Компьютерный практикум №3

Выполнили студенты

Группы ФИ-02мн

Олефир Полина

Попов Артём

Бещук Андрей

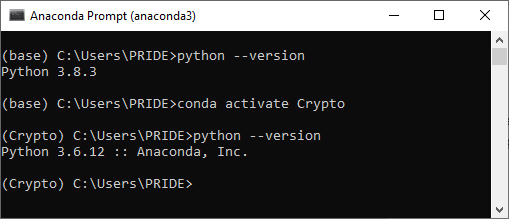
1. **Цель практикума**

Реализовать цифровую подпись по Эль Гамалю, используя криптобиблиотеку PyCrypto для языка программирования Python. Реализовать возможность хранения открытых ключей и подписей в файлах.

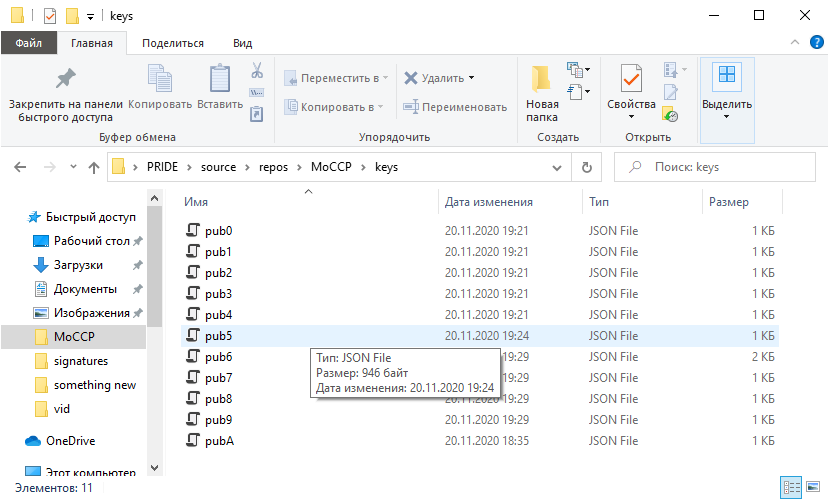
1. **Детали выполненной работы**

В связи с долгими практическими попытками использовать библиотеку PyCrypto в Python, было решено использовать модуль Anaconda, в котором библиотека PyCrypto уже установлена.

В дополнении к этому оказалось, что Python 3.8.3 не умеет в PyCrypto даже с установленным модулем Anaconda. Именно поэтому в самом модуле Anaconda была создана отдельная среда Crypto с Python 3.6.12 для работы с криптобиблиотекой PyCrypto.

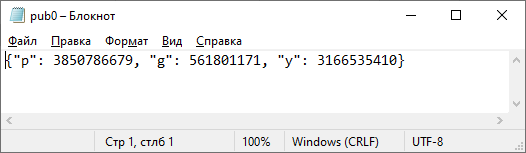


В папке вместе с этим протоколом лежит файл «pycrypto\_elgamal.py», в котором содержится код компьютерного практикума. При желании читатель может его прочитать в любом текстовом редакторе. Разработка велась в приложении Spyder, которое идёт вместе с модулем Anaconda в комплекте, потому что там удобно подсвечиваются ключевые слова.

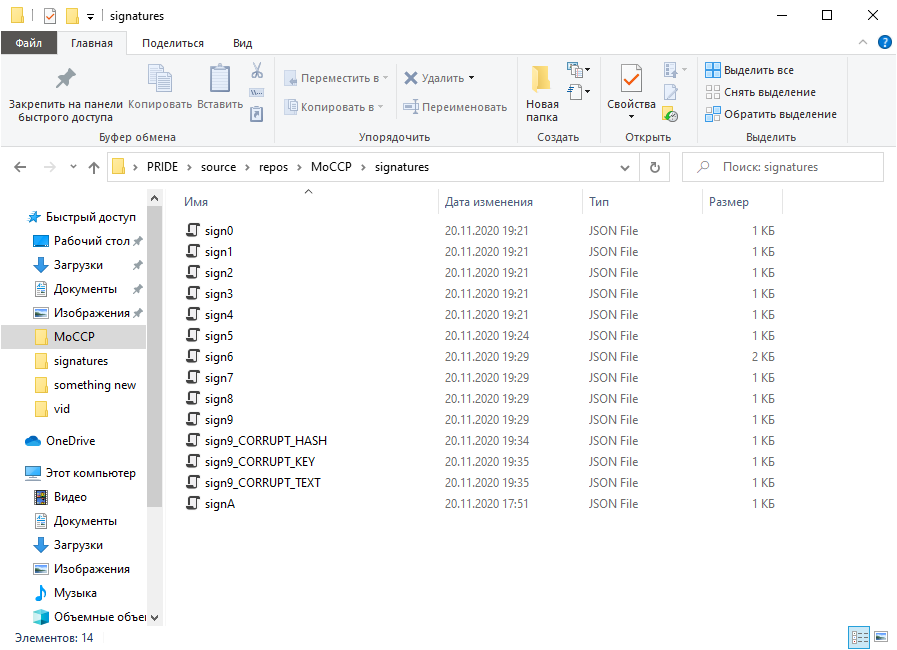
В коде реализуется класс «Ключ», который позволяет с собой делать разное: от двух путей его создания (генерация и чтение из файла), до выделения «публичной» части ключа, записи, подписи сообщения и верификации подписи. 

