ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Факультет программной инженерии и компьютерной техники

Дисциплина «Информационная безопасность»

Лабораторная работа №2.3

«Атака на алгоритм шифрования RSA методом бесключевого чтения»

Вариант: 4

Учебно-методическое пособие: Криптографические системы с секретным и открытым ключом: учебное пособие. / А.А. Ожиганов; УНИВЕРСИТЕТ ИТМО. — Санкт-Петербург, 2015

Автор: Калинин Даниил Дмитриевич

Группа: P34141

Преподаватель: Маркина Татьяна Анатольевна

г. Санкт-Петербург

2024

Содержание

Цель работы

Изучить атаку на алгоритм шифрования RSA посредством метода бесключевого чтения.

Порядок выполнения работы

- Ознакомьтесь с теорией в [3], в подразделе («Бесключевое чтение»);
- Получите вариант задания у преподавателя;
- По полученным данным определите значения r и s при условии, чтобы $e_1 r + e_2 s = 1$. Для этого необходимо использовать расширенный алгоритм Евклида;
- Используя полученные выше значения r и s, запишите исходный текст;
- Результаты и промежуточные вычисления значений для любых трех блоков шифрованного текста оформите в виде отчета.

Вариант

Вариант	**Модуль, *N***	**Экспоненты**		**Блок зашифрованного текста**	
		e1	***e2***	**C1**	**C2**
4	535598392051	455341	396971	444982997352 277831853272 133187882628 331361392426 273206302188 470299046774 168157171491 258737286129 312335302650 489235057221 427689116872 418723605534 135022585485	358696089912 360292494113 91390259562 534590606880 193203217609 166702058071 68207231399 487524624411 325841328769 533726724224 369967614519 247201359991 478832067683

Выполнение работы

Код

```
def extended_euclidean_algorithm(e1, e2):
   if e2 == 0:
       print(f'Расш. алг. Евклида (e1 = {e1}, e2 = {e2}) --> r = {1}, s = {0}')
       return 1, 0
   else:
       print(f'Расш. алг. Евклида (e1 = {e1}, e2 = {e2})')
       r, s = extended_euclidean_algorithm(e2, e1 % e2)
       print(f'Расш. алг. Евклида (e1 = {e1}, e2 = {e2}) --> r = {s}, s = {r - s * (e1
// e2)}')
       return s, r - s * (e1 // e2)

def decode(N, C1, C2, r, s):
   """Декодирует полученные сообщение в текст"""

   print("-- Дешифрование сообщения --")

   # Раздаляем закодированное сообщение на части и подготавливаем их
   raw_parts_C1 = C1.split("\n")
   parts_C1 = []
   for i in range(len(raw_parts_C1)):
       if raw_parts_C1[i].strip() != "":
           parts_C1.append(int(raw_parts_C1[i].strip()))

   raw_parts_C2 = C2.split("\n")
   parts_C2 = []
   for i in range(len(raw_parts_C2)):
       if raw_parts_C2[i].strip() != "":
           parts_C2.append(int(raw_parts_C2[i].strip()))

   if (len(parts_C1) != len(parts_C2)):
       print("Дешифрование прервано - Ввод C1 и C2 с разным количеством частей")
       exit(1)

   # Декодируем каждую часть
   original_message = ""
   for i in range(len(parts_C1)):
       y1 = parts_C1[i]
       y2 = parts_C2[i]

       int_decoded_part = pow(y1, r, N)*pow(y2, s, N) % N
       decoded_part = int_decoded_part.to_bytes(4, byteorder='big').decode('cp1251')
       original_message += decoded_part
       print(f'Декодирована часть C1 = {y1} и C2 = {y2} -----> {int_decoded_part} ----
-> {decoded_part}')

   return original_message

if __name__ == '__main__':
   # Описание варианта
```

```
N = 535598392051
e1 = 455341
e2 = 396971
C1 = """
    444982997352
    277831853272
    133187882628
    331361392426
    273206302188
    470299046774
    168157171491
    258737286129
    312335302650
    489235057221
    427689116872
    418723605534
    135022585485
    """
C2 = """
    358696089912
    360292494113
    91390259562
    534590606880
    193203217609
    166702058071
    68207231399
    487524624411
    325841328769
    533726724224
    369967614519
    247201359991
    478832067683
    """

print("-- Исходные данные --")
print(f'N = {N}')
print(f'e1 = {e1}')
print(f'e2 = {e2}')
print(f'C1 = \"{C1}\"')
print(f'C2 = \"{C2}\"')
print()

# Решаем уравнение  r*e1 + s*e2 = 1
print("-- Расширенный алгоритм Евклида --")
r, s = extended_euclidean_algorithm(e1, e2)
print(f'r = {r}')
print(f's = {s}')
print(f'r*e1 + s*e2 = {r}*{e1} + {s}*{e2} = {r*e1 + s*e2}')
print()

# Декодируем сообщение
original_message = decode(N, C1, C2, r, s)
print(f'\nОригинальное сообщение - \"{original_message}\"')
```

