A RSA Key Security Gradating Algorithm Based on Threshold Attack Time

Wenxue Tan , Jinju Xi and Xiping Wang
Institute of Network Technology , Hunan University of Arts and Science , Changde , P.R.China
College of Computer Science and Technology , Hunan University of Arts and Science , Changde , P.R.China
College of Economy and Management, Hunan University of Arts and Science Changde , P.R.China
Email: {twxpaper ,jjxihuas , xpwang}@163.com

Abstract— **In this paper, we pioneer a key security level gradation scheme which is proved to efficient to counteract *I*terative-*E*ncrypting-*A*ttack against RSA. And we make it clear that the bug which hides after the traditional key generation algorithm is exploited by *I*terative-*E*ncrypting-*A*ttack and weakens RSA security, and that the case can be improved if the traditional key generation algorithm is modified delicately. After analyzing the preliminaries and practical steps of *I*terative-*E*ncrypting-*A*ttack in detail, we propose the concept of security grade of key-pair and depict the hierarchy of grades systematically, and bring forward an algorithm that grades security grade of key pairs. Furthermore, we introduce the concept of attack cost into the gradation prototype, then program for grading algorithm and set in motion a series of experiments for surveying the relationship between attack-cost and key-security-grade. At last, by the attained result from experimental statistics, we point it out that if key-pair is properly chosen RSA system can acquire a satisfying immunity from *I*terative-*E*ncrypting-*A*ttack.**

Index Terms— **Iterative Encrypting Attack ; Security Grade; Attack Time Cost; Threshold Time; RSA.**

I. Introduction

RSA is an asymmetric (public key) encryption scheme which used widely in security application of **E**-Commerce and **I**nternet-**B**ank. By protocol its security precondition is keeping private-key secrete. However, it is found that it is possible to restore plaintext by iteratively encrypting cipher-text a limited times. On this attack, some fundamentals, mechanism and countermeasure are discussed and explored in depth in this literature, which are unknown by most professionals.

A. Chosen Cipher-text Iterative Encrypting Attack

In the case above cipher-text is gotten easily by interception, while the encryption key is public to everyone, condition of attack isnt less than required.

For example.

$e = 215861$, $n = 283189$,$plaintext = 3250$.

a hacker iteratively encrypts message 3250 as followed.

3250^{215861} mod 283189 = 1598.

61598^{215861} mod 283189 = 11801.

11801^{215861} mod 283189 = 233121.

233121^{215861} mod 283189 = 3250.

after 4 times he restores $plaintext$ and labels 4-1=3. Similarly hacker intercepts any cipher-text which is sent from sender to receiver on insecure channel, operates it 3 times encryption and corresponding $plaintext$ can be restored ,who breaks the privacy of communication into pieces. Cipher-text **I**terative-**E**ncrypting-**A**ttack is defined by such attack method.

Such thing happens not only as to small integer, but also as to integer of which length reaches 36 bits or more. Here is an example.

$e = 330244721$, $n = 5271699073$.

After 256 times encryption you can restore plaintext.

B. Previous Work

Cipher-text **I**terative-**E**ncrypting-**A**ttack against RSA is firstly referred in [1], but further progress research has been ceased with regret since then. Crypto-analysis focus in recent years in respect of RSA is transferred to other attack methods. E.g. the attack is named **W**eak-**P**rime-**A**ttack ,which exploiting computational errors that occur during the key generation operation, power-analysis attacks, and some other attack trying factoring modulus n of RSA by a list of methods such as **L**attice **F**actoring **M**ethod [2], **P**artial **K**ey **E**xposure **F**actoring **M**ethod [2], and so on.

Recent some related research result points out that RSA still can provide security not less than good if you accept advice given against some discovered and possible effective attack.For example, adopting strong prime, avoiding the fixed public exponent, evading small prime p or q, and carefully extract key-pair on generating keys [3] and so on.

However, facts show that none of those advices can provide RSA immunity from **I**terative-**E**ncrypting-**A**ttack.

C. Further Progress

In the book titled "Number Theory for Computing" written by Song Y. Yan and published in 2002, some further progress involved has been made [5]. The relationship between determinant of "k-th residual equation" and primitive root is covered clearly, which points out the direction to counteract **I**terative-**E**ncrypting-**A**ttack.

Manuscript received October 1,2010; revised February 15,2011.
Supported by Natural Science Foundation Project of Hunan, with No. 09JJ6086. Corresponding author:wenxue.tan.
Correspondence to: Western part of Dong-ting Road, No. 170,415000, Changde, Hunan,P.R.China.

doi:10.4304/jsw.6.9.1873-1880

D. Our Contribution

In the course of survey and experiment concerned the subject, we analyze and pioneer a key security level gradation scheme which is proved to counteractive not less than required against **I**terative-**E**ncrypting-**A**ttack. In the meantime, the comprehension of Iterative Encrypting Attack and the theoretical flaws after attack algorithm itself is improved up to a level which hasn't been touched [4].Definitely, 2 items are summarized as follows.

