Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»
(Университет ИТМО)

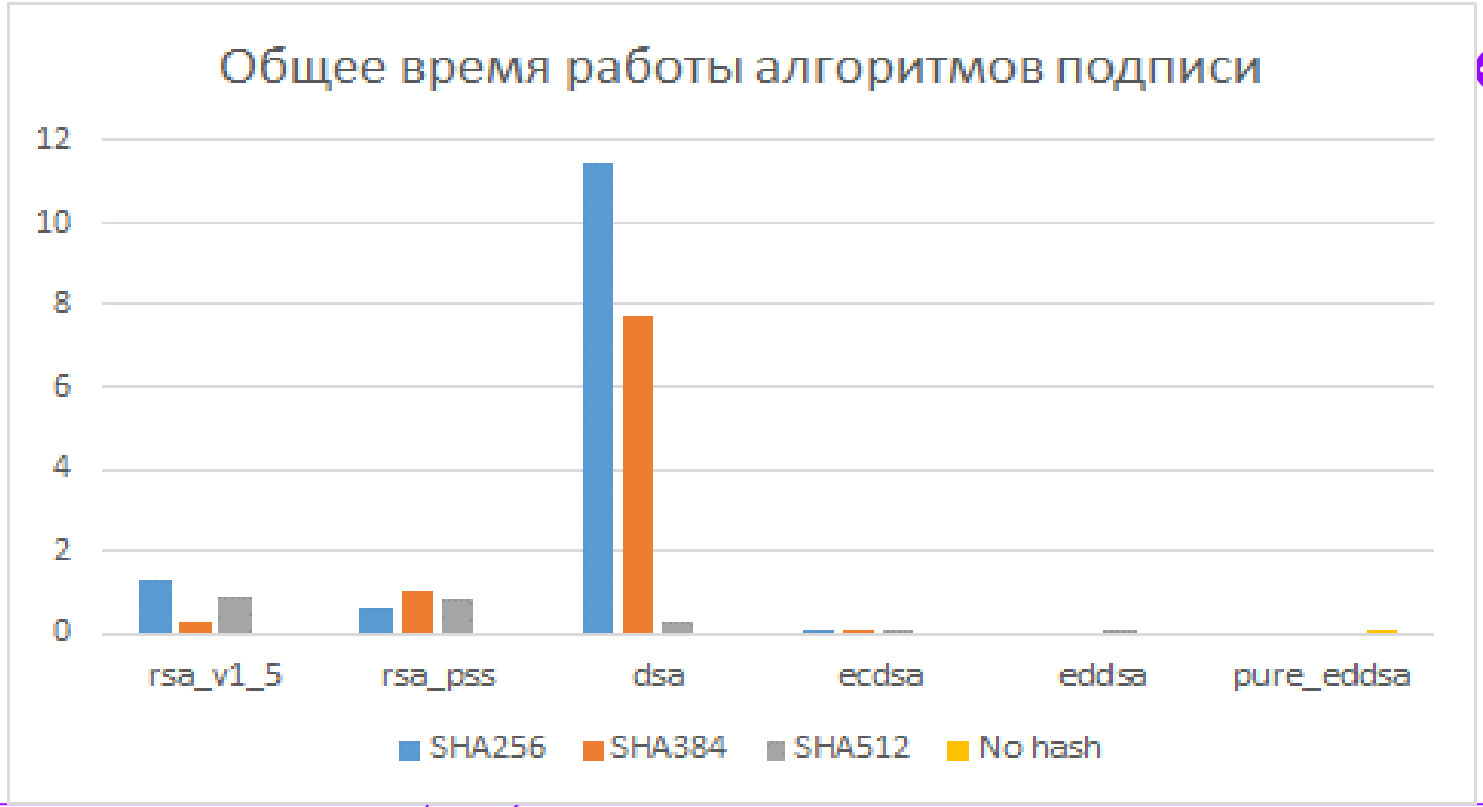
ИТМО

Выпускная квалификационная работа

Разработка интерактивной методики оценки эффективности систем цифровой подписи на основе библиотеки PyCryptodome

Факультет безопасности информационных технологий
Образовательная программа Технологии защиты информации
Направление подготовки (специальность) 10.03.01 Информационная безопасность
Выполнил студент: Гутник Дмитрий Вячеславович, N34491
Научный руководитель: доцент ФБИТ, к.т.н., Таранов Сергей Владимирович
Санкт-Петербург, 2023

Актуальность



Цель и задачи

Цель: повышение эффективности применения алгоритмов цифровой подписи путём рекомендации подходящего вида ЦП в зависимости от условий применения.

