Different Attacks on RSA system

Muzakkir Qadri Mohammed

December 2021

Abstract

Cryptography is a study of techniques and algorithms to secure communication in the presence of an adversary. RSA is a public-key cryptosystem that is widely used secure transmission of data. Since its initial publication in 1977, many researchers have tried to look for vulnerabilities in the system. Some clever attacks have been found. This paper illustrates in brief the most widely known attacks on RSA cryptosystem. However, none of the known attacks are devastating and the RSA system is still considered secure.

1 Introduction

RSA cryptosystem was invented by Ron Rivest, Adi Shamir, and Len Adleman. It was published in 1977 and focused mainly on providing a secure algorithm for digital authenticity and encryption. The RSA algorithm is used in applications where data privacy is a concern and hence is widely used in electronic credit card payment systems.

After the first implementation, the RSA cryptosystem has been analyzed for its susceptibility to break apart. This paper encompasses a number of interesting attacks performed on various aspects of the RSA system[1]. Hence it is not a trivial task to maintain the integrity and security of the RSA system if it is not initialized properly. Conventionally, we have used the names(Alice and Bob) for the two parties trying to perform communication with each other. Let Marvin be the name of the adversary trying to perform malicious attack on the RSA system to hinder the communication between Alice and Bob. Some common aspects of the RSA System have been noted below:

1. N is the product of two large prime numbers p and q, $N = pq$
2. $\phi(N) = (p-1)(q-1)$
3. Let e and d be two integers satisfying $ed = 1 \mod \phi(N)$, where e is the encryption exponent and d is the decryption. exponent.
4. (N, e) is the public key for encryption.
5. (N, d) is the private key used for decryption.
6. Let M be the message or the plaintext to the communicated then:
   - Encryption: $C = M^e \mod N$, where C is the ciphertext.
   - Decryption: $M = C^d \mod N$, where M is the plaintext .

The decryption works as $ed = 1 \mod \phi(N)$, which implies that : $C^d = M^{ed} = M \mod N$. Hence if for the ciphertext C, the decryption exponent d is known then the RSA system can be broken and d is therefore called as a *trapdoor* that can invert the function $M^e \mod N$. The RSA function $C = M^e \mod N$ is an example of a trapdoor one-way function. It can be easily computed, but cannot be efficiently inverted without the trapdoor d except in special circumstances. Trapdoor one-way functions can be used for digital signatures.

Digital signatures are an important application of RSA. Digital signatures provide authenticity to electronic legal documents. They are used for signing digital checks or electronic purchase orders. To sign a message M using RSA, Alice applies her private key (N, d) to M and obtains a signature $S = M^d \mod N$. Given (M, S), It can be verified that S has Alice's signature on M by checking that $S^e = M \mod N$. Since only Alice can generate S, one may suspect that an adversary cannot forge Alice's signature. Unfortunately, things are not so simple; extra measures are needed for proper security.

2 Elementary Attacks

Elementary attacks refer to the attacks caused by the misuse of RSA algorithm. In practice, many elementary attacks can be accomplished by segregating the steps in the algorithm. Two such attacks have been mentioned in the paper as described below.

2.1 Common Modulus

Suppose that there are multiple users apart from Alice and Bob. In such a case, Alice must be provided with different $N = pq$ values by each individual user for encryption which may be tedious work. To avoid this, for every user same modulus N is used but with different values of encryption and decryption exponents (e and d). Hence the public key of users would be (N, e_i) and private keys would be (N, d_i) for each user.

At first glance this may seem to work but that is not actually the case. Marvin can pose as a user with his own private and public keys. Since N is common for all users, all Marvin has to do is compute the factorization of N using his encryption exponents e_m and decryption exponents d_m. Multiple algorithms to factorize N given (e,d) can be utilized. One such algorithm called Fast Factorization for computing factors of N is given below:

1. Set $k = de - 1$.
2. Let g be some random number between $[2, \ldots, N-1]$ and set $t = k$.
3. If t is divisible by 2, set $t = t/2$ and $x = g^t \mod N$. Otherwise go to step 2.
4. If $x > 1$ and $y = \gcd(x-1, N) > 1$ then set $p = y$ and $q = N/y$, output (p,q) and terminate the algorithm. Otherwise go to step 3.

