

АЛГОРИТМ

Райвеста-Шамира-Адлемана
(Rivest, Shamir и Adleman)

RSA

Область применения

- Шифрование;
- Цифровое подписывание.

RSA и DH

Алгоритм RSA во многом схож с алгоритмом Диффи-Хеллмана.

Принципиальное различие:

- Алгоритм DH основан на использовании односторонней функции;
- Алгоритм RSA основан на использовании односторонней функции с лазейкой.

Лазейка для RSA

- Вычисления производятся по модулю составного числа $n=r*q$, где r и q - простые числа;
- Зная публично известные n и e , мы можем найти $m^e \pmod n$ по заданному m , но не наоборот. При этом, зная, как n раскладывается на множители; выполнить обратную операцию очень легко. Разложение числа n на множители и есть "лазейка". (Если мы знаем эту информацию, то можем легко выполнить обратное действие, а если не знаем, то не можем).

Китайская теорема об остатках (Сунь Цзе 1 век д.н.э.)

Числа по модулю n — это $0, 1, \dots, n - 1$. Они не образуют конечное поле, как в том случае, если бы n было простым числом. Математики обозначают это множество чисел как Z_n и называют его кольцом.

Для каждого x из Z_n можно легко вычислить пару $(x \bmod p, x \bmod q)$. Китайская теорема об остатках утверждает, что можно выполнить и обратную операцию: зная пару $(x \bmod p, x \bmod q)$, можно восстановить исходное значение x .

Доказательство. Единственность решения

Для упрощения записи введем обозначение $(a, b) := (x \bmod p, x \bmod q)$. Вначале покажем, что восстановление исходного значения x вообще возможно, а затем приведем алгоритм его вычисления.

Чтобы вычислить x по заданным (a, b) , следует убедиться, что в \mathbb{Z}_n не существует второго такого числа x' , для которого $x' \bmod p = a$ и $x' \bmod q = b$. В противном случае x и x' привели бы к появлению одной и той же пары (a, b) , и ни один алгоритм не смог бы распознать, какое из этих значений является исходным.

Пусть $d := x - x'$ — это разность чисел, которым соответствует одна и та же пара (a, b) .

Имеем $(d \bmod p) = (x - x') \bmod p = (x \bmod p) - (x' \bmod p) = a - a = 0$, а следовательно, d кратно p .

Аналогичным образом получаем, что d кратно q .

Отсюда следует, что d является кратным НОК(p, q), так как НОК это, наименьшее общее кратное. Поскольку p и q — это неодинаковые простые числа, $\text{НОК}(p, q) = pq = n$, а значит, $x - x'$ кратно n .

Но и x и x' лежат в диапазоне $0, 1, \dots, n - 1$, поэтому разность $x - x'$, кратная n , находится в диапазоне: $-n + 1, \dots, n - 1$. Единственным возможным значением такой разности, кратным n , является $x - x' = 0$, или $x = x'$. Это доказывает, что для любой заданной пары (a, b) существует не более одного x , удовлетворяющего условиям теоремы. Остается лишь найти значение x .

Доказательство. Существование решения

Самым удобным способом вычисления x является формула Гарнера:

$$x = (((a - b)(q^{-1} \bmod p)) \bmod p) * q + b (*).$$

Здесь множитель $(q^{-1} \bmod p)$ — это константа, которая зависит только от p и q . Мы можем выполнять деление по модулю p , а значит, и вычислять $(1/q \bmod p)$, что является всего лишь другой формой записи выражения $(q^{-1} \bmod p)$.

Вначале покажем, что x находится в диапазоне $0, \dots, n - 1$.

Очевидно, $x \geq 0$.

Обозначим за t выражение $((a - b)(q^{-1} \bmod p)) \bmod p$. Значение t должно попадать в диапазон $0, \dots, p - 1$, так как является результатом вычисления по модулю p .

Если $t \leq p - 1$, тогда $tq \leq (p - 1)q$ и $x = tq + b$ (из $(*)$) $\leq (p - 1)q + (q - 1) = pq - 1 = n - 1$.

Очевидно, значение x действительно находится в диапазоне $0, \dots, n - 1$.