Задачи:

1. Анализ и разделение на виды актуальных алгоритмов цифровой подписи
2. Разработка критериев к методике оценки цифровых подписей
3. Создание методики оценки эффективности цифровых подписей
4. Разработка программной реализации методики
5. Разработка рекомендаций к практическому применению методики

PyCryptodome

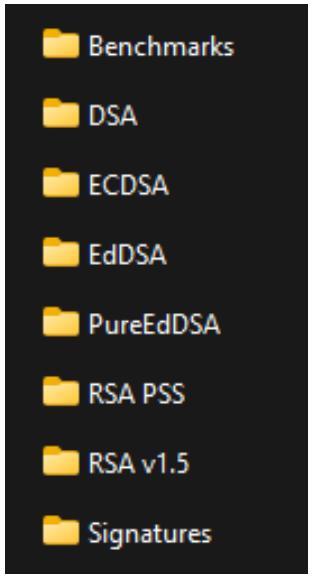
Поддерживаемые алгоритмы

1. RSA со следующими вариациями:
    - PKCS#1 v1.5
    - PKCS#1 PSS
2. EdDSA со следующими вариациями:
    - EdDSA
    - Pure EdDSA
3. DSA со следующими вариациями:
    - DSA
    - ECDSA

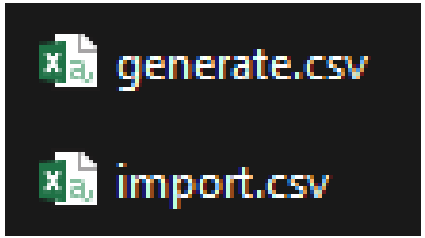
Область применения

- Системные администраторы
- Настройка средств обработки информации
- Программисты, использующие библиотеку PyCryptodome
- Пользователи, далёкие от криптографии

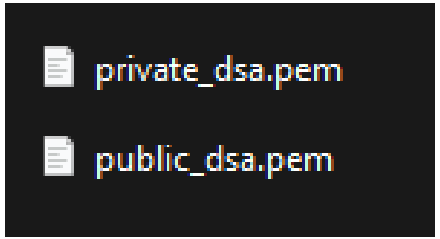
Программная реализация



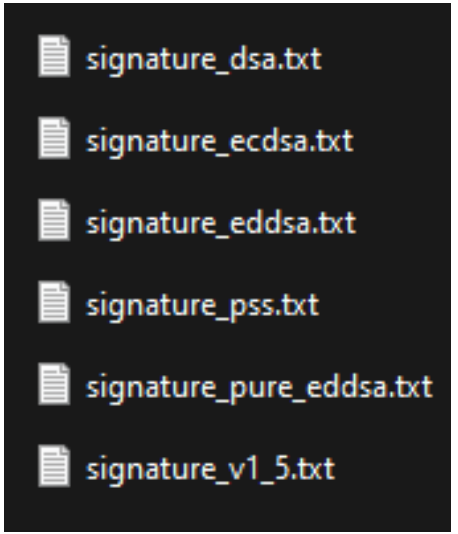
Файловая структура методики



Хранение тестов производительности



Хранение открытого и закрытого ключей



Хранение подписей

Тест производительности алгоритмов

import benchmark

algorithm	hash	sign	verify	total
rsa_v1_5	SHA256	0.03400	0.00351	0.03752
rsa_v1_5	SHA384	0.03355	0.00300	0.03655
rsa_v1_5	SHA512	0.03350	0.00200	0.03551
rsa_pss	SHA256	0.03275	0.00300	0.03575
rsa_pss	SHA384	0.03227	0.00400	0.03627
rsa_pss	SHA512	0.03233	0.00255	0.03488
dsa	SHA256	0.08068	0.08228	0.16296
dsa	SHA384	0.08088	0.08126	0.16214
dsa	SHA512	0.07934	0.08100	0.16034
ecdsa	SHA256	0.00200	0.00651	0.00852
ecdsa	SHA384	0.00200	0.00554	0.00755
ecdsa	SHA512	0.00201	0.01552	0.01753
eddsa	SHA512	0.00101	0.00200	0.00301
pure_eddsa	-	0.00000	0.00200	0.00200