Using Fast Factorization algorithm Marvin would factorize N using (e_m, d_m) and use the factorization of N (i.e., p and q) to compute $\phi(N)$. He then uses $\phi(N)$ along with the public key of Alice e_a to compute the private key of Alice d_a using the basic principle of RSA, $ed = 1 \mod \phi(N)$.

$$d_a = e_a^{-1} \mod (\phi(N))$$

For example, let (N, e_m, d_m) be $(25777, 3, 16971)$ where $N = 25777$ is the common modulus. Let the Public key(N,e_a) of Alice be $(25777, 5)$. Using Fast Factorization algorithm, the factors of $N = 25777$ are computed by Marvin to be $p = 173$ and $q = 149$. He then computes $\phi(N) = (p-1)(q-1)$ which gives $\phi(N) = 25456$. Using $(25456, 5)$ to compute the private key of Alice d_a:

$$d_a = 5^{-1} \mod 25456$$
$$d_a = 20365$$

Hence Marvin can effectively compute the private key(N, d_a) of Alice or for that fact any other user(N, d_i) in the environment. To avoid this exploitation of RSA, it is advised that in case of multi-user environment, common modulus should not be used.

2.2 Blinding

Blinding refers to the addition of a random entity r to the message M to retrieve the Signature of a User. For example, suppose Marvin wants Bob's Signature $S = C^d \mod N$ on the cipher-message C, but Bob being no fool, refuses to sign the ciphertext C.

Marvin generates a random entity r and encrypts it using the public key of Bob $r^e \mod N$ and adds it to the original Ciphertext C to get $\hat{C} = r^e C \mod N$. Marvin then gives $\hat{C}$ to Bob for adding his signature. Bob adds his signature thinking that the message is legit, therefore $\hat{C}^d = r^{ed} C^d \mod N$. All Marvin has to do now to get Bob's Signature is to remove the entity r as follows $S = \hat{C}^d / r \mod N$.

For Example, let $p = 17$ and $q = 23$. Hence $N = 391$ and $\phi(N) = 352$. Let the encryption exponent $e = 5$. Using $d = e^{-1} \mod \phi(N)$, we get $d = 141$. So the public key is $(391, 5)$ and the private key is $(391, 141)$. Let the message C be 89. Marvin sends $C = 89$ to Bob for his signature, but Bob refuses. Marvin then takes some random value for r (say $r = 48$) and computes $\hat{C}$ as $\hat{C} = r^e C \mod N$. Hence :

$$\hat{C} = 48^5 \cdot 89 \mod 391$$
$$\hat{C} = 65$$

Marvin gives this $\hat{C}$ to Bob for his signature. Bob not suspecting anything applies his signature as $S = \hat{C}^d \mod N$. That gives us the following:

$$\hat{C}^d = 65^{141} \mod 391$$
$$\hat{C}^d = 56$$

Marvin then finally computes the modular inverse of r to get Bob's signature. Modular inverse of r is $r^-1 \mod N$=$48^{-1} \mod 391 = 334$. Therefore Bob's Signature is:

$$S = \hat{C}^d \cdot r^{-1} \mod N$$
$$S = (56) \cdot (334) \mod 391$$
$$S = 327$$

Although Blinding has been presented as an attack on RSA, it is actually one of the useful properties of RSA which allows for anonymous digital transaction where the identity of the purchaser is undisclosed.

3 Low Private Exponent

In order to decrease the decryption time or signature generation time, it may seem desirable to select a small decryption exponent d. However research by M. Weiner[2] shows that a small d can be computed from the public information (n,e) and therefore break RSA system if certain conditions are satisfied.