(1). After analyzing the course of **I**terative-**E**ncrypting-**A**ttack in detail ,we pioneer the concept of security grade of key-pair ,devise an algorithm that grades security grade of key-pairs, and profile the hierarchy of grades.

(2).We introduce the concept of attack cost into security grade analysis, then program grading algorithm and practice a series of experiments for the purpose of going through the relationship between attack cost and key security grade.

(3).Experiment statistics and the analysis of them make it clear that if key is chosen properly and in reason RSA system can provide the satisfying immunity from **I**terative-**E**ncrypting-**A**ttack.

II. Some Related Preliminaries

There are many good formal definitions for public key crypto-systems, and we don't try to cover all of them here. Instead ,a public key cryptosystem is described by the following.

1.A set $\mathbf{M}$ of plaintexts (or messages), and a set $\mathbf{C}$ of cipher texts.

2. A set $\mathbf{K}_p$ of public keys, and a set $\mathbf{K}_s$ of secret keys.

3. A key generation algorithm $key-gen : \mathbf{Z} \rightarrow \mathbf{K}_p \times \mathbf{K}_s$.

4. An encryption algorithm $E : \mathbf{K}_p \times \mathbf{M} \rightarrow \mathbf{C}$.

5. A decryption algorithm $D : \mathbf{K}_s \times \mathbf{C} \rightarrow \mathbf{M}$.

The inputs to the key generation algorithm are called the security parameters. It is expected that as the security parameter increases, the resources required to break the crypto-system should increase more rapidly than the resources required to use it. Ideally, the running time of a break should be a (sub-)exponential function of n, while the running time of $key-gen$, E, and D should be some polynomial in n.

As to RSA, security parameters are extracted on the basis of number theory subject to a condition that a instance of (sub-)exponential problem is constructed while RSA key pair being generated. While attack method is improved based on some preliminaries as followed in order to counteract the restraint above.

In this section, we present the notion of a public key crypto-system and discuss some related preliminaries laying foundation for the RSA system and **I**terative-**E**ncrypting-**A**ttack.

A. Introduction of Notations

We introduce some basic notations that are used throughout the paper as Figure I, the rest notations are introduced where referred.

B. Definitions and Theorems on Primitive Root

Definition 1. q-th **root of unity**.Let n be a positive integer, and q be a integer, we call p a q-th **root of unity** if (1).

$$p^q \equiv 1 mod\, n \tag{1}$$

Definition 2.**Order** of $p\, mod\, n$. Suppose n be a positive integer, and p be a integer subject to (2) if there exists the smallest positive value of x as (3), we call x the **Order** of $p\, mod\, n$, denoted by $ord_n(p)$ or $ord(p, n)$.

$$gcd(p, n) = 1 \tag{2}$$

$$p^x \equiv 1 mod\, n \tag{3}$$

Definition 3. **Primitive root of** n. Suppose n is a positive integer, and p is an integer subjected to (2), if p satisfies (2) we define p a **Primitive root of** n .

Theorem 1. If n is a positive integer, and p is an integer for which (2) and (4) then ,
(1) if m is a positive integer for which (5),then $r \mid m$.
(2) $r \mid \phi(n)$.
(3) for any integers s and t , (6) is satisfied if and only if (7) .
(4) any two integers in an integer sequence $q, q^2, q^3, \cdots, q^r$ are not congruent to modulus .
(5) if m is a positive integer , then the order of a^m modulus n is (8).
(6) the order of a^m modulus n is r if and only if (9).

Theorem 2.If an integer n for which $n > 1$ has primitive roots if and only if $n =$ 2,or 4,or p^a,or $2p^a$ for which p is an odd prime, and a is a positive integer.

TABLE I.
Basic Notations

Notations
p : the less prime of both primes in RSA key-pair.
$\mathbf{M}$: A set of messages.
$\mathbf{C}$:A set of Encrypted messages
q : the bigger prime of both primes in RSA key-pair.
$\mathbf{K}_p$: A set of public keys
e : the exponential index for encryption in RSA key-pair.
$\mathbf{K}_s$: A set of secret keys.
d : the exponential index for decryption in RSA key-pair.
$\mathbf{Z}$: A set of positive integers.
gcd(x, y) : a function which returns great common divider.
$\phi(n)$: a function which returns Euler value.
key-gen: A key-generation algorithm . **D**: A decryption algorithm ..

