МАТЕРИАЛЫ К ЗАДАНИЮ №35 – RSA

Постановка задачи

Пусть $M = (m_0, m_1, \ldots, m_{t-1})$ – сообщение длины $t \in \mathbb{N}$, причём для каждого $i \in \overline{0, t-1}$ символ m_i имеет шестнадцатеричный ASCII-код, который будем обозначать $\mathrm{Ord}(m_i)$. В итоге сообщению M ставится в соответствие число

$$\mathrm{Ord}(m_{t-1}) \parallel \ldots \parallel \mathrm{Ord}(m_1) \parallel \mathrm{Ord}(m_0),$$

где $\parallel$ – символ конкатенации, то есть склеивания битовых последовательностей. К примеру, в предложении «abcd» символ «a» имеет ASCII-код 97 (или `0x61`), символ «b» – 98 (или `0x62`), символ «c» – 99 (или `0x63`), символ «d» – 100 (или `0x64`). Значит, этому предложению будет соответствовать шестнадцатеричное число `0x64636261`, которое в десятичной системе счисления записывается в виде 1684234849.

Каждое сообщение, зашифрованное с помощью криптосистемы RSA, в числовом представлении не должно превосходить значение ключевого параметра n. В противном случае сообщение разбивается на блоки подходящего размера, каждый из которых зашифровывается отдельно.

В распоряжении криптоаналитика имеются:

- пара чисел (e, n), соответствующих открытому ключу криптосистемы RSA;
- перехваченные блоки закрытого текста y.

$$n = 40414990965097589204096 69, \qquad e = 5012342898331229.$$

y: `0x1df441a3025ba0894d996` `0x222596a3860e0e7390f95` `0x6babda245e863eeca72a`
`0x1f2adbe40a89e44bf26e2` `0x26502401ce19788f0dd79` `0x27b31051691d02c41644e`
`0x2a6dffa9163de78a19722` `0x1c06fe0ffb26d9c7988da` `0x2a4689fc1130e84294da7`
`0x2c7d0a32d4842aabfa59f` `0x247a0392625d92441d6a5` `0x73f6d735a75ad0a29fda`
`0xc0b42969cbd1c6bab1e` `0x278f53c05ed7641c2193e` `0x1f660927099f915919b7f`
`0xdcac589b4bd612ed129c` `0x1397a3630b237d796c758`

Необходимо выполнить следующее:

- изучить и описать схему работы криптосистемы RSA;
- написать программную реализацию RSA на языке программирования C или Python, способную работать с числами произвольной разрядности;
- изучить и описать метод Ферма факторизации целых чисел;
- написать программную реализацию метода Ферма на языке программирования C или Python, способную работать с числами произвольной разрядности;
- по известному открытому ключу (e, n) криптосистемы RSA определить закрытый ключ (d, n) и с его помощью восстановить открытый текст x, из которого был получен закрытый текст y.

Полезные источники

1. *Глухов М.М., Круглов И.А., Пичкур А.Б., Черемушкин А.В.* Введение в теоретико-числовые методы криптографии: Учебное пособие. – СПб.: Издательство «Лань», 2011.
   Здесь, помимо очевидно необходимого описания метода Ферма, излагается алгоритм извлечения квадратного корня, который понадобится реализовать на определённом этапе работы.
2. *Лось А.Б, Нестеренко А.Ю., Рожков М.И.* Криптографические методы защиты информации: учебник для академического бакалавриата. – 2-е изд., испр. – М.: Издательство Юрайт, 2019.
3. *Черемушкин А.В.* Лекции по арифметическим алгоритмам в криптографии. – М.: МЦНМО, 2002.