Weiner's Attack uses continuous fractions, quadratic expansion to guess the value for decryption exponent d. The attack works as following: the equation $ed = k\phi(N) = 1$ can be rewritten as $ed - k\phi(N) = 1$. Dividing on both sides with $d\phi(N)$ gives $\frac{e}{\phi(N)} - \frac{k}{d} = \frac{1}{d\phi(N)}$. But since $d\phi(N)$ is very large, it is discarded leading to the following approximation:

$$\frac{e}{\phi(N)} \approx \frac{k}{d}$$

Since $\phi(N)$ is approximately equal to N (because p and q are relatively large prime numbers) we get the approximation:

$$\frac{e}{N} \approx \frac{k}{d}$$

The public key being available to anyone, one would be able to use continuous fractions to determine the value of $\frac{k}{d}$ from the convergents of $\frac{e}{N}$. The correct value of encryption exponent d would satisfy the following conditions:

1. d must be an odd number if e is odd (from $ed = 1 \mod \phi(N)$ and $\phi(N)$ is an even number)
2. $\phi(N) = \frac{ed-1}{k}$ should be a whole number(since $\phi(N)$ is whole number)

Once the value of d is obtained, one can use the following quadratic equation to find the factorization of N to get the values of p and q:

$$(x-p)(x-q) = 0$$
$$x^2 - (p+q)x + pq = 0$$

It can be seen that $\phi(N) = (p-1)(q-1)$ can be expanded into $\phi(N) = pq - (p+q) + 1$ and hence rearranged to get $(p+q) = pq - \phi(N) + 1$. We know that $N = pq$. Substituting both in the above equation gives us:

$$x^2 - (N - \phi(N) + 1)x + N = 0$$

For better understanding, let the Public key be $(N, e) = (64741, 42667)$. Then $\frac{e}{N} \approx \frac{k}{d}$ implies that:

$$\frac{e}{N} \approx \frac{42667}{64741}$$

Then the convergents of $\frac{42667}{64741}$ are computed. The first convergent is $\frac{k}{d} = \frac{0}{1}$. Hence $d = 1$, which is not possible. The second convergent is $\frac{k}{d} = \frac{1}{1}$ which is again discarded for the same reason. The Third convergent is $\frac{k}{d} = \frac{1}{2}$ which implies that $d = 2$ but d must be odd because e is even. The fourth convergent is $\frac{k}{d} = \frac{2}{3}$. Hence $d = 3$ and $k = 2$. These values are verified using the two conditions specified above as follows:

1. $d = 3$ is an odd number
2. $\phi(N) = \frac{ed-1}{k} = \frac{(42667)(3)-1}{2} = \frac{12800}{2} = 64000$. $\phi(N)$ is a whole number.

Hence the value of decryption exponent d is 3. Substituting the values of $d, \phi(N)$ and N in the quadratic equation:

$$x^2 - (N - \phi(N) + 1)x + N = 0$$
$$x^2 - (64741 - 64000 + 1)x + 64741 = 0$$
$$x^2 - 742x + 64741 = 0$$

The roots of the above equation are $x = 101$ and $x = 641$. These roots are the values of p and q. Therefore $p = 101$ and $q = 641$. So if the decryption exponent is low enough, then Weiner's attack becomes possible.

Using Number Theory [3], it was found out that to avoid this attack, the value of d must have one fourth times the number of bits of N i.e., $d > \frac{1}{3}\sqrt[4]{N}$

4 Low Public Exponent

In an attempt to simplify the encryption process or to reduce the time of signature verification, one might be tempted to modify the RSA cryptosystem by choosing the public exponent to be some small number(say $e = 3$). So now the encryption can be done simply by computing $C = M^3 \mod N$, which can be done using two multiplications. At first this may seem to reduce complexity while keeping the security intact, but in fact it leads to a clever attack based on Coppersmith's Theorem[4].

4.1 Hastad's Broadcast Attack

This attack works when Alice wants to broadcast a message to multiple parties (say $P_1, P_2, .., P_k$). For simplicity, let us assume that $k = 3$. So the premise is that Alice wants to send the same message M to 3 recipients P_1, P_2, and P_3. Each party has their own key (N_i, e_i) but since this attack works only when the encryption exponent e is small, we suppose the smallest value of e as 3. We assume that the message M is less in size to each modulus $N_i (i = 1, 2, 3)$. Another assumption is that the $\gcd(N_i, N_j) = 1$ (where $i, j = 1, 2, 3$ and $i < j$) otherwise Marvin can compute the factors of some of the N_i's. Now Alice encrypts the message M using the encryption exponent $e = 3$ and the corresponding moduli N_i. That gives us the following three equations:

