Analysis of RSA Digital Signature Key Generation using Strong Prime

Abdelmajid Hassan Mansour*

University of Jeddah, Faculty of Computers and Information Technology, Department of Information Technology, Khulais, 21921, Jeddah, Saudi Arabia
Email: majdemam@gmail.com

Abstract

RSA digital signature is a public key algorithm, uses a private key for signing and a public key for verifying. Achieving the efficiency and acceptable level of time for generating strong keys is an important aspect and a key factor of the different security issue that facing the RSA. This paper proposes a new scheme for generating private and public key of the RSA Digital Signature using “Strong prime” concept, state that \( p = 2p_0 + 1 \), \( q = 2q_0 + 1 \), based on Gordon’s algorithm. In order to optimize the efficiency of key generation time strategy for the prime factorization that relying on such probability.

Keywords: Private & Public key; Strong prime; Gordon’s algorithm; Hash Function; Message Digest; Factorization problem.

1. Introduction

A digital signature is a public key cryptographic algorithm that is designed to protect the authenticity of a digital message or document. A message is signed by a secret key of the sender to produce a signature and the signature is verified against the message by a public key. Thus any party can verify the signatures, but only one party with the secret key can sign the messages. Digital signatures are used widely in e-commerce applications, banking applications, software distribution, and in other cases where jurisdiction is involved and it is important to detect

Permanent Address: Faculty of Computer Science & Information Technology, Alneelain University, Khartoum, 11111, Sudan.

------------------------------------------------------------------------

* Corresponding author.
forgery or tampering. They are the digital counterpart of handwritten signatures that can be transmitted over a
computer network. Only the sender can make the signature, but other people can easily recognize as belonging
to the sender. The sender produces a signature consisting of a number associating a message in digital form with
a secret key, digital signature provides three types of services such as [9]:

- **Authentication**: is a procedure to verify that received messages come from the valid source. It must verify the author and the date and time of the signature.
- **Message integrity**: it must authenticate the contents at the time of the signature and does not alter
during data transfer. If the message has been changed, then we cannot get the same signature.
- **Non-repudiation**: It means that the signer (sender) cannot claim that they did not signed the document or message.

The digital signature provides a means of integrity checking. This is done to provide assurance for the receiver
that the data was in fact sent by the assumed party. The integrity plays a critical role in virtual society and it’s
important to protect it from coming out to the public ensure data integrity so that every important data has to be
signed by owner in order to send it safely inside the network [8]. Digital signature is the most effective
technique for ensuring authentication, integrity, and non-repudiation of data in an open network such as the
Internet. Digital signature is a verification method requires the signature holder to have two keys: the private-
key for signing a message and the public-key for verification of authenticity of the message. The main goal of
the Digital signature is to verify that a message has not been modified in transit after it was signed and also, to
give the receiver of the message confidence that it was sent by the expected party [12].

Digital signature used to detect whether or not the information was modified after it was. These assurances may
be obtained whether the data was received in a transmission or retrieved from storage. A digital signature
algorithm includes a signature generation process and a signature verification process. A signatory uses the
generation process to generate a digital signature on data, a verifier uses the verification process to verify the
authenticity of the signature. Each signatory has a public and private key and is the owner of that key pair. The
private key is used in the signature generation process. The key pair owner is the only entity that is authorized to
use the private key to generate digital signatures. In order to prevent other entities from claiming to be the key
pair owner and using the private key to generate fraudulent signatures, the private key must remain secret [7].

2. **RSA Digital Signature**

The RSA algorithm was developed at Massachusetts Institute of Technology (MIT) in 1977 by Ron Rivest, Adi
Shamir and Leonard Adelman. The RSA concept is based on the factorization of big numbers which means the
larger sequence of numbers you have, the more you are protected. The RSA provides a strong security; therefore
an adversary should not be able to break RSA by factoring due to its complexity and large keys. RSA is used to
encrypt/decrypt data and also has the ability to sign and/or verify the data packets. RSA does not mandate the
use of a particular hash function, so the security of the signature and encryption are partly dependent on the
choice of hash function used to compute the signature [8]. The security assumption was based on the intractable
complexities of factoring a large composite integer $n = p^aq$, where $p$ and $q$ are two distinct large primes [5].
RSA is an asymmetric digital signature algorithm as it uses a pair of keys, one of which is used to sign the data in such a way that it can only be verified with the other key. RSA is based on one way trap-door function. In case of RSA, the idea is that it is relatively easy to multiply prime numbers but much more difficult to factor. Multiplication can be computed in polynomial time whereas factoring time can grow exponentially proportional to the size of the numbers. The algorithm is as follows [10]:

2.1. Key Generation

The followings are the key generation steps:

- Generate two large random primes, p and q.
- Compute \( n = p \times q \) and \( \varphi = (p-1) \times (q-1) \).
- Choose an integer e, satisfying \( 1 < e < \varphi \), such that \( \gcd(e, \varphi) = 1 \).
- Compute the secret exponent d, satisfying \( 1 < d < \varphi \), such that \( e \times d \mod \varphi = 1 \).
- The public key is e and the private key is d. By using these keys, signature generation and signature verification are performed.

