ФИТ ПОИТ 3 курс 5 группа Сойкель Александр

**1. Используя некоторый вектор S = (103, 107, 211, 430, 863, 1716, 3449, 6907, 13807, 27610), вычислить ключи для зашифрования и расшифрования сообщений(относится к ранцевому алгоритму).**

**Решение**:

Вектор S – сверхвозрастающий рюкзачный вектор, является закрытым ключом.

Сумма элементов последовательности:

103+107+211+430+863+1716+3449+6907+13807+27610 = 55203.

∑𝑆=55203

Далее выберем простое число n, превосходящее полученное нами значение суммы

Это число 55207 => n=55207

Выберем также простое число *a* из интервала [1,n]

Допустим,что a =5

∑𝑆, n и a образуют закрытый ключ.

Чтобы сгенерировать открытый ключ, построим последовательность β, умножая каждый элемент из последовательности *w* на *r* по модулю *q*.

Получим S =(515,535,1055,2150,4315,8580,17245,34535,13828,27636)

**2. Найти первообразные корни (если они существуют) чисел (р): 19, 23, 27, 31, 37.**

Для того, чтобы найти все первообразные корни числа, необходимо найти наименьший первообразный. Далее необходимо будет найти все такие γ, что (γ, ϕ(p)) и 1 ≤ γ ≤ ϕ(p).

19: По модулю 19 всего ϕ(ϕ(19)) = 6 первообразных корней, не превосходящих 19.

Первым первообразным корнем является число 2, так как оно является взаимнопростым с 19, а также 2, 22, . . . ,2ϕ(18). попарно несравнимы по модулю 19

Найдем все такие γ, что (γ, ϕ(19)) и 1 ≤ γ ≤ 18. Таких ровно 6: 1, 5, 7, 11, 13, 17

Отсюда получаются следующие первообразные корни: 21≡2𝑚𝑜𝑑19; 25≡13𝑚𝑜𝑑19; 27≡14𝑚𝑜𝑑19; 211≡15𝑚𝑜𝑑19; 213≡3𝑚𝑜𝑑19; 217≡10𝑚𝑜𝑑19;

Первообразные корни: 2, 3, 10, 13, 14, 15

23: По модулю 23 всего ϕ(ϕ(23)) = 10 первообразных корней, не превосходящих 23.

Первым первообразным корнем является число 5, так как оно является взаимнопростым с 23, а также 5, 52, . . . ,5ϕ(23). попарно несравнимы по модулю 23.

Найдем все такие γ, что (γ, ϕ(23)) и 1 ≤ γ ≤ 22. Таких ровно 10: 1, 3, 5, 7, 9, 13, 15, 17, 19, 21

Отсюда получаются следующие первообразные корни: 51≡5𝑚𝑜𝑑23; 53≡10𝑚𝑜𝑑23;

55≡20𝑚𝑜𝑑23; 57≡17𝑚𝑜𝑑23; 59≡11𝑚𝑜𝑑23; 513≡21𝑚𝑜𝑑23; 515≡19𝑚𝑜𝑑23; 517≡15𝑚𝑜𝑑23; 519≡7𝑚𝑜𝑑23; 521≡14𝑚𝑜𝑑23;

Первообразные корни: 5, 10, 11, 13, 14, 15, 17, 19, 20, 21

27: не имеет первообразных корней

31: По модулю 31 всего ϕ(ϕ(31)) = 8 первообразных корней, не превосходящих 31.

Первым первообразным корнем является число 3, так как оно является взаимнопростым с 31, а также 3, 32, . . . ,3ϕ(31). попарно несравнимы по модулю 31.