$$C_1 = M^3 \mod N_1$$
$$C_2 = M^3 \mod N_2$$
$$C_3 = M^3 \mod N_3$$

Marvin eavesdrops on the communication line and obtains C_1, C_2, and C_3 when Alice sends these ciphertexts to their respective parties. Marvin then can obtain the message M using the Chinese Remainder Theorem[5]. Hence by applying Chinese Remainder Theorem on C_1, C_2, and C_3 gives a $\hat{C}$ defined as:

$$\hat{C} = M^3 \mod N_1 N_2 N_3$$

We know that M is less than all the N_i's i.e., $M^3 < N_1 N_2 N_3$. Thus it can be concluded that $\hat{C} = M^3$ for all integers. Marvin can simply compute the cube root of $\hat{C}$ to retrieve the original message M ($M = \sqrt[3]{\hat{C}}$). This attack is hence called as a broadcast attack because Alice broadcasts the same message to different parties.

To illustrate this attack, let us take an example. Let us assume that there are 3 parties (P_1, P_2, P_3) to whom Alice wants to broadcast.

- Let $(p_1, q_1) = (59, 23)$, $(p_2, q_2) = (5, 41)$, and $(p_3, q_3) = (89, 3)$.
- Therefore $N_1 = 1357$, $N_2 = 205$, and $N_3 = 267$ are the corresponding moduli and let the encryption exponent $e = 3$. Computing the product of the three moduli:

$$N_1 \cdot N_2 \cdot N_3 = (1357) \cdot (205) \cdot (267) = 74,275,395$$

- $\phi(N_1) = 1276$, $\phi(N_2) = 160$, and $\phi(N_3) = 176$
- Let the message be $M = 53$. Computing M^3 :

$$M^3 = 53^3 = 148,877$$

- As can be seen, $M^3 < N_1 N_2 N_3$.
- Applying encryption to get ciphertexts, using the corresponding moduli and $e = 3$.

$$C_1 = 53^3 \mod 1357$$
$$C_1 = 964$$
$$C_2 = 53^3 \mod 205$$
$$C_2 = 47$$
$$C_3 = 53^3 \mod 267$$
$$C_3 = 158$$

- Applying Chinese Remainder Theorem on the above 3 ciphertexts gives us :

$$\hat{C} = 148,877$$

- But $\hat{C} = M^3$. Therefore $M = \sqrt[3]{\hat{C}} = \sqrt[3]{148877}$=53

As can be seen from the above example, when the public exponent e is sufficiently small, the message M can be extracted. Hastad's Broadcast attack[6] works well when the value of e is small and when the value of k is equal or more than the value of e ($k \leq e$). For larger values of e more ciphertexts C_i are required.

4.2 Franklin-Reiter Related Message Attack

Franklin-Reiter identified an attack when two messages which are related are encrypted. Related in this context refers to a message being a function in terms of the second message. Suppose that 2 messages M_1 and M_2 are encrypted by Alice using Bob's public key (N,e) and sent to Bob. Let $M_2 = f(M_1) \mod N$ for some linear polynomial $f = ax + b$, which signifies that M_2 is a function of M_1. C_1 and C_2 are the corresponding ciphertexts sent by Alice to Bob and are stated below:

$$C_1 = M_1^e \mod N$$
$$C_2 = (f(M_1) \mod N)^e \mod N$$

Franklin-Reiter stated that given the quantities (N, e, C_1, C_2, f), and the condition $M_2 = f(M_1) \mod N$, Marvin can compute the values of both M_1 and M_2 using the Euclidean Algorithm[7].

Marvin uses the ciphertexts C_1 and C_2 in the polynomial equations $g_1(x)$ and $g_2(x)$ as follows:

$$g_1(x) = x^e - C_1$$
$$g_2(x) = f(x)^e - C_2$$

Since from the equations of the ciphertexts C_1 and C_2, it can be observed that M_1 is a common root for both $g_1(x)$ and $g_2(x)$ which is in the linear polynomial form $x - M_1$. Marvin then uses the Euclidean Algorithm to compute $\gcd(g_1(x), g_2(x))$. If the gcd is linear i.e., the resulting polynomial is linear then M_1 is retrieved. Using the message M_1, the message M_2 is obtained. If the gcd is not linear then the attack fails.