Further deduction from theorem 2. If p_i, p_j where $i \neq j$ are distinct primes and there is an integer n for which $n = a^a$ and $a > 3$ or (10) for which $a \geq$ 2,or $k \geq 2$, then there doesn't exist **Primitive root** modulus

n .

$$r = ord_n(p) \tag{4}$$

$$p^m \equiv 1 \, mod \, n \tag{5}$$

$$p^s \equiv p^t \, mod \, n \tag{6}$$

$$s \equiv t \, mod \, r \tag{7}$$

$$\frac{r}{gcd(r,m)} \tag{8}$$

$$gcd(m,r) = 1 \tag{9}$$

$$n = 2^a {p_1}^{e_1} {p_2}^{e_2} \cdots {p_k}^{e_k} \tag{10}$$

Theorem 1 reveals that the order of $p \, mod \, n$ is a factor of Euler value of n.**Further deduction from theorem 2** demonstrates that if a positive integer n has multi odd-distinctive-prime factors, the primitive root to modulus doesn't exist. Both results are used in the analysis of Cipher-text **I**terative-**E**ncrypting-**A**ttack covered in the sections followed. Proof of both **theorem 1** and **theorem 2** is covered in detail in reference [2].

C. Principles of Iterative Encrypting Attack

The mathematical proof of RSA is demonstrated in [5] in detail which isn't provided here.

According to RSA algorithm, because (11).

$$gcd(e,(p-1)(q-1)) = 1 \tag{11}$$

So there always exists an integer d subject to (12).

$$ed \equiv 1 \, mod \, (p-1)(q-1) \tag{12}$$

And by Euler theorem [6],(13) is always satisfied. Which can be rewritten into (14).

$$e^{\phi((p-1)(q-1))} \equiv 1 \, mod \, (p-1)(q-1) \tag{13}$$

$$e \times e^{\phi((p-1)(q-1))-1} \equiv 1 \quad mod \, (p-1)(q-1) \tag{14}$$

So, (15) is gotten.

$$d = e^{\phi((p-1)(q-1))-1} \tag{15}$$

That is to say that d can be expressed by some form of modulus-exponent of e . By **definition 1**, denote $(p-1)(q-1)$ by n,then e is a q-th **root of unity** which satisfies (16).

$$e^{\phi(n)} \equiv 1 \, mod \, (n) \tag{16}$$

By the **further deduction from theorem 2**, because n can't be a composite which only contains unity-prime factor, it is impossible for e to be primitive root of n .In other words, there certainly exists an integer x smaller than $\phi(n)$ subject to (17), by which extract d as (18).

$$e^x \equiv 1 \, mod \, (n) \tag{17}$$

$$d = e^{x-1} \tag{18}$$

In order to counteract "$p-1$-factoring method " [7], p and q are subject to that both $p-1$ and $q-1$ should contain enough big prime factors, and a conventional process is selecting two big primes p_1 and q_1 subject to (19), p,q being another pair primes, and these primes like p named strong primes or secure primes.

$$p = 2p_1 + 1, q = 2q_1 + 1 \tag{19}$$

Because of (19),(20) can be gotten and rewritten further into (20).

$$n = 2p_1 \times 2q_1 = 2^2 \times p_1 \times q_1 \tag{20}$$

By Euler Formula, we deduce (21).

$$\phi(n) = \frac{1}{2}(p_1-1)(q_1-1) \tag{21}$$

By conclusion (2) of **theorem 1**, we demonstrate $x \mid \phi(n)$.That is to say that x is sure to be a factor of $\phi(n)$.Obviously,p_1, q_1 are prime integers, and are odd integers too, and $\phi(n)$ is sure to be expressed in (22),for which k is a positive integer,θ_1, θ_2 are 2 odd distinctive integers but unable be confirmed to be 2 odd prime integers.

$$\phi(n) = 2^k \times \theta_1 \times \theta_2 \tag{22}$$

We define S as a set of all positive factors of $\phi(n)$ which is smaller than $\phi(n)-1$ .Namely, as (23).

$$S = \{s \mid 0 < s < \phi(n)-1, s \mid \phi(n)\} \tag{23}$$

For example.

$p_1 = 53, q_1 = 83.$

θ_1, θ_2 are 2 odd distinctive prime integers 13 and 43 respectively as (24) and $s = 4$ to satisfy (25).

$$\phi(n) = 2^4 \times 13 \times 43 \tag{24}$$

$$s^4 \equiv 1 mod \, \phi(n) \tag{25}$$

Another example.

$p = 179, p_1 = 89, q = 263, q_1 = 131.$

θ_1, θ_2 are 2 odd distinctive integers 11,65 respectively, but one of both be non-prime as (26).

$$\phi(n) = 2^5 \times 11 \times 65 \tag{26}$$

Set S can be extracted by "exhaustively-listing-method". However,by further progress analysis,it is more meaning to divide S into several subsets as far as researching Cipher-text **I**terative-**E**ncrypting-**A**ttack is concerned.