Доказательство. Существование решения

Теперь покажем, что найденное значение x дает правильный результат и по модулю p , и по модулю q .

$$x \bmod q = (((a-b)(q^{-1} \bmod p)) \bmod p) * q + b \bmod q = (K * q + b) \bmod q = b \bmod q = b.$$

Выражение $((a - b)(q^{-1} \bmod p)) \bmod p$, которое умножается на q — это некоторое целое число K , но при выполнении операций по модулю q любое кратное q можно отбросить.

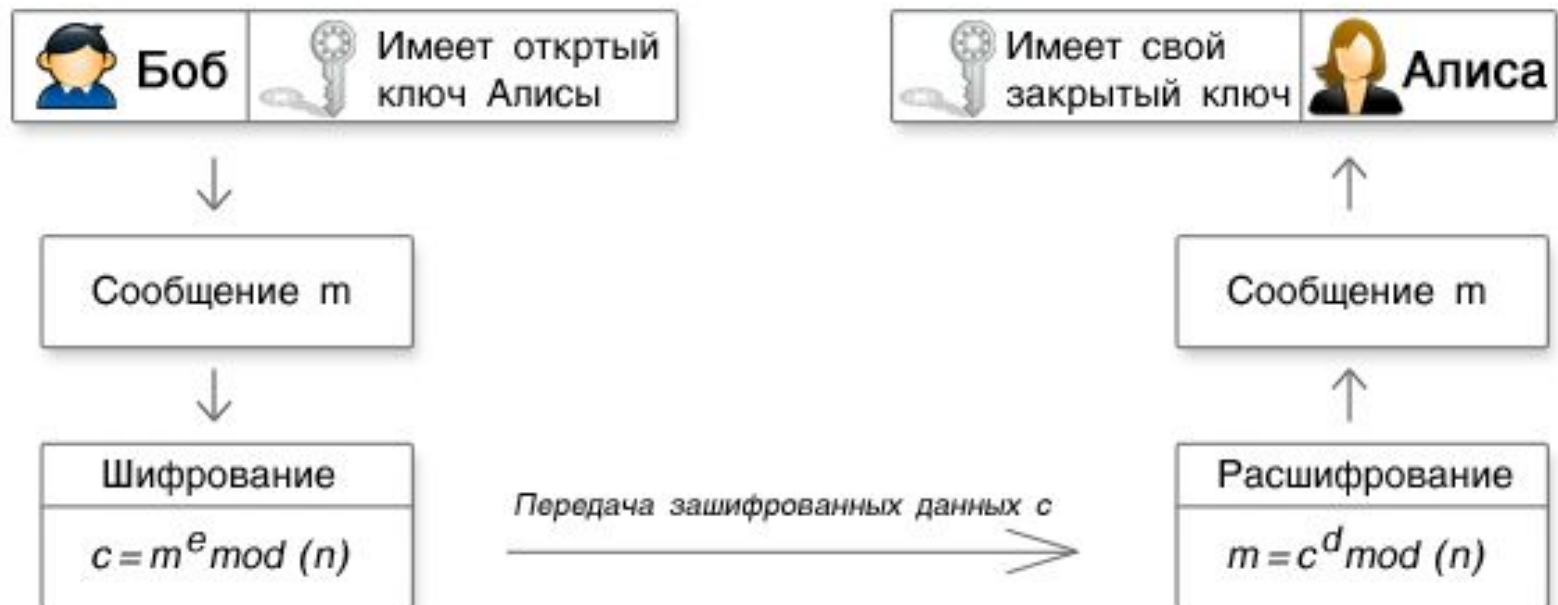
$$x \bmod p = (((a - b)(q^{-1} \bmod p)) \bmod p) * q + b \bmod p = ((a - b) q^{-1}) * q + b \bmod p = ((a - b)(q^{-1} * q) + b) \bmod p = (a - b + b) \bmod p = a \bmod p = a.$$

Избавляемся от нескольких лишних операторов по $\bmod p$, изменяем порядок умножения и замечаем, что $(q^{-1} * q) = 1 \pmod p$.

Таким образом формула Garnera дает результат x , который лежит в нужном диапазоне и для которого $(a, b) = (x \bmod p, x \bmod q)$. Поскольку мы уже знаем, что такое решение может быть только одно, результат формулы Garnera полностью решает китайскую теорему остатков.

RSA. Шифрование.

