Cryptology exercises

Martino Tommasini

November 4, 2020

1 Attack on textbook RSA

Description

Security proofs in crypto are usually allowing the attacker access to a decryption oracle, i.e. an algorithm that returns the decryption of any valid ciphertext. In the schoolbook version of RSA, any ciphertext is valid. The attacker wins if he finds the plaintext m belonging to ciphertext c without ever asking the oracle for a decryption of c itself; any $c' \equiv c$ mod n is fair game. Show how the attacker can recover m from $c \equiv m^e$ mod n with one oracle query and some (easy) computation.This exercise shows you that schoolbook RSA should not be used in practice

Solution

The attack described below is only applicable to textbook RSA, where no padding schemes are used.
The attacker knows the encrypted message c and, instead of sending the c itself to the decription oracle, he computes a new encrypted message $c' = c * s^e$ where s is a plaintext of his choice.
Therefore we have that:

$$c' \equiv c * s^e \equiv m^e * s^e \pmod{n}$$

The encrypted message is sent to the oracle that, unaware of the attack, return its decryption:

$$(c')^d \equiv (m^e * s^e)^d \equiv m^{e*d} * s^{e*d} \equiv m * s \pmod{n}$$

since we know that $x^{e*d} \equiv x^1 * x^{k\varphi(n)} \equiv x$ mod n because $x^{k\varphi(n)} \equiv 1$ mod n by Fermat's little theorem. This holds only if m and s are coprime to n. In case they are not coprime, the same property can be proved using the CRT on $x^{k\varphi(n)}$ mod p and mod q.

Since the attacker knows the value of s, he can use it to divide the result by s mod n and find the plaintext m, belonging to the chipertext c.

2 Factor n using Pollard-rho with Floyd's cycle finding method

Description

Use Pollard's rho method for factorisation to find a factor of 27887. Use starting point $x_0 = 17$, iteration function $x_{i+1} = x_{2i} + 1$ and Floyd's cycle finding method,

Solution

The implementation of the attack is stored in the *cryptology/scripts* directory.
By following Floyd's cycle finding method, we calculated a slow-walk (x_i, x_{i+1}, $x_{i+2}, ...$), a fast-walk ($x_{2i}, x_{2i+2}, x_{2i+4}, ...$), and *gcd* of n and the difference of two walks. The iteration function is $x_{i+1} \equiv x_i^2 + 1 (\text{mod } n)$. n is 27887, starting point is $x_0 = 17$. In the table below, i is the number of iterations. It starts at 0. x_i is the value of the slow-walk in each iteration and x_{2i} is the fast-walk one. The *gcd* value is computed and showed in each iteration as well. From the

i	0	1	2	3	4
x_i	17	290	440	26279	20061
x_{2i}	17	440	20061	7866	23221
$gcd(x_{2i} - x_i, n)$	27887	1	1	1	79

Table 1: Steps of Floyd's cycle finding methods

table, it is clear that at iteration 4, we get a non-trivial *gcd*, 79, which is indeed a factor of 27887 ($= 79 \times 353$).

Why Pollard-rho works

We compute the x modulus n and we hope that:
$x_j \equiv x_i$ mod p
$x_j \not\equiv x_i$ mod n.
If this happens, we have that $x_j - x_i \equiv 0$ mod p but not mod n. This means that $x_j - x_i$ is a multiple of p but not a multiple of n. Therefore we can find the factor p as $p = gcd(x_j - x_i, n)$. Pollard rho suggested doing a pseudo random walk: choosing a starting point and using the iterative function to progressively find the x. While the x are computed modulo n, we can look at the walk in modulo p. In mod p there are just p elements and then the circle repeats. The shape the walk resemble is a ρ, where the start of the circle represents the start of the modulo p walk. So, instead of computing the gcd for every old x_i and new x_j, we can use a fast walk $x_{2i}, x_{2i+2}...$ and a slow walk x_i, x_{2i}, x_{3i} and find

the factor p when the $gcd(x_{fast} - x_{slow}, n)$ returns a non trivial factor of n. This works because the 2 walks could eventually meet in the modulo p walk.