III. Key Security Grading Algorithm

As analysis above, the number of positive factors of $\phi(n)$ is different by $\phi(n)$,but x is sure to be some factor of $\phi(n)$ from which d can be derived. Suppose $\theta_1 < \theta_2$,the positive factors of $\phi(n)$ denoted by x of which expression form only ranges in the 4 types as follows.

(1).Composite factors expressed as (27) , where t be a positive integer.

$$x = 2^t \tag{27}$$

(2).Composite factors expressed as (28) , where t be a positive integer and θ_1 be the unique odd factor.

$$x = 2^t \times \theta_1 \tag{28}$$

(3). Composite factors expressed as (29) , where t be a positive integer and θ_2 be the unique odd factor.

$$x = 2^t \times \theta_2 \tag{29}$$

(4). Composite factors expressed as formula (30) , where t be a positive integer and $\theta_1 \times \theta_2$ be the unique odd factor.

$$x = 2^t \times \theta_1 \times \theta_2 \tag{30}$$

According to analysis of **I**terative-**E**ncrypting-**A**ttack against RSA, there inevitably exists x for which subject to (17). The essence of issue is how big is x for which it is enough to defeat **I**terative-**E**ncrypting-**A**ttack as far as current computation power is concerned, and how to choose p, q and e in order to keep x in a rational magnitude.

In section II,we prove that it is possible to recover plaintext by operating cipher-text with $x-1$ times encrypting operations. That is to say the time cost of cracking security for adversary is $O(n)$. Whereas, x is one of factors of $\phi(n)$,where (17) is satisfied. If is conditioned by the enough length in bits or the rational magnitude when RSA key-pair (e, d) is randomly generated, adversary is defeated to recover plaintext or secret key exponential d by **I**terative-**E**ncrypting-**A**ttack.

Let be a set of all positive factors of $\phi(n)$, which can be divided into 4 subsets, where the factors of each subset are with an approximate equal length in bits. For example, the bit-length of composite factors approximately equal to $log_2\theta_2$,which are expressed as formula (29) where t be a positive integer and θ_2 be the biggest odd factor. That provides an index to security level or security grade of key-pair. We construct a scheme that grade security of key according to the approximate bit-length.

A. Hierarchy of Key-security Grades

Definition 4. **A-Grade security**.Given a key-pair (e, n) and (d, n) ,where n be a multiply of two distinctive strong big primes and e, d subject to (17) and (12), x denotes the same object as section , if the factor-multiply-expression of x is as equation (27) where t be a positive integer, we define the key-pair like so **A-Grade security** key. In other words, the key-pair is with **A-Grade security**.

In general, key with **A-Grade security** is insecure at all. Under condition of that n is a multiply of two distinctive strong big primes t usually is a small integer far less than 1024, if the key-pair like so in use, adversary is able to restore plaintext easily by siding secret key.

For example,$e = 10789, n = 17869$, x is extracted to equal to 4,which is **A-Grade security** key.

Definition 5.**D-Grade security** and **D-Grade security**.Given a key-pair(e, n) and (d, n),where n and e, r, x represent the same object as above and be subject to the same condition as above , if the factor-multiply-expression of x is as equation (28) where t be a positive integer, and θ_1 be the biggest odd factor in expression of x,but subject to $\theta_1 < \theta_2$ in factor-multiply-expression of x , we define the key-pair like so **B-Grade security** key. Namely, the key-pair like so be with **B-Grade security**. Similarly, replace parameter θ_1 by an alternative θ_2, and **C-Grade security** key is defined clearly.

Keys with **B-Grade security** or with **C-Grade security** are more secure than the key-pair with **A-Grade security**. If selecting two primes p, q subject to $\mid q \mid \approx \mid q \mid$ to generate key-pair, bit-length of θ_1 or θ_2 reaches half of log_2n [8].

For example.

$e = 96957$,$n = 45378321827173574521$.

x is extracted to equal to 1736370016256,which is **C-Grade security** key.

Definition 6.**D-Grade security**.Among prime factors of $\phi(n)$,$\theta_1 \times \theta_2$ is the biggest. If factor-multiply-expression of x is as equation (30) where t be a positive integer, we define the key-pair like so **D-Grade security** key, that is to say the key-pair like so with **D-Grade security**.

In general, key with **D-Grade security** is most secure in all keys. If n is a multiply of two distinctive strong big primes of 512 bits, n usually is with length of 1024 bits approximately. According to Euler formula (31), the length of $\phi(n)$ is the same as n nearly, this length equals to that of x by and large. If **D-Grade security** key-pair is used, it is impossible for adversary to restore plain-text by **I**terative-**E**ncrypting-**A**ttack.

For example.

$e = 649012839502789295925486568601639 49793$.

$n = 1038420543204462873545578353796907431177$.

$x = 2028165123446216549514021136121200640$.

