Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Інститут комп'ютених наук та інформаційних технологій

Кафедра «Системи штучного інтелекту»

Лабораторна робота №5

з курсу "Технології захисту інформації"

Виконала:

студентка групи КН-308

Ріжко Марія

Перевірив:

Яковина В.С.

Львів 2020 р.

Тема роботи

Створення програмного засобу для цифрового підпису інформації з використанням Microsoft CryptoAPI

Завдання

З використання функцій CryptoAPI створити прикладну програму для створення і перевірки цифрового підпису за стандартом DSS. Програмна реалізація повинна виводити значення підпису як для рядка, заданого в полі вводу, так і для файлу. Результат роботи програми повинен відображатись на екрані з можливістю наступного запису в файл. Крім того програма повинна мати можливість перевірити цифровий підпис будь-якого файлу за наявним файлом підпису, записаним у шістнадцятковому форматі. У звіті навести протокол роботи програми та зробити висновки.

Програмна реалізація

```
from Crypto.Hash import SHA256
from Crypto.PublicKey import DSA
from Crypto.Signature import DSS


def generate_keys(username, bytes_num=1024):
    key = DSA.generate(bytes_num)

    with open(username + '_private.pem', 'w') as f:
        f.write(key.export_key().decode('ascii'))

    with open(username + '_public.pem', 'w') as f:
        f.write(key.publickey().export_key().decode('ascii'))


def create_signature(file, username):
    with open(username + '_private.pem', 'r') as f:
        private = DSA.import_key(f.read().encode('ascii'))

    with open(file, 'r') as f:
        message = f.read()

    dss = DSS.new(private, 'fips-186-3')
    signature = dss.sign(SHA256.new(message.encode('utf-8')))

    with open('signed_file.txt', 'wb') as f:
        f.write(message.encode('utf-8') + b'\n' + signature)


def verify_signature(file, username):
    with open(username + '_public.pem', 'r') as f:
        public = DSA.import_key(f.read().encode('ascii'))
```

```
    dss = DSS.new(public, 'fips-186-3')

    with open(file, 'rb') as f:
        message = f.read()

    sign = message.rfind(b'\n')

    try:
        dss.verify(SHA256.new(message[:sign]), message[sign + 1:])
        print('verified')
    except ValueError:
        print('corrupted')


if __name__ == '__main__':
    while True:
        print('Choose what to do:\n'
              '1. Generate key\n'
              '2. Sign file\n'
              '3. Verify file')

        c = input()
        if c == '1':
            print('Enter username')
            username = input()
            generate_keys(username)
        elif c == '2':
            print('Enter file')
            file = input()
            print('Enter username')
            username = input()
            create_signature(file, username)
        elif c == '3':
            print('Enter file')
            file = input()
            print('Enter username')
            username = input()
            verify_signature(file, username)
        else:
            break
```

Результати виконання

```
C:\Users\mariz\PycharmProjects\tzi\venv\Scripts\python.exe C:/Users/mariz/PycharmProjects/tzi/lab5/lab5.py
Choose what to do:
1. Generate key
2. Sign file
3. Verify file
1
Enter username
user
Choose what to do:
1. Generate key
2. Sign file
3. Verify file
2
Enter file
test.txt
Enter username
user
Choose what to do:
1. Generate key
2. Sign file
3. Verify file
3
Enter file
signed_file.txt
Enter username
user
verified
```

Змінимо підписаний файл.

```
C:\Users\mariz\PycharmProjects\tzi\venv\Scripts\python.exe C:/Users/mariz/PycharmProjects/tzi/lab5/lab5.py
Choose what to do:
1. Generate key
2. Sign file
3. Verify file
3
Enter file
signed_file.txt
Enter username
user
corrupted
```

Висновок

На лабораторній роботі я створила програмний засіб для цифрового підпису інформації з використанням Microsoft CryptoAPI.