Университет ИТМО

Факультет ФПИ и КТ

Отчёт

по лабораторной работе 2.1

«Иформационная безопасность»

Вариант 7

Студент:

Ляо Ихун

Гр.P34131

Преподаватель:

Маркина Татьяна Анатольевна

Цель работы:

изучить атаку на алгоритм шифрования RSA посредством метода Ферма

Задача:

7	84032429242009	2581907	54879925681459 72167008182929 17828219756166 17814399744948 37136636080011 77223434260215 4272415279426 73759271926435 74021335775875 16903113250201 77520052156956 41247980943013

Выполнение:

```
import math

if __name__ == '__main__':
    N = 84032429242009
    e = 2581907
    C = [54879925681459,
         72167008182929,
         17828219756166,
         17814399744948,
         37136636080011,
         77223434260215,
         4272415279426,
         73759271926435,
         74021335775875,
         16903113250201,
         77520052156956,
         41247980943013
         ]
    n = int(math.sqrt(N)) + 1
    i = 0
    b = 2
    a = N
    p = 0
    q = 0
    while True:
        i = i + 1
        t = n + i
        d = a ^ b
        w = t ** 2 - N
        sqrt_w = math.sqrt(w)
        if sqrt_w % 1 == 0:
```

```
            sqrt_w = int(sqrt_w)
            p = t + sqrt_w
            q = t - sqrt_w
            print(f"p={p}, q={q}")
            break
        else:
            print("error, keep searching for p and q")
phi = round((p - 1) * (q - 1))
d = pow(e, -1, phi)

result = ""
for i in C:
    m = pow(int(i), d, N)
    part = m.to_bytes(4, byteorder='big').decode('cp1251')
    print(f'{i}^{d} mod {N} = {m} => text({m}) = {part}')
    result += part
print(f"result = {result}")
```

Результат:

```
error, keep searching for p and q
error, keep searching for p and q
error, keep searching for p and q
p=9176129, q=9157721
54879925681459^2475823295643 mod 84032429242009 = 4024496352 => text(4024496352) = пара
72167008182929^2475823295643 mod 84032429242009 = 3958105579 => text(3958105579) = ллел
17828219756166^2475823295643 mod 84032429242009 = 4243454956 => text(4243454956) = ьным
17814399744948^2475823295643 mod 84032429242009 = 3894471918 => text(3894471918) = и мо
37136636080011^2475823295643 mod 84032429242009 = 4059226348 => text(4059226348) = стам
77223434260215^2475823295643 mod 84032429242009 = 3895206112 => text(3895206112) = и, а
4272415279426^2475823295643 mod 84032429242009 = 551743973 => text(551743973) =  все
73759271926435^2475823295643 mod 84032429242009 = 3974164728 => text(3974164728) = марш
74021335775875^2475823295643 mod 84032429242009 = 4042519277 => text(4042519277) = рутн
16903113250201^2475823295643 mod 84032429242009 = 4226097391 => text(4226097391) = ые п
77520052156956^2475823295643 mod 84032429242009 = 3773490674 => text(3773490674) = акет
41247980943013^2475823295643 mod 84032429242009 = 4213189983 => text(4213189983) = ы -_
result = параллельными мостами, а всемаршрутные пакеты -_
```

Вывод:

В ходе выполнения работы мы реализовали метод Ферма для атаки на алгоритм шифрования RSA на языке python.