It is **D-Grade security** key.

$$\phi(n) = (p-1)(q-1) \tag{31}$$

B. Maximum Attack Time and Threshold Attack Time

Definition 7.**Maximum Attack Time**.Given a key-pair (u, n) and (r, n) ,where its security Grade is "**A-Grade**, by the definition of **A-Grade security**, if an adversary sets in motion **I**terative-**E**ncrypting-**A**ttack he can succeed at cost of 2^k times encryption operations at the utmost, where (30) is satisfied, for which k is a positive integer, θ_1, θ_2 are 2 odd distinctive integers. We define the time cost of the 2^k times encryption operations on computation platform installed a visual CPU at speed of Giga FLOPS(**G**iga of **FLO**ating **P**oint operations per **S**econd, abbreviated by GFLOPS) is **Maximum Attack Time** abbreviated by **MAT**.

Definition 8.**Threshold Attack Time**.Given a key-pair (e, n) and (d, n),where its security Grade is **B-Grade**, by the definition of **B-Grade** security, if an adversary triggers **I**terative-**E**ncrypting-**A**ttack it is possible to succeed for him at cost of θ_1 times encryption operations at lowest, where (30) is satisfied, for which k is a positive integer,θ_1, θ_2 are 2 odd distinctive integers. We define the time cost of θ_1 times encryption operations on computation platform installed a visual CPU at speed of GFLOPS is **Threshold Attack Time** of **B-Grade** abbreviated by **TAT-B** [9].

Similarly, we define **Threshold Attack Time** of **C-Grade** which abbreviated by **TAT-C**, and **Threshold Attack Time** of **D-Grade** which abbreviated by **TAT-D**.

IV. Design Of Grading Secure Level Algorithm

On the basis of some theorems, definitions and analysis above, we pioneer an algorithm to grade the security of keys. However, the security requirement of key can be defined in many ways. In general, while defining a security goal it is important to state what resources are available to an attacker and what success criteria the attacker must fulfill. Stated in detail, a very basic requirement is that it shouldnt be possible to derive the secret key from the public key efficiently. Indeed, and it is considered the most devastating cryptanalytic break aiming for computing K_s from K_p at cost of polynomial time. There are many issues that arise in determining good notions of security, and we do not try to address them all here.

In view of other legacy secure requirement as mentioned above, the process of filtering prime is still reserved in our design and some restraint condition is left untouched. Such as the condition of strong primes and the measures against **P**artial **K**ey **E**xposure [10] and so on. Some concerned examples is to exhibited in the section followed.

A. Generation of a Strong Prime

In this section we recommend an algorithm for generation of a safe prime. It follows the single party protocol proposed by [3], where p and p_1 prime subject to (19).

1. The players generate a random number $p_0$with proper bit-length.

2. If $j = 1$ compute (32) .If $j \neq 1$ compute (33) .

$$\langle p \rangle_j^I := 2 \langle p' \rangle_j^I + 1 \tag{32}$$

$$\langle p \rangle_j^I := 2 \langle p' \rangle_j^I \tag{33}$$

3. Run the trial division on p and p_0. If either of them appears to be divisible by a small prime, go to step 1.

4. Run the Miller-Rabin test on p_0 with $\zeta = 1$ and $g = 1$. If it fails, go to step 1.

5. Run the Miller-Rabin test on p' with $\zeta = 1$ and $g = 1$. If it fails, go to step 1.

6. Run the Miller-Rabin on p' with a sufficiently large ζ and some random g, and ensure a small error probability (e.g. 2^{-64}). If it fails, go to step 1, else return a probability safe prime p'.

$$lgk \ll n, B = O(n) \tag{34}$$

As the p_0 are not random $(n-1)$ bit numbers, some care must be taken in choosing the parameter ζ in step 6.We do not address these details here. Assuming (34) is satisfied, and that safe primes are sufficiently dense , as is widely conjectured and supported by empirical evidence. The expected bit-complexity of this algorithm is as (35), where $\gamma = 128$ is a security parameter smaller than n. Assuming that one tests about $n^2/(lgn)^2$ candidates in parallel, the round-complexity is $O(n)$.

$$\begin{aligned} &O(n^3/(\lg n)^2(k^3 \lg k\gamma + k^2\gamma^2 \\ &+nk^2 \lg k + n^2k)) \end{aligned} \tag{35}$$