3 Factor n using Pollard's p -1 method

Description

Factor 27887 with exponent s = lcm{1,2,3,4,5...,11} = 27720

Solution

Normally a smoothness bound B would be chosen to define an exponent s composed of smaller prime factors. Then a random base a would be raised to this exponent s and if the $GCD(a^s \text{mod n} - 1, n)$ is not 1 or n we found a factor. Else repeat with a different random a or a new smoothness bound B, larger if $g = 1$ and smaller if $g = n$.
In this case $n = 27887$ and exponent $s = lcm(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11) = 27720$

Take a random $a = 2$.
Compute: $r = GCD(2^{27720} \text{mod } 27887 - 1, 27887) = GCD(9531, 27887)$

The result is 353 which is indeed a factor for $27887 (= 353 \times 79)$.

4 Factor n using Dixon's method

Description

Factor $n = 403$ using Dixon's method

Solution

Let $n = 403$.
We try to find a and b such that:

$$a^2 \equiv b^2 \text{ mod n}$$

We pick a random $a_1 = 22$ and we compute b_1:

$$22^2 \equiv 81 \text{ mod } 403$$

The right member can be factored as follows:

$$22^2 \equiv 9^2 \text{ mod } 403$$

We found a and b. We can now compute the factors of n.

$$gcd(a-b,n) = gcd(22-9,403) = gcd(13,403) = 13$$

Therefore, 403 is factored as $403 = 13 \times 31$.

5 Attack RSA stereotyped messages using Coppermith method

Description

You learn that I sent ciphertext c = 221742016667880335235086977604419933217657946219108301 to a user with RSA public key (e, n) = (3,529774210762246675161318616746995617835565246251635147) and that this was the result of a form which sends a stereotyped message *"myfavoritenumberis "* in base 35, where the empty spaces indicate 6 unknown characters. Use LLL to recover those 6 characters.

Solution

We use Coppersmith's attack to decrypt a stereotyped message.
We can write the polynomial $f(x) = (x + a)^3 - c$ where 3 is the exponent e, c is the encrypted message, a is the stereotyped part of the message and x is the unknown part of the message we want to find.
We know that the text x is 6 characters and base 35 has been used. We use Sage to calculate the value a and the coefficient X, computed to have more balance in the matrix we are going to construct. It represents the upper bound on the x (we take the letter y because we are working in base 35 and we want the upper bound):

$a = Integer(\text{“}myfavoritenumberis000000\text{”}, base = 35)$
$X = Integer(\text{“}yyyyyy\text{”}, base = 35)$

Now we construct the lattice of polynomials such that each row is $\equiv 0$ mod n:

$$M = \begin{bmatrix} X^3 & 3X^2a & 3Xa^2 & a^3 - c \\ 0 & X^2n & 0 & 0 \\ 0 & 0 & Xn & 0 \\ 0 & 0 & 0 & n \end{bmatrix}$$

The first row is $f(xX) = (xX + a)^3 - c \equiv 0$ mod n. The others row are multiple of n. Therefore, each row is $\equiv 0$ mod n.
We build the matrix on Sage and we compute the linear combinations of the vectors in the lattice using LLL, to find a shorter vector:

$M = matrix([[X^3, 3{*}X^2{*}a, 3{*}X{*}a^2, a^3{-}c], [0, X^2n, 0, 0], [0, 0, Xn, 0], [0, 0, 0, n]])$
$B = M.LLL()$

We can obtain the message x, finding the roots of the polynomial given by one of the three rows of the reduced matrix B. Each cell is divided by the respective scaling factor.
We try with the first row:

$$Q = B[0][0] * x^3/X^3 + B[0][1] * x^2/X^2 + B[0][2] * x/X + B[0][3]$$

It's not possible to find the factor in the first row since Q is a constant ($B[0][0] = B[0][1] = B[0][2] = 0$). We try with the second row:

$$Q = B[1][0] * x^3/X^3 + B[1][1] * x^2/X^2 + B[1][2] * x/X + B[1][3]$$
$$Q.roots(ring = ZZ)[0][0].str(base = 35)$$

The factor has been found and the unknown part of the message is "23or42".

We can verify the solution confronting c with the RSA encryption of the message m found. Using Sage:

$$m = Integer(\text{“}myfavoritenumberis23or42\text{”}, base = 35)$$
$$c == (m^3 \ \% \ n)$$

The result returns true. Solution confirmed.

6 Break El-gamal signature

Description

You find two signatures made by Alice. You know that she is using the ElGamal signature scheme over F_{2027} and that the order of g is 1013, which is prime. The signatures are for $h(m_1) = 345$ and $h(m_2) = 567$ and are given by $(r_1, s_1) = (365, 448)$ and $(r_2, s_2) = (365, 969)$. Compute (a candidate for) Alice's long-term secret a based on these signatures, i.e. break the system.