For example, let the values of prime numbers be $p = 433$ and $q = 953$. Then $N = (433) \cdot (953) = 412649$. Let the message $M_1 = 133075$ $(M_1 < N)$. Let $M_2 = f(M_1) \mod N$ implies that $M_2 = (M_1 + r) \mod N$. Let $r = 217827$. Therefore $M_2 = M_1 + r \mod N = 133075 + 217827 \mod 412649 = 350902 (M_2 < N)$. Encrypting the messages M_1 and M_2 using the encryption exponent as $e = 3$:

$$C_1 = M_1^e \mod N$$
$$C_1 = (133075)^3 \mod 412649$$
$$C_1 = 406170 \; C_2 = (f(M_1) \mod N)^e \mod N$$
$$C_2 = (M_2)^e \mod N$$
$$C_2 = (350902)^3 \mod 412649$$
$$C_2 = 138392$$

After obtaining the ciphertexts C_1 and C_2, the equations $g_1(x)$ and $g_2(x)$ are computed by Marvin. The equations $g_1(x)$ and $g_2(x)$ are assumed to be modular ring. Hence each function is computed to $\mod N$. Let the function $f(x) = ax + b$ where $a = 1$ and $b = r$:

$$g_1(x) = x^e - C_1$$
$$g_1(x) = x^3 - 406170$$
$$g - 1(x) = x^3 + 6479 (\text{ Since } -406170 \mod 412649 = +6479 \mod 412649)$$
$$g_2(x) = f(x)^e - C_2$$
$$g_2(x) = (x + r)^3 - C_2$$
$$g_2(x) = (x + 217827)^3 - 138392$$
$$g_2 = x^3 + 240832x^2 + 57343x + 263734$$

Then Marvin computes the $\gcd(g_1(x), g_2(x))$ using the Euclidean Algorithm[7] which gives:

$$\gcd(g_1(x), g_2(x)) = x + 279574 \text{ (which is linear)}$$

Hence the value of common root for $g_1(x)$ and $g_2(x)$ is $x = -279574$ which is equivalent to $x = 133075$ (when computed with modulo N).

Therefore if related messages are send between Alice and Bob then using this method, the messages can be retrieved in quadratic time in $e \cdot \log N$. This attack works on all values of encryption exponent e but the time for computation increases as the value of e increases. Ideally the value of e must be small enough to be computed within quadratic time.

5 Conclusions

Twenty years of research on RSA has provided us with some very fascinating attacks. The attacks that this paper covered are:(a)elementary attacks which exploit the system semantics, (b)low private exponent attack which is serious enough to break RSA and (c)low public exponent attacks which although are significant in number but are relatively situational in nature. None of the attacks discovered are devastating to RSA. They merely signify how RSA should not be implemented and the loopholes to avoid. If RSA is implemented keeping these attacks into focus, then a robust and secure RSA system can be developed.

References

[1] D Boneh and G Durfee. New results on cryptanalysis of low private exponent RSA. *preprint*, pages 1–18, 1998.

[2] Michael J Wiener. Cryptanalysis of short RSA secret exponents. *IEEE Transactions on Information theory*, 36(3):553–558, 1990.

[3] Godfrey Harold Hardy, Edward Maitland Wright, et al. *An introduction to the theory of numbers*. Oxford university press, 1979.

[4] Don Coppersmith. Small solutions to polynomial equations, and low exponent RSA vulnerabilities. *Journal of cryptology*, 10(4):233–260, 1997.

[5] Dingyi Pei, Arto Salomaa, and Cunsheng Ding. *Chinese remainder theorem: applications in computing, coding, cryptography*. World Scientific, 1996.

[6] Johan Hastad. Solving simultaneous modular equations of low degree. *siam Journal on Computing*, 17(2):336–341, 1988.

[7] Th Motzkin. The Euclidean Algorithm. *Bulletin of the American Mathematical Society*, 55(12):1142–1146, 1949.