$$Euler = (p-1)(q-1) \tag{36}$$

$$p_1 = \lfloor (p-1)/2 \rfloor \;, q_1 = \lfloor (q-1)/2 \rfloor \tag{37}$$

$$EulerofEuler = 2 \times (p_1 - 1)(q_1 - 1) \tag{38}$$

B. Grading Security Level of Key

```
1. The users input 2 random safe primes $p, q$ with proper bit-length.
2. Compute (36).
3. Compute (37) .
4. Compute (38). 5. While($IsEven(p_1)$) $p_1 = p_1/2$
6. While($IsEven(q_1)$) $q_1 = q_1/2$.
7.If($p_1 > q_1$) $Swap(p_1, q-1)$.
8. Choose $u$ with proper bit-length at random.
9. If($GCD(u, Euler) > 1$) go to step 7.
10. $x_0 = 2$.
11. While($x_0 < EulerofEuler/(p_1 \times q_1)$ ). {
  11.1.If($EulerofEuler mod\, x_0 == 0$ ).
  11.2. If($Powermod(u, x_0, Euler) == 1$){
    11.2.1. Output (Grade A );$return(u)$.}
  11.3.$x_0 = x_0 \times 2$.}
12. $x_0 = p_1$.
13. While($x0 < EulerofEuler/p_1$). {
  13.1.If($EulerofEuler mod\, x_0 == 0$ ).
  13.2. If($Powermod(u, x_0, Euler) == 1$){
    13.2.1. Output (Grade B );$return(u)$.}
  13.3.$x_0 = x_0 \times 2$ .}
14.$x_0 = q_1$ .
15. While($x_0 < EulerofEuler/a_1$).
  15.1.If($EulerofEuler mod\, x_0 == 0$).
  15.2. If($Powermod(u, x_0, Euler) == 1${
    15.2.1 Output (Grade C );$return(u)$.}
  15.3. $x_0 = X_0 \times 2$.}
16.$x_0 = p_1 \times q_1$ .
17. While($x_0 < EulerofEuler$ ). {
  17.1 .If($EulerofEuler mod\, x_0 == 0$ ).
  17.2. If($Powermod(u, x_0, Euler) == 1$) {
    17.2.1. Output (Grade D); $return(u)$.}
18. End.
```

On purpose to counterattack **L**ow-**P**ublic-**E**xponential-**A**ttack [8], [11], the bit-length of candidates of e should be assured to attain to enough big value. When some number of e with a certain security grade is returned, and if grade of security satisfying the security requirement of application, it is easy to compute secret exponential d by (12), then a RSA key-pair is generated. Let $p \times q$ be with k bits in bit-length. The possible most loops of algorithm is as (39). So, the time-complexity of this algorithm is $O(k)$.

$$\begin{aligned} & log_2 n + log_2 \frac{n^2}{p_1 q_1} + log_2 \frac{n}{q_1 p_1} \\ & \leq 4log_2 n = 4k \end{aligned} \tag{39}$$

V. Implimentation And Experiment

We program the security level gradation algorithm using Matlab version 7.0 and Victor Shoups Number Theory Library package [12]. The experiment program operates in two phases.

At the beginning, given a pair of p, q, it extracts each candidate key and its corresponding security grade, and count the keys at uniform in security strength, so as to survey the distribution of different security grade, which would provide help for classifying candidate keys. Second, it tries to grade keys according to current security requirements and attack cost of typical attack method in order to estimate security index as viewed from nowadays computation power usually owned by an average entity.

A. Distribution of Keys with Different Grade

First, it exhaustively grades the security level of all possible key pairs which derived from a certain given prime pair, then classifying and counting key-pairs by security grade; where subjects of prime is required not more than 32 bits in length. However, exhaustively-grading-method requests to cost a lot of computation power. As to the bigger prime it is unrealistic to grade all possible keys but to grade a part of keys instead, e.g. grading 0x7ffffff (about 2^{27}) keys.

A segment of data is shown in Table II,which reveals that the key-pairs with different security grade always exist for a prime-pair and what distinguishes them is total or density. The upper is the security grade of key and the total of that is bigger. Over 90 percent of keys are **Grade D** keys, i.e. nearly all keys are substantially secure, but which is far from an excuse to put it on one side. A bit of flaw or neglect probably leads security to end in the area of information security and the disastrous lesson like which is far from of no occurrence [13].

B. Cost of Attack and Security Grade

Second, it simulates adversary to trigger **I**terative **E**ncrypting **A**ttack against key-pairs in different security grade as to estimate the cost of time and computation power.

In this phrase of experiment, most operation of attack is modulus exponential, which with a high cost of time and memory space. In order to better running speed, much implementation introduces some improving measures. E.g. taking a share fixed public key exponential e on encrypting. Saving the key parameters p and q, by **C**hinese **R**emainder **T**heorem reducing a big modulus into a small one in decryption ,but not computing the modulus exponential of a big integer directly [14].

