**Отчет по лабораторной работе №3 по курсу «Криптография»**

Выполнил Лисин Роман Сергеевич, М8О-306Б-20.

***Задание***

Разложить число на нетривиальные сомножители. Ниже представлены 256 вариантов. Вариант выбрать следующим образом: свое ФИО подать на вход в хеш-функцию, являющуюся стандартом, выход хеш-функции представить в шестнадцатеричном виде и рассматривать младший байт как номер варианта. В отчете привести подробности процесса вычисления номера варианта.

***Ход работы***

Для разложения числа на нетривиальные сомножители я пытался воспользоваться готовыми онлайн реализациями методов факторизации, но из-за того, что число очень большое, за приемлемое время разложить его с помощью этих методов нельзя. Далее я обратил внимание на необычное количество вариантов (256!) чисел, и эти числа часто имеют примерно одинаковое количество разрядов. Из этих соображений я решил найти НОД моего числа и каждого из чисел из других вариантов в поиске делителя, отличающегося от 1 и моего числа. А если найти хотя бы один делитель моего числа, то второй ищется делением моего числа на этот делитель. Проверил вручную я только НОД моего числа и числа из самого первого варианта. Чтобы найти НОД моего числа и каждого из оставшихся чисел, я перенёс числа из других вариантов в файл nums.txt и распарсил его. К письму приложил этот файл. Мне повезло, у числа из варианта 62 и моего числа из варианта 63 оказался общий нетривиальный делитель. Подробности ниже.

Код на Python (в комментариях пояснения по ходу выполнения работы):

import hashlib  
import math  
  
sha = hashlib.sha256()  
sha.update(b"Lisin Roman Sergeevich")  
print(sha.hexdigest())  
# b1bf248f84f3b35537dea29b3174917a56e4adb6c4c687c7192886846ff25963 => 63 вариант  
  
  
n = "4289768347595033163815777647496477228222324248034487112498698931372704251830081\  
9982530306849273760404545083259358361074281693497521845607618494659430911365415613960\  
9657846698238730278408301124726462849001925627991836137981845339998268005673309727418\  
393848509240279252932212027953803979379908094841205497301831"  
n = int(n)  
  
# Парсинг файла с числами из других вариантов  
other\_numbers = []  
f = open('nums.txt', 'r')  
new\_number = ""  
for line in f:  
 if line.startswith('n'):  
 if new\_number:  
 try:  
 other\_numbers.append(int(new\_number))  
 except ValueError:  
 # некоторые числа содержат ещё другие символы (к примеру [164/2679]) => пропускаем их, надеясь,  
 # что повезёт найти среди оставшихся нужное  
 pass  
 new\_number = ""  
 line = line[6:] # Отрезаем n(XX)=  
 new\_number += line.replace(" ", "").replace("\n", "")  
f.close()  
  
p, q = 0, 0  
for num in other\_numbers:  
 gcd = math.gcd(n, num)  
 if gcd != 1 and gcd != n:  
 p = gcd  
 q = n // gcd  
 # Подошло число из 62 варианта  
 break  
  
"""  
p = 231415785143337817650927036140017035633451428361201872548185461962202110636843  
q = 1853706023095171059591881315865868346276156632812287104471773351812388566315010423681716145191732864939897465802650124236337881495991938576981339490645005359934137842265014861311952930926391091029402313821538963064140947964580178517  
"""  
print(n == p \* q) # True