Solution

By signing the message, s_1 and s_2 were computed as follows:

$$s_i \equiv ((h(m_i) - a * r_i)k_i^{-1}) \mod l$$

since $l|(p-1)$.
for $i = 1, 2$, we can rearrange the above equation to represent a by s_1 and s_2 as follows:

$$a \equiv \frac{h(m_1) - s_1 k_1}{r_1} \mod l, \quad a \equiv \frac{h(m_2) - s_2 k_2}{r_2} \mod l \rightarrow$$

$$\frac{h(m_1) - s_1 k_1}{r_1} \equiv \frac{h(m_2) - s_2 k_2}{r_2} \bmod l$$

Since $r_1 \equiv r_2 \bmod p$, we have that $g^{k_1} \equiv g^{k_2} \bmod l$. This implies that k does not change and $k_1 \equiv k_2 \equiv k \bmod l$. Therefore, we can simplify the equation and get

$$h(m_1) - s_1 k \equiv h(m_2) - s_2 k \bmod l$$
$$h(m_1) - h(m_2) \equiv (s_1 - s_2) k \bmod l$$
$$k \equiv \frac{h(m_1) - h(m_2)}{s_1 - s_2} \bmod l$$

Plugging in the numbers, we get

$$k \equiv \frac{345 - 567}{448 - 969} \equiv 679 \bmod 1013$$

Then a can be computed as

$$a \equiv \frac{h(m_1) - s_1 k_1}{r_1} \equiv \frac{345 - 448 \times 679}{365} \equiv 39 \bmod 1013$$

Using $a \equiv 39 \bmod 1013, k \equiv 679 \bmod 1013$ to verify s_1 and s_2:

$$s_1 \equiv ((h(m_1) - a * r)k^{-1}) \equiv (345 - 39 \times 365) \times 679^{-1} \equiv 448 \mod 1013$$

$$s_2 \equiv ((h(m_2) - a * r)k^{-1}) \equiv (567 - 39 \times 365) \times 679^{-1} \equiv 969 \mod 1013$$

The results are indeed correct.

7 Break DLP using Pohlih-Hellman

Description

$13 \in F_{1321}^*$ generates a group of order $1320 = 233511$. Solve the discrete logarithm problem when $g = 13$, $h = 320$ by using the Pohlig-Hellman attack.

Solution

The implementation of the attack is stored in the *cryptology/scripts* directory.
Here follows a detailed explanation

Let $p = 1321$, the generator $g \equiv 13 \bmod 1321$. It generates a group of order $1320 = 2^3 * 3 * 5 * 11$. We want to solve the DLP of $h \equiv 320 \bmod 1321$.
Using Pohlig-Hellman attack we can break the DLP in the full group of order 1320 by breaking the DLP in subgoups of prime order p_1,where $p_i|1320$.

We consider the **subgroup of order 11**.
• We compute $h^{\frac{p-1}{11}} \equiv h^{120} \equiv 938 \bmod 1321$
• We compare the above value to $g^{\frac{i(p-1)}{11}} \bmod 1321$, where $i = 0, 1, 2, ..., 10$.
To reduce the computational costs, we can compute g_i by multiplication instead of exponentiation, namely $g_{i+1} = g_i * g$ where $g_i = g^{\frac{i(p-1)}{11}}, g = g^{\frac{(p-1)}{11}}$:

$$g_{11} = g^{\frac{(p-1)}{11}} \equiv g^{120} \equiv 925 \bmod 1321$$

$$g_{11}^2 = g_{11} * g_{11} \equiv 938 \bmod 1321$$

We found:

$$h^{\frac{p-1}{11}} \equiv g^{\frac{2(p-1)}{11}} \bmod 1321$$

$$g^{\frac{a(p-1)}{11}} \equiv g^{\frac{2(p-1)}{11}} \bmod 1321$$

Therefore $a \equiv 2 \bmod 11$. $(a_{11} = Mod(2, 11))$

We consider the **subgroup of order 5**. Following the same steps above:
• $h^{\frac{p-1}{5}} \equiv h^{264} \equiv 735 \bmod 1321$
• $g_5 = g^{\frac{(p-1)}{5}} \equiv g^{264} \equiv 133 \bmod 1321$
$g_5^2 = g_5 * g_5 \equiv 516 \bmod 1321$
$g_5^3 = g_5^2 * g_5 \equiv 1257 \bmod 1321$
$g_5^4 = g_5^3 * g_5 \equiv 735 \bmod 1321$
(Note that g_5^2 and g_5^3 in the right-hand sides don't have to be computed since we already know them from the previous steps).
We found:

$$h^{\frac{p-1}{5}} \equiv g^{\frac{4(p-1)}{5}} \bmod 1321$$

Therefore $a \equiv 4 \bmod 5$. $(a_5 = Mod(4, 5))$

We consider the **subgroup of order 3**.
• $h^{\frac{p-1}{3}} \equiv h^{440} \equiv 297 \bmod 1321$
• $g_3 = g^{\frac{(p-1)}{3}} \equiv g^{440} \equiv 297 \bmod 1321$
We found:

$$h^{\frac{p-1}{3}} \equiv g^{\frac{(p-1)}{3}} \bmod 1321$$

Therefore $a \equiv 1 \bmod 3$. $(a_3 = Mod(1, 3))$

We can efficiently solve the DLP in subgroup of order $8 = 2^3$ by solving the DLP in subgroup of order 2. In this way, we can find a one bit at a time, computing the exponentiations of g just once ($g^{\frac{p-1}{2}} \equiv 1320 \equiv -1 \bmod 1321$).
We consider the **subgroup of order 2**.
• $h^{\frac{p-1}{2}} \equiv h^{660} \equiv 1 \bmod 1321$, means a is even $\rightarrow a_0 = 0$
Update the h': $h' = \frac{h}{g^0}$
• $(h')^{\frac{p-1}{4}} \equiv (h')^{330} \equiv 1320 \bmod 1321$, means that a_1 is odd $\rightarrow a_1 = 1$
Update the h': $h' = \frac{h'}{g^{1\times 2}}$

• $(h')^{\frac{p-1}{8}} \equiv (h')^{165} \equiv 1 \bmod 1321$, means a_2 is even $\rightarrow a_2 = 0$
We can write a as follows:

$$a = a_0 + a_1 * 2 + a_2 * 2^2 = 0 + 1 * 2 + 0 * 2^2 = 2$$

Therefore $a \equiv 2 \bmod 8$. $(a_2 = Mod(2, 8))$

Since the modulos 8,3,5,11 are pairwise coprime, we can compute a using the Chinese Remainder Theorem.
(Pari-GP commands $chinese(a_2, a_3); chinese(\%, a_5); chinese(\%, a_{11}))$

$$a \equiv 1234 \bmod 1320$$

We can verify the result by computing

$$g^a - h \equiv 13^{1234} - 320 \equiv 0 \bmod 1321$$

The result is indeed correct.

8 Break DLP using Pollard-rho method

Description

Use the schoolbook version of Pollard's rho method to attack the discrete logarithm problem given by $g = 3$, $h = 245$ in F^*_{1013}

Solution

The implementation of the attack is stored in the *cryptology/scripts* directory.
Here follows a detailed explanation

```
                    xf    bf      cf       xs     bs    cs
Interations: 1,  F: 27,   3,      0,    S: 9,    2,    0
Interations: 2,  F: 243,  5,      0,    S: 27,   3,    0
Interations: 3,  F: 161,  7,      0,    S: 81,   4,    0
Interations: 4,  F: 666,  28,     0,    S: 243,  5,    0
Interations: 5,  F: 231,  29,     1,    S: 729,  6,    0
Interations: 6,  F: 53,   31,     1,    S: 161,  7,    0
Interations: 7,  F: 323,  63,     2,    S: 596,  14,   0
Interations: 8,  F: 589,  126,    5,    S: 666,  28,   0
Interations: 9,  F: 364,  127,    6,    S: 985,  29,   0
Interations: 10, F: 108,  128,    7,    S: 231,  29,   1
Interations: 11, F: 972,  130,    7,    S: 693,  30,   1
```

```
Interations: 12, F: 947, 262,   14,   S: 53,   31,  1
Interations: 13, F: 531, 524,   29,   S: 783,  62,  2
Interations: 14, F: 280, 525,   30,   S: 323,  63,  2
Interations: 15, F: 161, 526,   31,   S: 1003, 126, 4
Interations: 16, F: 666, 2104,  124,  S: 589,  126, 5
Interations: 17, F: 231, 2105,  125,  S: 459,  126, 6
Interations: 18, F: 53,  2107,  125,  S: 364,  127, 6
Interations: 19, F: 323, 4215,  250,  S: 36,   127, 7
Interations: 20, F: 589, 8430,  501,  S: 108,  128, 7
Interations: 21, F: 364, 8431,  502,  S: 324,  129, 7
Interations: 22, F: 108, 8432,  503,  S: 972,  130, 7
Interations: 23, F: 972, 8434,  503,  S: 890,  131, 7
Interations: 24, F: 947, 16870, 1006, S: 947,  262, 14
```