TABLE II.
Security Grade Of Key-Pairs And Distribution

1.p:107 q:167 bits of n:15			
Grade-A keys	Grade-B keys	Grade-C keys	Grade-D keys
14	192	640	7680
2.p:263 q:347 n:91261 bits of n:17			
Grade-A keys	Grade-B keys	Grade-C keys	Grade-D keys
30	5568	19200	115200
3.p:503 q:563 n:283189 bits of n:19			
Grade-A keys	Grade-B keys	Grade-C keys	Grade-D keys
5	7133	8130	8373338
4.p:4127 q:4703 n:19409281 bits of n:25			
Grade-A keys	Grade-B keys	Grade-C keys	Grade-D keys
7	1013	53424	8334162

1∼4: grading all key-pairs, 5: grading 2^{27} key-pairs

TABLE III.
A Case of Key

p_1:334958292187850323050465726729607241393071535512172 865924288676285920727982868603395967614756760312411804 30051162397. Digits of p_1:116 Bits of p_1: 384
q_1:445995034091126057501697296859966792546417715586445 8 416366682283369378789781242834886978624743902267687942 6 89783826052378976599603140015744345592710027051 Digits of q_1:154 Bits of q_1: 511
p_1_q_1:149389734943425667810383818505846483419014335076 7 955040310048389046822528021660779504892577547734620376 093410245021868887653908992765977371696451973207722810 233733351644202919754153340648053194753222337331762739 699240067646941563272145646639445728222301912426398640 01247 Digits of p_1_q_1:270 Bits of p_1_q_1: 879
n:4066775408019241227491734510485860014976411410975085 363855578932501554954115099404385730672101947685302 09 660310561145046858551478307489833748818975517571164 63 727483903888032408913069950869965436520426968778938 62 186193037257611971250376566130105158950808808907602 62 69349345793334868031931794623523344702072986 9 Digits of n:309 Bits of n: 1026

The experiments are operated on the computing platform of a high performance cluster, named DeepComp 6800, made in Lenovo, with 24 computing nodes networked by Myrinet, each of which is provided with both processors of Intel IA-32 Pentium 4 Xenon 2000 MHz, of which overall maximum computing speed of system reaching 89.6 GFlops [15]. Statistics about a key-pair with 1026 bits in length are listed in Table III, and Table IV exhibits that **MAT-A** is teen **GFLOPS•**Seconds, that is to say grade-A keys are insecure at all, and that other 3 **TAT**s are far over 10^{100} **GFLOPS•**Years, which are strong and distinctively immune from **I**terative **E**ncrypting **A**ttack as viewed from today computation power usually owned by an average entity.

TABLE IV.
ATTACK COST OF KEY-PAIRS AND SECURITY GRADE

MAT-A	**MAT-B**	**MAT-B**	**MAT-D**
$0.12 \times 10\mathbf{S}$	$0.39{\times}10^{110}\mathbf{Y}$	$0.53{\times}10^{148}\mathbf{Y}$	$0.18{\times}10^{264}\mathbf{Y}$
Computer Workload Unit: **S**: **GFLOPS** × Second or **GFLOPS** • Second **Y**: **GFLOPS** × Year or **GFLOPS** • Year			

VI. CONCLUSION AND OUTSTANDING PROBLEMS

During the exploration and research, we pioneers a key security level gradation scheme by which those weak keys threatened by **I**terative-**E**ncrypting-**A**ttack are filtered easily, in other words, which build up the immunity of RSA system from **I**terative-**E**ncrypting-**A**ttack. After analyzing the preliminaries and practical steps of **I**terative-**E**ncrypting-**A**ttack in detail, we initiate the concept of security grade of key-pair and profile the hierarchy of grades systematically. Furthermore, we introduce the concept of attack cost into the gradation prototype, then program for grading algorithm and practice a series of experiments for the purpose of exploring the relationship between attack-cost and key-security-grade.

As to RSA security application in reality and further work about the subject,2 items are summarized as follows.

A. Proposal for key-pair Extraction

In this paper, we present key-pair security gradation scheme against **I**terative-**E**ncrypting-**A**ttack, and illustrate 4 security grades of keys and their grading algorithm. In addition, the time performance of key and the corresponding experiment data is analyzed and summarized in depth. Customarily, people tend to relate the security of system to crypto algorithm, the center of gravity of concern on security is placed on algorithm. As a matter of fact, and security grade is a primary characteristic of key. In the engineered application of a strong algorithm there are chances of generating weak keys, which puts the information security in danger of collapse. It is potential and necessary to quantitatively meter security attribute of key and to promote it in information security engineering as a memorandum.

By the conclusion drawn from this research we propose that candidate keys should be graded and should be proved with a sound security grade not lower than the point where the computation power owned by adversary is evaluated as inadequate to break as possible. As to key-pair generation of RSA, here we don't recommend keys with some imperative specific security grade. Research and experiment makes it clear that the advice is constructive to defeat **I**terative-**E**ncrypting-**A**ttack.

B. Outstanding problems Involved in the Research

In the study, we find two interesting outstanding problems.

(1). Density of key derived from two very big strong primes varies with different security grade. Does it manifest a similar distribution profile with prime less than some integer? And theory about prime distribution is illustrated in **P**rime **N**umber **T**heorem [5].