Найдем все такие γ, что (γ, ϕ(31)) и 1 ≤ γ ≤ 30. Таких ровно 8: 1, 3, 7, 11, 13, 17, 19, 23

Отсюда получаются следующие первообразные корни: 31≡3𝑚𝑜𝑑31; 37≡17𝑚𝑜𝑑31; 311≡13𝑚𝑜𝑑31; 313≡24𝑚𝑜𝑑31; 317≡22𝑚𝑜𝑑31; 319≡12𝑚𝑜𝑑31; 323≡11𝑚𝑜𝑑31; 329≡21𝑚𝑜𝑑31;

Первообразные корни: 3, 11, 12, 13, 17, 21, 22, 24

37: По модулю 37 всего ϕ(ϕ(37)) = 12 первообразных корней, не превосходящих 37.

Первым первообразным корнем является число 2, так как оно является взаимнопростым с 37, а также 2, 22, . . . ,2ϕ(37). попарно несравнимы по модулю 37.

Найдем все такие γ, что (γ, ϕ(37)) и 1 ≤ γ ≤ 36. Таких ровно 12: 1, 5, 7, 11, 13, 17, 19, 23, 25, 29, 31, 35

Отсюда получаются следующие первообразные корни: 21≡2𝑚𝑜𝑑37; 25≡32𝑚𝑜𝑑37; 27≡17𝑚𝑜𝑑37; 211≡13𝑚𝑜𝑑37; 213≡15𝑚𝑜𝑑37; 217≡18𝑚𝑜𝑑37; 219≡35𝑚𝑜𝑑37; 223≡5𝑚𝑜𝑑37; 225≡20𝑚𝑜𝑑37; 229≡24𝑚𝑜𝑑37; 231≡22𝑚𝑜𝑑37; 235≡19𝑚𝑜𝑑37;

Первообразные корни: 2, 5, 13, 15, 17, 18, 19, 20, 22, 24, 32, 35

**3. Пользователю системы RSA с его собственными ключевыми параметрами n = 3337, е = 79 передано зашифрованное сообщение С, состоящее из трех блоков: чисел, соответствующих порядковым номерам Ваших фамилии, имени и отчества. Расшифровать это сообщение (взломав систему RSA пользователя). C = {19, 1, 3}**

Для взлома системы RSA пользователя можно воспользоваться выражением 3.1.

e ⋅ d ≡ 1 mod φ(n) (3.1)

Исходя из выражения 3.1: 𝑑= 79−1𝑚𝑜𝑑3220=1019 . Зная d, можно посчитать символы исходного сообщения на основе выражения 3.2.

𝑚𝑖 = (A𝑖)𝑑 mod 𝑛 (3.2)

𝑚1 = (A1)𝑑 mod 𝑛=19^1019 𝑚𝑜𝑑 3337= 2521

𝑚2 = (A2)𝑑 mod 𝑛=1^1019 𝑚𝑜𝑑 3337= 1

𝑚3 = (A3)𝑑 mod 𝑛=3^1019 𝑚𝑜𝑑 3337= 2140

**4. Положим, что в системе применяется алгоритм шифрования/ расшифрования Эль-Гамаля. Известны некоторые параметры системы: р = 167, g = 5, y = g^29 mod p = 55 mod p.**

Используя указанные и недостающие (выбрать самостоятельно) параметры, зашифровать свое имя (в любом языке) в предположении:

**а)** первая буква алфавита соответствует числу 0 и т. д., б) первая буква алфавита соответствует числу 1.

Для того, чтобы зашифровать сообщение, необходимо, во-первых, вычислить y, используя формулу 4.1

𝑦=𝑔^𝑥 𝑚𝑜𝑑 𝑝 (4.1)

Из анализируемого выражения следует, что x=29, а y = 55, поскольку

55 mod 167=55.

Исходя из этого, можно установить, что открытый ключ будут составлять числа p, g, y, а именно 167, 5, 55. Блок шифротекста будет состоять из двух чисел: 𝑎𝑖,𝑏𝑖

𝑎𝑖=𝑔^𝑘 𝑚𝑜𝑑 𝑝 (4.2)

𝑏𝑖=(𝑦^k ∗ 𝑚𝑖) 𝑚𝑜𝑑 𝑝 (4.3)

а) Зашифруем сообщение ‘sasha’, при этом примем, что букве ‘a’ соответствует число 0.