Результаты работы программы

```
C:\Python310\python.exe "D:\Учеба\4 курс\7 семестр\(ИБ) Информационная безопасность\information-security-labs\lab_2_3\lab_2.3.py"
-- Исходные данные --
N = 535598392051
e1 = 455341
e2 = 396971
C1 = "
        444982997352
        277831853272
        133187882628
        331361392426
        273206302188
        470299046774
        168157171491
        258737286129
        312335302650
        489235057221
        427689116872
        418723605534
        135022585485
        "
C2 = "
        358696089912
        360292494113
        91390259562
        534590606880
        193203217609
        166702058071
        68207231399
        487524624411
        325841328769
        533726724224
        369967614519
        247201359991
        478832067683
        "

-- Расширенный алгоритм Евклида --
Расш. алг. Евклида (e1 = 455341, e2 = 396971)
Расш. алг. Евклида (e1 = 396971, e2 = 58370)
Расш. алг. Евклида (e1 = 58370, e2 = 46751)
Расш. алг. Евклида (e1 = 46751, e2 = 11619)
Расш. алг. Евклида (e1 = 11619, e2 = 275)
Расш. алг. Евклида (e1 = 275, e2 = 69)
Расш. алг. Евклида (e1 = 69, e2 = 68)
Расш. алг. Евклида (e1 = 68, e2 = 1)
Расш. алг. Евклида (e1 = 1, e2 = 0) --> r = 1, s = 0
Расш. алг. Евклида (e1 = 68, e2 = 1) --> r = 0, s = 1
Расш. алг. Евклида (e1 = 69, e2 = 68) --> r = 1, s = -1
Расш. алг. Евклида (e1 = 275, e2 = 69) --> r = -1, s = 4
Расш. алг. Евклида (e1 = 11619, e2 = 275) --> r = 4, s = -169
Расш. алг. Евклида (e1 = 46751, e2 = 11619) --> r = -169, s = 680
Расш. алг. Евклида (e1 = 58370, e2 = 46751) --> r = 680, s = -849
Расш. алг. Евклида (e1 = 396971, e2 = 58370) --> r = -849, s = 5774
Расш. алг. Евклида (e1 = 455341, e2 = 396971) --> r = 5774, s = -6623
r = 5774
s = -6623
r*e1 + s*e2 = 5774*455341 + -6623*396971 = 1

-- Дешифрование сообщения --
Декодирована часть C1 = 444982997352 и C2 = 358696089912 -----> 4059228192 -----> сти
Декодирована часть C1 = 277831853272 и C2 = 360292494113 -----> 3789613554 -----> байт
Декодирована часть C1 = 133187882628 и C2 = 91390259562 -----> 4007799840 -----> ов,
Декодирована часть C1 = 331361392426 и C2 = 534590606880 -----> 3760254437 -----> а не
Декодирована часть C1 = 273206302188 и C2 = 193203217609 -----> 552728549 ----->  сче
Декодирована часть C1 = 470299046774 и C2 = 166702058071 -----> 4076333290 -----> тчик
Декодирована часть C1 = 168157171491 и C2 = 68207231399 -----> 4079022048 -----> у па
Декодирована часть C1 = 258737286129 и C2 = 487524624411 -----> 3940938478 -----> кето
Декодирована часть C1 = 312335302650 и C2 = 325841328769 -----> 3794673873 -----> в. С
Декодирована часть C1 = 489235057221 и C2 = 533726724224 -----> 3957712110 -----> ледо
```

```
70 Декодирована часть C1 = 427689116872 и C2 = 369967614519 -----> 3806393061 -----> вате
71 Декодирована часть C1 = 418723605534 и C2 = 247201359991 -----> 3959221742 -----> льно
72 Декодирована часть C1 = 135022585485 и C2 = 478832067683 -----> 740302880 -----> ,
73
74 Оригинальное сообщение – "сти байтов, а не счетчику пакетов. Следовательно,   "
75
76 Process finished with exit code 0
```

Вывод

В ходе лабораторной работы была совершена атака на алгоритм шифрования RSA методом бесключевого чтения, в следствии чего было декодировано исходное сообщение. В процессе выполнения была изучена атака на алгоритм шифрования RSA посредством метода бесключевого чтения, а также был изучен расширенный алгоритм Евклида.