Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ОТЧЕТ

О ЛАБОРАТОРНОЙ РАБОТЕ

Лабораторная работа №1
по теме:
«Изучение свойств криптографических функций хеширования»

Исполнитель, студент группы 201-361
_____ Н.А. Фельдбуш

Москва, 2023

Постановка задачи:

Напишите программу, генерирующую из файла leasing.txt эквивалентные по смыслу текстовые документы в количестве, достаточном (условно) для возникновения коллизии функции хеширования SHA-1. Типовые приемы: замена слов и словосочетаний на синонимы; исключение или включение союзов, вводных слов и эпитетов; внедрение управляющих символов. Оцените изменение объема файла после модификации.

Ход работы:

Коллизия SHA-1 означает ситуацию, когда два разных входных сообщения дают одинаковое хэш-значение. Это связано с тем, что размер хэша SHA-1 – 160 бит, что позволяет находить коллизии с помощью современных вычислительных мощностей.

Выполним поставленную задачу при помощи средств языка программирования python. Перед началом следует обратить внимание на функцию “init” (рис.1).

```
def init():
    with open(rel_path, "r", encoding='utf-8') as f:
        with open("docs/leasing0.txt", "w", encoding='utf-8') as wrt:
            wrt.write("\n")
            a = f.readline()
            while a:
                wrt.write(a)
                a = f.readline()
            wrt.close()
        f.close()
    new_sha1 = subprocess.run(["C:/Program Files/Git/usr/bin/openssl.exe", "dgst",
                               "-sha1", 'docs/leasing0.txt'], stdout=subprocess.PIPE).stdout.decode().strip()
    new_sha1 = new_sha1[new_sha1.find(" ")+1:]
    counter = 1
    return[new_sha1, counter]
```

Рисунок 1 – Функция “init”

Эта функция проводит инициализацию: создает файл с названием “leasing0.txt” и устанавливает счетчик равным единице.

Затем при запуске файла, мы вызываем функцию “main”, внутри которой проводится инициализация с помощью функции “init” (рис.2).

```
def main():
    args = init()
    new_sha1 = args[0]
    counter = args[1]
    while counter < 300 and new_sha1 != hsh:
        with open(f"docs/leasing{counter-1}.txt", "r", encoding='utf-8') as f:
            with open(f"docs/leasing{counter}.txt", "w", encoding='utf-8') as wrt:
                wrt.write("\n")
                a = f.readline()
                while a:
                    wrt.write(a)
                    a = f.readline()
                wrt.close()
        f.close()
        new_sha1 = subprocess.run(["C:/Program Files/Git/usr/bin/openssl.exe", "dgst",
                                   "-sha1", f"docs/leasing{counter}.txt"], stdout=subprocess.PIPE).stdout.decode().strip()
        new_sha1 = new_sha1[new_sha1.find(" ")+1:]
        print(f"leasing{counter}.txt " + new_sha1 + " " +
              str(os.path.getsize(f"docs/leasing{counter}.txt")) + " bytes. Size difference: " + str(os.path.getsize(f"docs/leasing{counter}.txt") - os.path.getsize("docs/leasing.txt")) + " bytes
        counter += 1
```

Рисунок 2 – Функция “main”

Внутри функции “main” открывается созданный в “init” файл, в этот файл добавляется символ “\n”, что означает переход на новую строку, а затем создается новый файл с этим добавленным символом.

Вслед за этим с помощью библиотеки “subprocess” происходит получение хеша нового файла, а к счетчику прибавляется единица.

Цикл “while” происходит до тех пор, пока счетчик не станет равным 300, или пока хэш исходного файла не совпадет с хэшем нового. Результатом работы программы является создание от 1 до 300 новых файлов, в зависимости от выполнения второго условия (рис.3).

```
leasing287.txt 433b739e86288f68c4d01770f2f3d79773414a02 27243 bytes. Size difference: 576 bytes.
leasing288.txt 95417c4cc39a2fb36b3533029edb5ba97666e8bb 27245 bytes. Size difference: 578 bytes.
leasing289.txt 2081df9fd19dd80b5750f65baa1419213b9fc4dc 27247 bytes. Size difference: 580 bytes.
leasing290.txt d0340028acb167041d8b898073b73e6e99b86c44 27249 bytes. Size difference: 582 bytes.
leasing291.txt 14f3b0034bbdae13580f185f20f5126fd4659b1e 27251 bytes. Size difference: 584 bytes.
leasing292.txt 5fa14798860d2f191bb1740793c538a9e960ac43 27253 bytes. Size difference: 586 bytes.
leasing293.txt 419e539c6f85140567a7e4037a3756929bd28725 27255 bytes. Size difference: 588 bytes.
leasing294.txt 4fe97b3e9c2bb04d2d6aebf39e025769414e8317 27257 bytes. Size difference: 590 bytes.
leasing295.txt f47ec3469c5d79d8ee8a7ec14cd03dc18b091587 27259 bytes. Size difference: 592 bytes.
leasing296.txt 6ecd322c6423a0681ae3c9ba21174d9bbe3d30eb 27261 bytes. Size difference: 594 bytes.
leasing297.txt 4841f5a41130b81c6a2f6803f93470a3d9e69043 27263 bytes. Size difference: 596 bytes.
leasing298.txt 2a090e7064b9e6c887879617d57bd6af73bff5b4 27265 bytes. Size difference: 598 bytes.
leasing299.txt e245c580f1d8b2bd4e5884538b974a3ff60835ac 27267 bytes. Size difference: 600 bytes.
```

Рисунок 3 – Результат работы программы