𝑎1=5^81 𝑚𝑜𝑑 167= 20

𝑏1=(55^81∗18) 𝑚𝑜𝑑 167=43

𝑎2=5^43 𝑚𝑜𝑑 167= 50

𝑏2=(55^43∗0) 𝑚𝑜𝑑 167=0

𝑎3=5^97 𝑚𝑜𝑑 167= 41

𝑏3=(55^97\*18) 𝑚𝑜𝑑 167=69

𝑎4=5^29 𝑚𝑜𝑑 167= 55

𝑏4=(55^29∗7) 𝑚𝑜𝑑 167=146

𝑎5=5^31 𝑚𝑜𝑑 167= 126

𝑏5=(55^31∗0) 𝑚𝑜𝑑 167=0

(2043, 5000, 4169, 55146, 1260)

б) Зашифруем сообщение ‘sasha’, при этом примем, что букве a соответствует число 1.

𝑎1=5^81 𝑚𝑜𝑑 167= 20

𝑏1=(55^81∗19) 𝑚𝑜𝑑 167=166

𝑎2=5^43 𝑚𝑜𝑑 167= 50

𝑏2=(55^43∗1) 𝑚𝑜𝑑 167=142

𝑎3=5^97 𝑚𝑜𝑑 167= 41

𝑏3=(55^97 \* 19) 𝑚𝑜𝑑 167=120

𝑎4=5^29 𝑚𝑜𝑑 167= 55

𝑏4=(55^29∗8) 𝑚𝑜𝑑 167=143

𝑎5=5^31 𝑚𝑜𝑑 167= 126

𝑏5=(55^31∗1) 𝑚𝑜𝑑 167=110

(20166, 59142, 41120, 55143, 126110)

**5.Сколько попыток нужно сделать, чтобы с вероятностью более 0,5 (0,7; 0,8; 0,9) обнаружить коллизию при длине хеша (l) 64 (128;256; 512) бит?**

Для вычисления вероятности обнаружения коллизий воспользуемся формулой

𝑃(𝐴𝑛)≈exp⁡(−𝑛2/2𝑚)

Например, в случае с вероятностью 0,5 можно получить равенство 5.2

(𝑛22𝑚)=𝑙𝑛2

В итоге, можно использовать формулу для вычисления искомого значения числа попыток.

𝑛=(2𝑚∗𝑙𝑛2)1/2 (5.3)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| l | m | 0,5 | 0,7 | 0,8 | 0,9 |
| 64 | 1.85∗10^19 | 5.1∗10^9 | 6.7∗10^9 | 7.7∗10^9 | 9.2∗10^9 |
| 128 | 3.4∗10^38 | 2.2∗10^19 | 2.9∗10^19 | 3.3∗10^19 | 4∗10^19 |
| 256 | 1.2∗10^77 | 4.0∗10^38 | 5.4∗10^38 | 6.2∗10^38 | 7.4∗10^38 |
| 512 | 1.3∗10^154 | 1.4∗10^77 | 1.77∗10^77 | 2.1∗10^77 | 2.5∗10^77 |

**6. Рассчитать общую длину (L') хешируемого сообщения после предварительной стадии на основе алгоритма MD, если объем (L) исходного сообщения составлял: 0; 484; 512; 1000; 2000; 16000 бит. Какова в каждом случае будет длина хеша?**

Входное сообщение «дополняется» (расширяется) так, чтобы его длина (в битах) была конгруэнтной к 448 по модулю 512. Это значит, что сообщение начальной длиной L бит расширяется так, что остаются незаполненными всего лишь 64 бита, чтобы итоговая длина

L' была кратной 512.