We found $x_s = x_f$ and we have $b_s = 262$, $c_s = 14$, $b_f = 16870$, $c_f = 1006$.
Since $gcd(c_s - c_f, p-1) \neq 1$, then the congruence is not invertible.
Given that $gcd(c_s - c_f, \frac{p-1}{4}) = gcd(14 - 1006, 253) = 1$ We can proceed as follows:

$$a \equiv \frac{b_f - b_s}{c_s - c_f} \bmod \frac{p-1}{4}$$

$$a \equiv \frac{16870 - 262}{14 - 1006} \equiv 122 \bmod 253$$

• Check if 122 is the solution of the DLP:
$g^a = g^{122} = 895 \neq h \rightarrow$ Not the solution
• Check if 122+253=375 is the solution of the DLP:
$g^a = g^{375} = 245 = h \rightarrow$ Solution found.
Therefore $a = 375$

9 Break DLP using Index Calculus

Description

Use factor base $F = 2, 3, 5, 7, 11, 13$ to solve the DLP $h = 281$,$g = 2$, in F^*_{1019}.

Solution

Since $h = 281$ is a prime, it is not smooth in the factor base. We need to find a hg^i (for some i) which is smooth. Problem to solve:

$$2^a \equiv 281 \bmod 1019$$

Try random powers of a to fill matrix with linearly independent relations. After a while we started sweeping the matrix to generate the log_2 mod 1019 of all the numbers in the factor base 2,3,5,7,11,13. When sweeping we only subtracted and did every calculation mod 1018.

	2	3	5	7	11	13	
$2^1 \equiv 2 \bmod 1019$	1	0	0	0	0	0	1
$2^{10} \equiv 5 \bmod 1019$	0	0	1	0	0	0	10
$2^{27} \equiv 143 \equiv 11 \cdot 13 \bmod 1019$	0	0	0	0	1	1	27
$2^{71} \equiv 343 \equiv 7^3 \bmod 1019$	0	0	0	3	0	0	71
$2^{101} \equiv 77 \equiv 7 \cdot 11 \bmod 1019$	0	0	0	1	1	0	101
$2^{109} \equiv 351 \equiv 3^3 \cdot 13 \bmod 1019$	0	3	0	0	0	1	109

Solve the matrix we have

	2	3	5	7	11	13	
$2^1 \equiv 2 \bmod 1019$	1	0	0	0	0	0	1
$2^{10} \equiv 5 \bmod 1019$	0	0	1	0	0	0	10
$2^{27} \equiv 143 \bmod 1019$	0	0	0	0	0	1	$-151/3 \equiv 289 \bmod 1018$
$2^{71} \equiv 343 \bmod 1019$	0	0	0	1	0	0	$71/3 \equiv 363 \bmod 1018$
$2^{101} \equiv 77 \bmod 1019$	0	0	0	0	1	0	$232/3 \equiv 756 \bmod 1018$
$2^{109} \equiv 351 \bmod 1019$	0	1	0	0	0	0	$478/9 \equiv 958 \bmod 1018$

The results are indeed the same.

Finally to find our hidden power a we multiplied h by g^i for $i = 4$. This results in a factorisation within the factor base. So we can construct a as follows.

$$2^4 \cdot 281 \equiv 420 \equiv 2^2 \cdot 3 \cdot 5 \cdot 7 \bmod 1019$$

Use the corresponding logs of the factors from the table and the factors of 420 and add them all up. Then subtract i. And we will end up with our variable a.

$$(2 + 958 + 10 + 363 - 4) \equiv 311 \bmod 1018$$

And we can check that indeed:

$$2^{311} \equiv 281 \bmod 1019$$

Note this would work for any i, as long as the resulting number hg^i is smooth in our factor base. For instance, $i = 2$, $hg^2 \equiv 105 \equiv 3 \cdot 5 \cdot 7 \bmod 1019$.