МАТЕРИАЛЫ К ЗАДАНИЮ №34 – 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.

$$n = 376539155385972491237017 9, \qquad e = 4957756715208581.$$

y: `0x2e7953864ceadab169045` `0x22a0de2b287c3117a2305` `0x2bfd0773b5f630c60a930`
`0x201edbff952f21d942410` `0x2432238697c1fb4c0c353` `0x258e7476620092a6098fe`
`0x2bb3cdbe7793dfb8ddda7` `0x8580f52df51f0ce0f86a` `0x5608d71160542cb11370`
`0xcddb3006508569ef1dcf` `0x16d9e1a5ffa6109bebac9` `0x31ce9138e19543f71f74b`
`0x11a1110b625c943785586` `0x1b5b9fcf7c8d0fd80f899` `0x236edc822ee70965b8162`
`0x24a3a3e35f4a3862f9f93` `0x2752575df0697fb1fc88`

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.