**Методы защиты информации.**

**Лабораторная работа №3**

|  |
| --- |
| **Компьютерная реализация алгоритмов криптографии с открытым ключом на примере RSA.** |

Реализовать программные средства шифрования и дешифрования текстовых файлов при помощи алгоритма RSA.

**ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

**Алгоритм RSA**

Алгоритм, разработанный Ривестом, Шамиром и Адлеманом, использует выражения с экспонентами. Данные шифруются блоками, каждый блок рассматривается как число, меньшее некоторого числа n. Шифрование и дешифрование имеют следующий вид для некоторого незашифрованного блока М и зашифрованного блока С.

С = Ме (mod n)

M = Cd (mod n) = (Me)d (mod n) = Med (mod n)

Как отправитель, так и получатель должны знать значение n. Отправитель знает значение е, получатель знает значение d. Таким образом, *открытый ключ* есть KU = {e, n} и *закрытый ключ* есть KR = {d, n}. При этом должны выполняться следующие условия:

1.           Возможность найти *е*, *d* и *n* такие, что *Med = M mod n* для всех    М < n .

2.           Относительная легкость вычисления Ме и Сd для всех значений   М < n.

3.           Невозможность определить d, зная е и n.

**Рассмотрим некоторые математические понятия, свойства и теоремы, которые позволят нам определить e, d и n.**

1.           Если (а · b) ≡ (a · c) mod n, то b ≡ c mod n,

если а и n взаимнопростые, т.е gcd (a, n) = 1.

2.           Обозначим Zp - все числа, взаимнопростые с p и меньшие p. Если p - простое, то Zp - это все остатки. Обозначим w-1 такое число, что w · w-1 ≡ 1 mod p.

Тогдаw Zp z: w · z ≡1 mod p

3.           Определим функцию Эйлера следующим образом: Φ(n) - число положительных чисел, меньших n и взаимнопростых с n. Если p - простое, то Φ(р) = p-1.

Если p и q - простые, то Φ(p · q) = (p-1) · (q-1).

4.           *Теорема Ферма*.

an-1 ≡1 mod n, если n - простое.

5.           *Теорема Эйлера*.

aΦ(n) ≡1 mod n для всех взаимнопростых a и n.

Теперь рассмотрим все элементы *алгоритма RSA*.

|  |  |
| --- | --- |
| p, q - два простых целых числа | -открыто, вычисляемо. |
| n = p · q | - закрыто, вычисляемо. |
| d, gcd (Φ(n), d) = 1; | - открыто, выбираемо. |
| 1 < d < Φ(n) |
| е ≡ d-1 mod Φ(n) | - закрыты, выбираемы. |

*Закрытый ключ* состоит из {d, n}, *открытый ключ* состоит из {e, n}. Предположим, что пользователь А опубликовал свой *открытый ключ*, и что пользователь В хочет послать пользователю А сообщение М. Тогда В вычисляет С = Ме (mod n) и передает С. При получении этого зашифрованного текста пользователь А дешифрует вычислением                   М = С d (mod n).

Суммируем *алгоритм RSA*:

**Создание ключей**

|  |
| --- |
| 1)    Выбрать простые р и q |
| 2)    Вычислить n = p · q |
| 3)    Выбрать d     gcd (Φ(n), d) = 1; 1 < d < Φ(n) |
| 4)    Вычислить е     е = d-1 mod Φ(n) |
| 5)    *Открытый ключ* KU = {e, n} |
| 6)    *Закрытый ключ* KR = {d, n} |

**Шифрование**

|  |
| --- |
| 1)    Незашифрованный текст: М < n |
| 2)    Зашифрованный текст: С = М е (mod n) |

**Дешифрование**

|  |
| --- |
| 1)    Зашифрованный текст: С |
| 2)    Незашифрованный текст: М = Сd (mod n) |

**ЗАДАНИЕ:**

1)                Изучить теоретические сведения.

2)                Создать программы, читающие данные из файла и шифрующие (дешифрующие) их помощью алгоритма RSA.

**По результатам каждой работы студентом должен быть представлен и защищен отчет.**

**Содержание отчета включает:**

1. Титульный лист

2. Постановку задачи и описание использованных алгоритмов.

3. Блок – схему алгоритма.

4. Распечатку скриншотов результатов ввода данных и исполнения программы.

5. Распечатку программного кода.

6. Вывод.