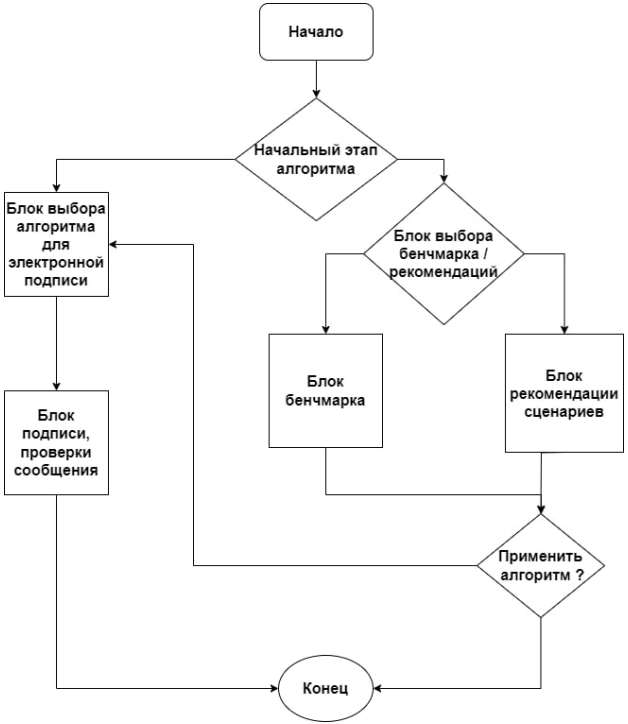
Тест производительности с импортом ключа

generate benchmark

algorithm	hash	sign	verify	total
rsa_v1_5	SHA256	1.25953	0.00652	1.26605
rsa_v1_5	SHA384	0.24042	0.00600	0.24643
rsa_v1_5	SHA512	0.85727	0.00651	0.86378
rsa_pss	SHA256	0.64162	0.00600	0.64762
rsa_pss	SHA384	1.04822	0.00552	1.05374
rsa_pss	SHA512	0.77940	0.00601	0.78541
dsa	SHA256	11.39181	0.08569	11.47750
dsa	SHA384	7.66034	0.08461	7.74495
dsa	SHA512	0.20279	0.08482	0.28761
ecdsa	SHA256	0.00251	0.00898	0.01149
ecdsa	SHA384	0.00201	0.00851	0.01052
ecdsa	SHA512	0.00251	0.00851	0.01102
eddsa	SHA512	0.00100	0.00601	0.00701
pure_eddsa	-	0.00150	0.00601	0.00752

