Алгоритмы. HW#8

Тураев Тимур, 504 (SE)

1 *Алёна отправила сообщение m, зашифрованное через RSA, трем людям. Для каждого человека определено свое $N_i = p_i \cdot q_i$, но везде одинаковое $e = 3$. Найдите сообщение Алёны.*

Будем считать все N_i взаимно простыми, иначе можно просто посчитать gcd каких-нибудь не взаимно простых N_i, N_j, каждое поделить на их gcd, получить разложение, посчитать $\phi(N_i)$, найти обратный тройке по этому модулю… ну и все сломается.

Итак, известно, что

$$c_1 = m^3 \mod N_1$$
$$c_2 = m^3 \mod N_2$$
$$c_3 = m^3 \mod N_3$$

И все N_i взаимно простые. Найдем по КТО такое $M \in \mathbb{Z}^*_{N_1 N_2 N_3}$, что

$$M = c_1 \mod N_1$$
$$M = c_2 \mod N_2$$
$$M = c_3 \mod N_3$$

Тогда $M = m^3 \mod N_1 N_2 N_3$, но так как наше сообщение было из $\mathbb{Z}_{\min(N_1, N_2, N_3)}$, то $m^3 < N_1 N_2 N_3$. Берем кубический корень из M и получаем исходное сообщение.

2 *RSA. В распоряжении взломщика появился "волшебный"оракул. Для любого открытого ключа (N, e) оракул может взломать 1% из возможных зашифрованных сообщений. Придумайте алгоритм, который взламывает любое сообщение по матожиданию за $O(poly(\log n))$.*

Все действия по модулю N.

Идея такая: давайте скормим шифрованное сообщение оракулу и проверим, а корректно ли он его расшифровал – это можно сделать обратным шифрованием, если получили исходную шифровку, то все хорошо. Однако проблема в том, что оракул дешифрует лишь часть возможных шифров, значить надо как-то преобразовать исходный шифр, проверить правильность его дешифровки и обратить преобразование.

Это можно сделать, умножив по модулю за *известное* сообщение (и обратимое – чтобы существовал обратный к нему элемент).

На входе имеем: (N, e) и $c = Enc(m)$ – публичный ключ и зашифрованное сообщение.

1. Выбираем какой-нибудь обратимый $y \in \mathbb{Z}^*_N$
2. Скармливаем оракулу сообщение $Enc(y) \cdot c$ – на выходе получаем $Decrypt(Enc(y) \cdot c) = m'$
3. Далее проверяем, правильна ли была расшифровка: $Enc(m') == Enc(y) \cdot c$?

4.a Если да, то выводим расшифрованное: $y^{-1} \cdot m' = Decrypt(c) = m$

4.b Иначе *goto* 1

Разбираем по пунктам: Сначала предположим, что исходное сообщение m было обратимым. Тогда его шифр $Enc(m)$ тоже обратим.

1. Тут выбирается обратимый элемент группы. Это делается просто: выбираем какой-нибудь элемент группы, проверяем его gcd, если не 1, то повторяем действия. Очевидно, что случайная величина, равная числу запусков цикла в этом пункте, имеет геометрическое распределение (номер первого успеха), поэтому матожидание числа запусков равно $1/p$, где p – вероятность успеха, которая равна $p = \frac{\phi(N)}{N}$. Значит среднее число запусков есть

$$\frac{N}{\phi(N)} = \frac{pq}{(p-1)(q-1)} = \frac{p}{p-1} \cdot \frac{q}{q-1} = \left(1 + \frac{1}{p-1}\right) \cdot \left(1 + \frac{1}{q-1}\right) < 2$$

Оценка верна для больших p и q. Или менее строго: матожидание не больше 4.

4.b Оценим число итераций этого (внешнего) цикла.

$Enc(y) \cdot c = Enc(y) \cdot Enc(m') = Enc(y \cdot m')$ – это достаточно очевидно, тут просто возведения в степень по модулю.

Так как и y и были случайными обратимыми элементами группы, то их произведение тоже случайный обратимый элемент (из теории групп: $\mathbb{Z}_{\mathbb{N}}^*\mathbb{Z}_{\mathbb{N}}^* = \mathbb{Z}_{\mathbb{N}}^*$ по Минковскому), значит с матожиданием в (примерно) 100 запусков оракул расшифрует шифр.

Почему примерно? Тут есть еще одна проблема: пусть оракул дешифрует **все** элементы из $\mathbb{Z}_{\mathbb{N}} - \mathbb{Z}_{\mathbb{N}}^*$. Сколько же он расшифрует элементов из обратимых ($\mathbb{Z}_{\mathbb{N}}^*$). Нетрудно понять, что это почти те же 1%: посчитаем долю необратимых во всей группе

$$\frac{|\mathbb{Z}_{\mathbb{N}} - \mathbb{Z}_{\mathbb{N}}^*|}{|\mathbb{Z}_{\mathbb{N}}|} = \frac{N - \phi(N)}{N} = \frac{pq - (p-1)(q-1)}{pq} = \frac{p+q-1}{pq}$$

А это число сильно меньше 1% всех сообщений при больших p и q.

Итак, расшифровывать обратимые сообщения мы умеем.

Что же делать с необратимыми сообщениями m? Предположим, мы знаем m. Тогда мы бы смогли найти $gcd(m, N) = k$ и $m = k \cdot m'$ где m' – уже обратимое сообщение. Тогда $Enc(m) = c = k^e \cdot m'^e$ и m'^e – тоже обратимое зашифрованное сообщение. Значит, $Decrypt(c) = k \cdot Decrypt(m'^e)$

Теперь понятно как взломать обратимое сообщение: ищем $gcd(N, c) = k^e$, делим c на gcd, взламываем c/gcd, и выводим ответ: взломанное сообщение умноженное на k, где k – корень e-ой степени из gcd.

3 *RSA. Пусть есть N, e и d. Пусть $e = 3$. Разложить N на множители.* С одной стороны известно, что N раскладывается на множители (пока неизвестные)

$$N = p \cdot q$$

С другой стороны, для d известно (по определению)

$$d = 3^{-1} \mod (p-1) \cdot (q-1)$$

То есть

$$3 \cdot d = 1 \mod (p-1) \cdot (q-1)$$

$$3 \cdot d - 1 = 0 \mod (p-1) \cdot (q-1)$$

$$3 \cdot d - 1 = k \cdot (p-1) \cdot (q-1)$$

$$3 \cdot d - 1 = k \cdot (pq - p - q + 1)$$

$$3 \cdot d - 1 = k \cdot (N - p - q + 1)$$

Осталось понять, что $k \in \{1, 2\}$: действительно, $d \in [1 \ldots (p-1)(q-1) - 1]$, тогда $3d - 1 \in [2 \ldots 3 \cdot (p-1)(q-1) - 4]$

$$p + q = N + 1 - (3 \cdot d - 1)/k$$

Дальше решаем две системы, как с практики: известны сумма чисел и их произведение и находим p и q (у одной из них решений не будет или будут решения в отрицательных p и q, это нам не подходит)