Сценарий: Боб посылает Алисе сообщение m



Генерация ключей

- Выбираются два разных случайных простых числа p и q (порядка 1024 бит каждое) и вычисляется $n=pq$;
- Вычисляется число $t = \text{НОК}(p-1, q-1)$;
- Выбирается целое число e взаимно простое с t и вычисляется число d такое, что $ed=1 \pmod{t}$;
- Пара (n, e) образует открытый ключ.
- Значения (p, q, t, d) образуют закрытый ключ и сохраняются в секрете человеком, который сгенерировал ключ RSA.

Шифрование. Расшифрование.

Чтобы зашифровать сообщение m , отправитель генерирует зашифрованный текст $c := m^e \pmod{n}$.

Чтобы расшифровать зашифрованный текст c , получатель вычисляет $c^d \pmod{n}$.

Это эквивалентно значению

$$\begin{aligned} (m^e)^d \pmod{n} &= m^{ed} \pmod{n} = m^{kt+1} \pmod{n} = (m^t)^k * m \pmod{n} = \\ &= m \pmod{n}, \end{aligned}$$

где k - некоторое целое число, которое всегда существует.

Таким образом, получатель может расшифровать зашифрованный текст m^e , чтобы получить открытый текст m .

Пример. Генерация ключей.

Выберем $p=3$ и $q=11$.

Определим $n=3*11=33$.

Найдем $(p-1)*(q-1)=20$. Следовательно в качестве e выберем любое число, которое является взаимно простым с числом 20, например $e=3$.

Выберем число d . В качестве такого числа может быть взято любое число, для которого выполняется равенство

$(d*3) \bmod 20 = 1$, например $d=7$.

- Пара **(33, 3)** образует открытый ключ.
- Значения **(3, 11, 20, 7)** образуют закрытый ключ

Пример. Шифрование.

Т.к. в качестве n было взято число 33, можем шифровать буквы русского алфавита производя вычисления по модулю 33.

Зашифруем слово «бал», переведем буквы в соответствующие числовые значения (2,1,13)

$$C1=(2^3)\text{mod}33=8\text{mod}33=8;$$

$$C2=(1^3)\text{mod}33=1\text{mod}33=1;$$

$$C3=(13^3)\text{mod}33=2197\text{mod}33=19.$$

Зашифрованное слово (8,1,12) или «жас»

Пример. Расшифрование.

Зашифрованное слово (8,1,19) или «жас»

$$M1=(8^7)\bmod 33=2097152\bmod 33=2,$$

$$M2=(1^7)\bmod 33=1\bmod 33=1,$$

$$M3=(19^7)\bmod 33=893871739\bmod 33=13.$$

Расшифрованное слово (2,1,13) или «бал»

RSA. Электронная подпись.



RSA. Создание электронной подписи.

Для сообщения m создается цифровая подпись s на основании секретного ключа пользователя A

$$s = S_A(m) = m^d \pmod{n};$$

Пара (m, s) передается пользователю B .

RSA. Проверка электронной подписи.

- Применить открытый ключ пользователя A к паре (m, s)
- $m' = P_A(s) = s^e \pmod{n}$;
- Проверить подлинность подписи сравнив m и m' .
- Если $m = m'$ – верификация проходит успешно.

Проблемы использования RSA

- **Четкая математическая структура**

(если пользователь А подпишет цифровой подписью два сообщения – m_1 и m_2 , пользователь Б сможет вычислить, какой должна быть подпись пользователя А для сообщения $m_3 := m_1 m_2 \pmod{n}$);

- **Шифрование сообщения малого ($<n$) размера**

(если при возведении в степень e символ сообщения принимает значение $<n$, то для его обратного преобразования достаточно выполнить извлечение корня степени e , вычисления по модулю не производится, что облегчает задачу злоумышленника);

- **Необходимость применять в качестве ключей шифрования и ЭП не пересекающиеся множества;**

- **Атаки основанные на методах решения полиномиальных уравнений по модулю n .**

Задание по лекции

1. Записать открытый и закрытый ключи алгоритма RSA, если

- $p=2$;
- $q=13$;
- $e=5$.

2. Зашифровать (и расшифровать) с помощью алгоритма RSA слово «сake» используя английский алфавит и ключи из п.1.