Тест производительности с созданием ключа

Блок схема методики

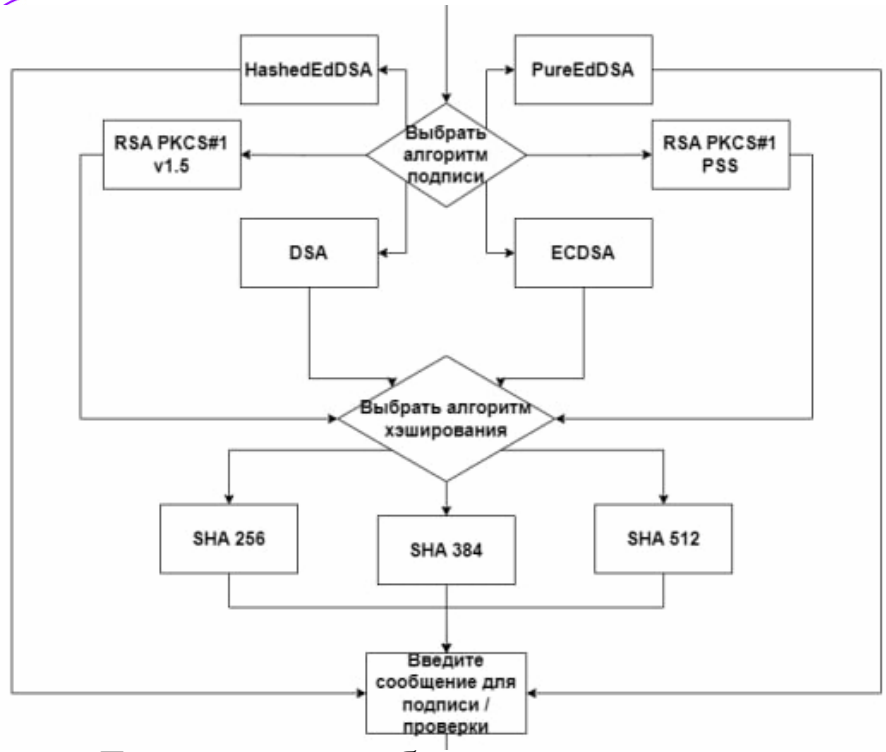


Упрощённая блок схема

```python
match input():
    case '1':
        print("Recompounded algorithms: PureEdDSA\n"
              "Not recommended algorithms: DSA, RSA v1.5\n")
    case '2':
        print("Recompounded algorithms: DSA, RSA v1.5, EdDSA, ECDSA\n"
              "Not recommended algorithms: - \n")
    case '3':
        print("Recompounded algorithms: ECDSA, EdDSA\n"
              "Not recommended algorithms: DSA, RSA v1.5, RSA PSS\n")
    case '4':
        print("Recompounded algorithms: ECDSA, EdDSA\n"
              "Not recommended algorithms: DSA, RSA v1.5, RSA PSS\n")
    case '5':
        print("Recompounded algorithms: PureEdDSA, ECDSA, EdDSA, RSA v1.5, RSA PSS\n"
              "Not recommended algorithms: DSA\n")
    case _:
        print("wrong action")
        sys.exit()
if input("Would you like to see the benchmark of all supported signature algorithms?\n"
         "y/n") == 'y':
```

Код блока рекомендации

Блок выбора алгоритмов



Блок схема выбора алгоритмов

```
Choose an algorithm:
 1. Rsa PKCS#1 v1.5
 2. Rsa PKCS#1 PSS
 3. DSA
 4. ECDSA
 5. HashedEdDSA
 6. PureEdDSA
3
Select a Hash algorithm:
1. SHA256
2. SHA384
3. Sha512
3
Enter the message to be signed or verified
test
Would you like to:
1.Sign
2.Verify
the message?
```

Демонстрация работы кода

Блок рекомендации сценариев

Критерии рекомендации сценариев

Основные критерии:

- скорость работы алгоритма
- длина подписи
- размер ключа
- распространённость алгоритма в существующем ПО

Атаки, рассматриваемые при формировании рекомендаций:

- коллизия Хэшей 1 и 2 рода
- перебор ключа
- атаки на алгоритм дополнения
- адаптивная атака с выбранным зашифрованным текстом
- атака с малыми значениями секретной экспоненты
- атаки по сторонним каналам
- атаки связанные с псевдослучайными числами
- и другие

Примеры размерности подписей и ключей

Алгоритм	Размер, байт
DSA	56
ECDSA	132
EdDSA	64
Pure EdDSA	64
PKCS#1 v1.5	256
PKCS#1 PSS	256

Размеры подписей

Алгоритм	Размер закрытого ключа, байт	Размер открытого ключа, байт
DSA	882	1189
ECDSA	390	272
EdDSA	120	114
Pure EdDSA	120	114
PKCS#1 v1.5	1678	450
PKCS#1 PSS	1674	450

Размеры ключей

Аналоги

	Разработанная методика	Cryptoy	Методика Жданова	Cryptool
Несколько алгоритмов подписи	+	-	-	-
Ассиметричные алгоритмы шифрования	+	-	-	+
Оптимально подобранные настройки алгоритмов	+	-	-	+
Подпись, проверка подписи	+	-	-	+
Возможность использования кода алгоритма вне программы	+	-	+	-

Итог

В результате выполнения этой работы была разработана методика, которая позволяет:

- Подписывать сообщения
- Проверять подписи
- Создавать ключи
- Использовать заранее созданные ключи для подписи
- Рекомендовать алгоритмы в зависимости от сценария использования
- Проводить тесты производительности алгоритмов электронной подписи

и отличается от аналогов тем, что:

- Поддерживает большее количество алгоритмов подписи
- Поддерживает несколько уровней защиты от коллизий хэшей
- Позволяет оценивать время работы алгоритма на конкретном устройстве

Выводы

Была достигнута следующая цель: повышена эффективность применения алгоритмов цифровой подписи путём рекомендации подходящего вида ЦП в зависимости от условий применения.

Были выполнены следующие задачи:

1. Проанализированы и разделены на виды актуальные алгоритмы цифровой подписи
2. Разработаны критерии к методике оценки цифровых подписей
3. Создана методика оценки эффективности цифровых подписей
4. Разработана программная реализация методики
5. Разработаны рекомендации к практическому применению методики

Спасибо
за внимание!

ITMO more than a UNIVERSITY