(2). Can it be confirmed that there dont exist more effective attack algorithms against a key with certain security grade than **I**terative-**E**ncrypting-**A**ttack? This will influences estimate of **MAT** and **TAT**.

ACKNOWLEDGMENT

This work is partially supported by the Provincial Natural Science Foundation Project of Hunan, with No. 09JJ6086.

The authors also gratefully acknowledge the helpful comments and suggestions from the reviewers, which have improved the presentation.

REFERENCES

[1] L. Kai-cheng, *Computer Cryptograph:data secutity and privacy of computer network*, ser. combinatorics in Computer Science. PeiKing,TsingHua University Press, 2003, vol. 7302075363, ch. The Role of Trust Management in Distributed Systems Security.

[2] A. S. Yevgeniy Dodis, Amit Sahai, "On perfect and adaptive security in exposure resilient cryptography," in *Proceedings of Eurocrypt*, Mar. 2001.

[3] V. S. J.Algesheimer, J. Camenisch, "Efficient computation modulo a shared secret with application to the generation of shared safe-prime products," in *Proceedings of crypto'2002*, May 2002.

[4] J. S. P. Nguyen, " Lattice reduction in cryptology: an update. In Algorithmic Number Theory," in *Proceedings of The 4th Algorithmic Number Theory Symposium* , 2000.

[5] William.Stallings, *Cryptography and Network Security:principles and Practices , (4th version, English)*. Publisher: Prentice Hall, 2006, vol. 3642077102, ch. Principles of Cryptography .

[6] S. Y. Yan, *Number Theory for Computing (2nd version, English)*, ser. combinatorics in Computer Science. Springer, 2002, vol. 3642077102, ch. Computational (Algorithmic) Number Theory .

[7] G. D. Boneh, "Cryptanalysis of RSA with private key d less than N 0.292," *IEEE Transactions on Information Theory*, vol. 46, no. 4, Apr. 2000.

[8] Y. Z. R. Steinfeld, "An advantage of low-exponent RSA with modulus primes sharing least significant bits," in *Proceedings of RSA Conference 2001*, Feb. 2001.

[9] K.ManuelL.Joachim, "Hardware Software Co-design of Elliptic curve Cryptography on an 8051 Mierocontroller," in *Proceedings of Cryptographic Hardware and Embedded Systems* , Oct. 2006.

[10] D. Boneh, "smooth integers using CRT decoding," in *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, May 2000.

[11] S. H. E. K. A. S. R. Canetti, Y. Dodis, "Exposure-Resilient Functions and All-or-Nothing Transforms," in *Proceedings of Eurocrypt 2000*, Aug. 2000.

[12] N. C. J. Camenisch, "A public key encryption scheme secure against key dependent chosen plaintext and adaptive chosen ciphertext attacks," in *Proceedings of Eurocrypt 2009*, Apr. 2009.

[13] C. F.R-Henriquez, "Parallel MultiPliers Based on Special Irreducible Pentanomials," *IEEE Transactions on Computers*, vol. 52, no. 12, Dec. 2004.

[14] P. Y. Y. Y. X. CHEN Huafeng, SHEN Haibin, "Characteristics of Parameterized Chaotic Map on Security and Implementation," *Chinese Journal of Electronics*, vol. 16, no. 4, Apr. 2007.

[15] C. Q. Zhao Yonghua, Chi Xuebin, "Efficient Algorithms for Matrix Eigenproblem Solver on SMP Cluster," *Journal of Computer Research and Development*, vol. 44, no. 2, Feb. 2007.

Wenxue Tan (1973-).
He graduated with Master's of Science in Information Technology and Earth Exploring from East China Institute of Technology, Jiang-xi, Mainland of China,2003.
In 2004, he joined Hunan University of Arts and Science as a lecturer,being approved and authorized as Computer Software System Analyst by Ministry of Personnel P.R.China in 2005,and being promoted to Associate Professor and Senior Engineer in 2008.
His current research interests include Computer Science and Technology, Network and Information Security.
Email to: twxpaper@163.com
Correspondence to: College of Computer Science and Technology,Hunan University of Arts and Science, Western part of Dongting Road,No.170,ZIP:415000, Changde, P.R.China.

Jinju Xi (1965-).
She graduated with Master's of Science in Computer Application and Technology from Huazhong University of Science and Technology, Hu-bei,Mainland of China,2005.
In 2005,she joined Hunan University of Arts and Science as a teacher, being promoted to Associate Professor in 2005,and being promoted to Professor in 2010.She engages in teaching in Hunan University of Art and Science.
Her research interests include: Artificial Intelligence and Pattern Recognition.

Xiping Wang (1980-).
She graduated with Bachelor's of Marketing from East China Institute of Technology, Jiang-xi,Mainland of China, 2004.
She is an Instructor of Hunan University of Arts and Science.
Her current research interests include Electronic Commerce and Information Security, Technology of Marketing by Internet.