|  |  |  |
| --- | --- | --- |
| L (битов) | L’ (битов) | Hash (битов) |
| 0 | 512 | 128 |
| 484 | 1024 | 128 |
| 512 | 1024 | 128 |
| 1000 | 1536 | 128 |
| 2000 | 2560 | 128 |
| 16000 | 16384 | 128 |

**7. Входное сообщение (прообраз) состоит из**

**а) вашего имени,**

**б) ваших фамилии\_имени\_отчества (алфавит – на свой выбор).**

**Используя представление сообщения в кодах ASCII, представить**

**в табличной форме (как выше в примере 5) содержание каждого 32-**

**битного подблока расширенного входного сообщения**

а) Исходное сообщение M=SASHA или в кодах ASCII – это двоичные (01010011 01000001 01010011 01001000 01000001). Длина L хешируемого сообщения составила 40 битов. Эту длину необходимо расширить до 448 бит. Для этого необходимо добавить один «1» и 407 «0». Далее в последнюю часть из 64 бит полученного модуля записывается двоичное представление числа L=: 101000. В остальные 58 разрядов заполняются «0». Далее полученный блок разделяется на 16 32-разрядных подблоков, которые представлены в таблице ниже

|  |  |
| --- | --- |
| Блок | Последовательность битов |
| 1 | 01010011010000010101001101001000 |
| 2 | 01000001100000000000000000000000 |
| 3 | 00000000000000000000000000000000 |
| 4 | 00000000000000000000000000000000 |
| 5 | 00000000000000000000000000000000 |
| 6 | 00000000000000000000000000000000 |
| 7 | 00000000000000000000000000000000 |
| 8 | 00000000000000000000000000000000 |
| 9 | 00000000000000000000000000000000 |
| 10 | 00000000000000000000000000000000 |
| 11 | 00000000000000000000000000000000 |
| 12 | 00000000000000000000000000000000 |
| 13 | 00000000000000000000000000000000 |
| 14 | 00000000000000000000000000000000 |
| 15 | 00000000000000000000000000000000 |
| 16 | 00000000000000000000000000101000 |

б) Исходное сообщение M = SOIKEL\_SASHA\_VLADIMIROVITCH или в кодах ASCII – это будет соответствующие двоичные (01010011 01001111 01001001 01001011 01000101 01001100 01011111 01010011 01000001 01010011 01001000 01000001 01011111 01010110 01001100 01000001 01000100 01001001 01001101 01001001 01010010 01001111 01010110 01001001 01010100 01000011 01001000). Длина L хешируемого сообщения составила 216 битов. Эту длину необходимо расширить до 448 бит. Для этого необходимо добавить один «1» и 231 «0». Далее в последнюю часть из 216 бит полученного модуля записывается двоичное представление числа L=200: 11001000. В остальные 56 разрядов заполняются «0». Далее полученный блок разделяется на 16 32-разрядных подблоков, которые представлены в таблице ниже.

|  |  |
| --- | --- |
| Блок | Последовательность битов |
| 1 | 01010011 01001111 01001001 01001011 |
| 2 | 01000101 01001100 01011111 01010011 |
| 3 | 01000001 01010011 01001000 01000001 |
| 4 | 01011111 01010110 01001100 01000001 |
| 5 | 01000100 01001001 01001101 01001001 |
| 6 | 01010010 01001111 01010110 01001001 |
| 7 | 01010100 01000011 01001000 10000000 |
| 8 | 00000000 00000000 00000000 00000000 |
| 9 | 00000000 00000000 00000000 00000000 |
| 10 | 00000000 00000000 00000000 00000000 |
| 11 | 00000000 00000000 00000000 00000000 |
| 12 | 00000000 00000000 00000000 00000000 |
| 13 | 00000000 00000000 00000000 00000000 |
| 14 | 00000000 00000000 00000000 00000000 |
| 15 | 00000000 00000000 00000000 00000000 |
| 16 | 00000000 00000000 00000000 11011000‬ |