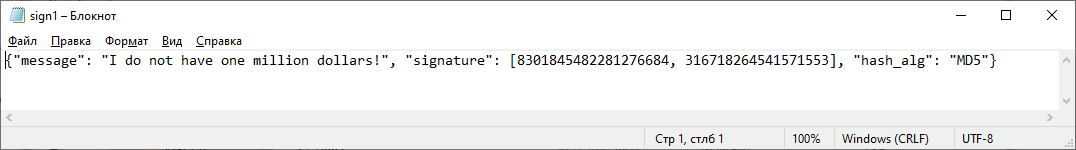
При создании ключа создается соответствующий файл в папке keys (если ещё не была создана). Каждый ключ (как и подпись) создаются в формате JSON File, потому что в них удобно хранить данные (данные сохраняются в формате пар «ключ:значение»).

Каждый файл ключа – это открытый ключ пользователя, который содержит в себе три публичных параметра по цифровой подписи Эль Гамаля:

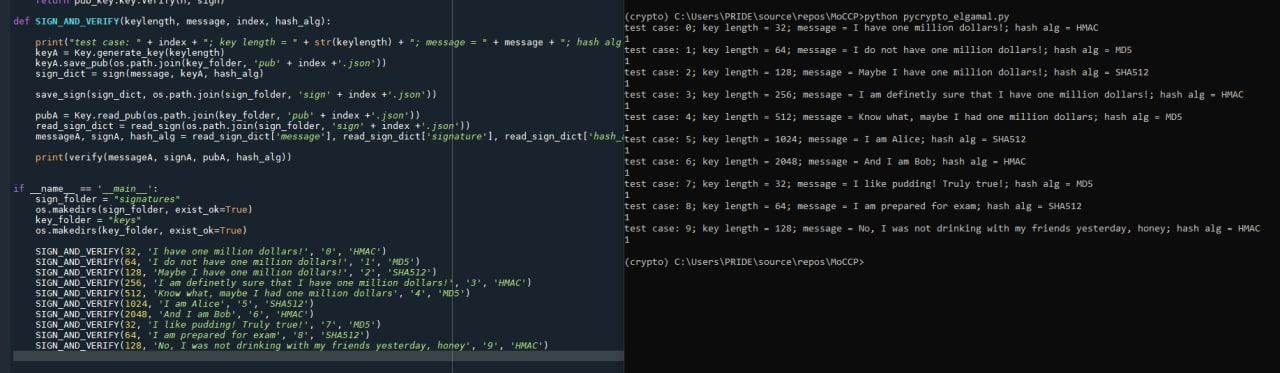


После подписи некоторого сообщения соответствующим ключом создаётся файл подписи:

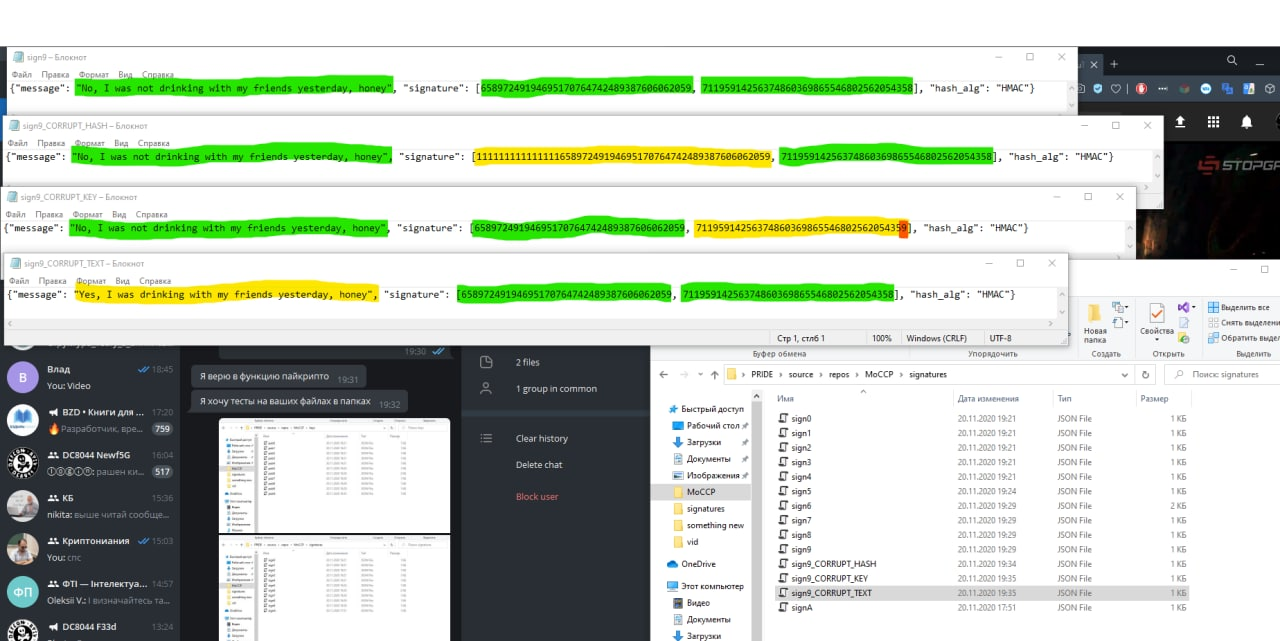


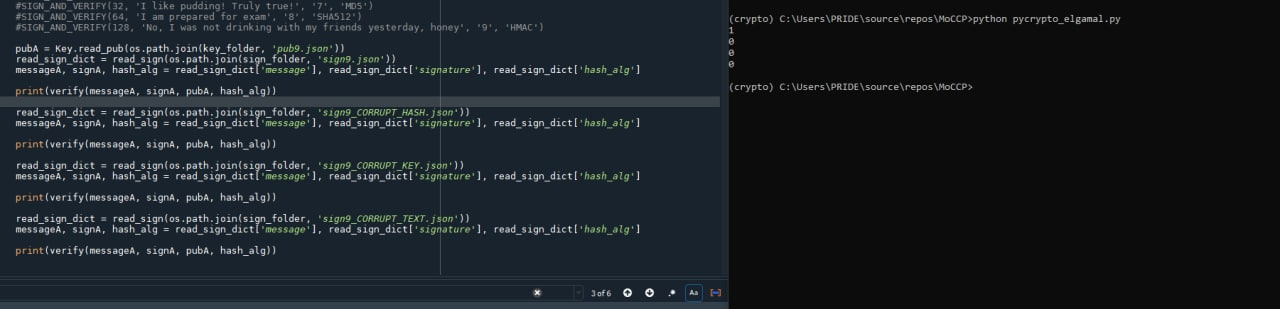
Каждый из которых в свою очередь состоит из параметров: подписанного сообщения, подписи (хеш-сообщения и открытый ключ) и алгоритм хеш-функции.

Примеры работы алгоритма можете увидеть на следующих скриншотах:

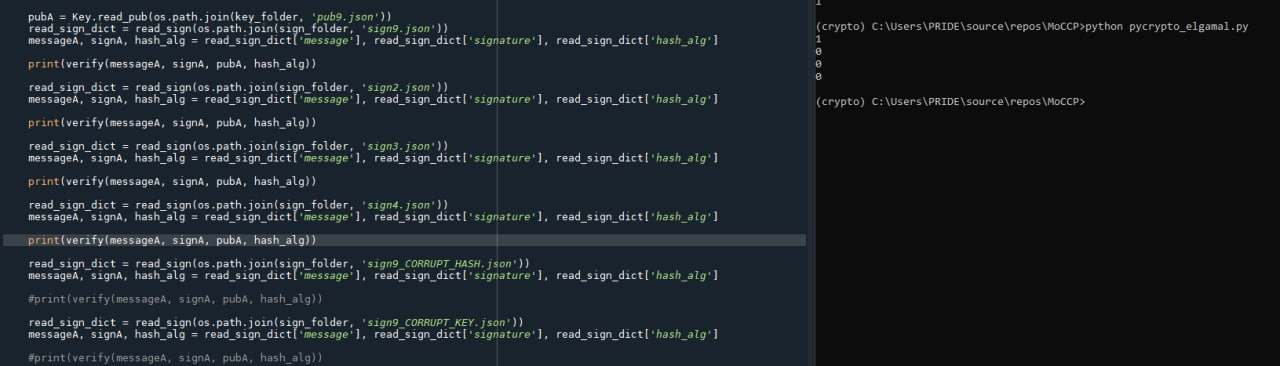


И попытки проверить невалидную подпись





И попытку проверить подпись не тем ключом:



1. **Выводы**

PyCrypto в самом деле очень удобен для реализации криптопримитивов, однако печально, что новые версии Python его не поддерживают.

PyCrypto не умеет в сертификаты: он их не создает, не проверяет.