**2. Create Python Virtual Environment**

Creating a virtual environment is crucial to avoid conflicts with globally installed packages.

#Navigate to the project directory

cd path\to\McEliece

# Install virtualenv if it's not installed

pip install virtualenv

# Create a virtual environment

virtualenv venv

# Activate the virtual environment

.\venv\Scripts\activate

3. **Install Required Packages**

Install all the necessary packages using pip.

# Install required Python packages

pip install numpy

pip install cryptography

pip install pyldpc

**Running the Application**

**Start the Server**

Open a Command Prompt or PowerShell window and execute the following command to start the server

python server\_version5.py

**Start the Client**

Open another Command Prompt or PowerShell window and execute the following command to start the client.

python client\_version5.py

**Running Tests**

**Encryption Tests**

To perform encryption tests, run the following command.

python encryption\_tests.py

**Efficiency Tests**

To conduct efficiency tests on the cryptographic processes, run the following command.

python Testing\_metrics.py

**Additional Notes**

* Make sure Python and pip are correctly installed and added to the Windows PATH to allow command-line access.
* Always ensure that your Python packages and virtual environment are up to date.

**Troubleshooting**

* If pip commands do not work, ensure your virtual environment is activated and pip is updated.
* If Python packages fail to install, review the command line error messages for possible causes, like missing prerequisites or internet connectivity issues.