2.2. Signature Generation

The followings are the signature generation steps:

- Creates a message digest \( H(m) \) as an integer of the information to be sent between 0 and \( n - 1 \).
- Compute the signature by using the private key d as \( s = H(m)^d \mod n \)
- s is the signature of the message m. Send s with the message m to recipient.
- Signature Verification:
- Signature verification steps are as follows:
  - Using sender public key e, compute integer \( v = s^e \mod n \). v is message digest calculated by sender.
  - Independently computes the message digest of the message that has been signed.
  - If both message digests are identical, the signature is valid.

3. Prime number

A prime number is a positive integer greater than 1 whose only positive integer divisors are 1 and itself [15]. A prime is a positive integer p having exactly two positive divisors, namely 1 and p. An integer n is composite if \( n > 1 \) and n is not prime. (The number 1 is considered neither prime nor composite.) Thus, an integer n is composite if and only if it admits a nontrivial factorization \( n = ab \), where a, b are integers, each strictly between 1 and n [16].

3.1. Definition

An integer \( p \geq 2 \) is said to be prime if its only positive divisors are 1 and p. Otherwise, p is called composite
3.2. Definition

A prime number $p$ is said to be a strong prime if integers $r$, $s$, and $t$ exist such that the following three conditions are satisfied [15]:

- $p - 1$ has a large prime factor, denoted $r$.
- $p + 1$ has a large prime factor, denoted $s$.
- $r - 1$ has a large prime factor, denoted $t$.

3.3. Algorithm

Gordon’s algorithm for generating a strong prime: a strong prime $p$ is generated as [15].

- Generate two large random primes $s$ and $t$ of roughly equal bit length.
- Select an integer $i_0$. Find the first prime in the sequence $2it + 1$, for $i = i_0, i_0 + 1, i_0 + 2,...$. Denote this prime by $r = 2it + 1$.
- Compute $p_0 = 2(s^{r-2} mod r)s - 1$.
- Select an integer $j_0$. Find the first prime in the sequence $p_0 + 2jrs$, for $j = j_0, j_0 + 1, j_0 + 2,...$. Denote this prime by $p = p_0 + 2jrs$.
- Return(p).

4. Related work

RSA is an asymmetric digital signature algorithm which is the most popular public key cryptosystem, there are several studies and researches have been proposed on RSA for efficiency and security. Xianmeng Meng, Xuexin Zheng were revisited the birthday attack against short exponent RSA, they show that if $e > \sqrt{k(p + q)}$, then $N$ can be factored in both time and space complexity of $O(\sqrt{k})$, they improved the former result [1]. Santanu Sarkar, Subhamoy Maitra were proposed a different lattice based technique to show that RSA is weak beyond this bound. Their analysis provided improved results and it showed that for two exponents, RSA is weak when the RSA decryption exponents are less than $N^{0.416}$ [2]. Santanu Sarkar, Subhamoy Maitra analyzed the security of the RSA public key cryptosystem where multiple encryption and decryption exponents are considered with the same RSA modulus $N$. their result improved the bound of Howgrave-Graham and Seifert (CQRE 1999) for $N \geq 42$ and also generalized their work for $N = 2$ (IPL 2010) [3]. Reducing the search range of a certain parameter $k$, which is a bottleneck of Heninger–Shacham attack, was proposed by ShigeyoshiImai, KaoruKurosawa [4]. M. Thangavel, P. Varalakshmi, Mukund Murali, K. Nithya, were proposed a modified and an enhanced scheme based on RSA public-key cryptosystem. The proposed algorithm makes use of four large prime numbers which increases the complexity of the system as compared to traditional RSA algorithm which is based on only two large prime numbers [6]. Kamal Kr. Gola, Bhumika Gupta, Zubair Iqbal were proposed a modified RSA digital signature scheme for data confidentiality is to provide the data confidentiality during the
data transfer by using the concept of public key encryption [9]. The presenting of a new variant of digital
signature algorithm that based on two hard problems, prime factorization and xth root problem. That is a
modification of the RSA digital signature algorithm were proposed by Ashish Vijay, Priyanka Trikha, Kapil
signature scheme using a novel message digest algorithm, Algorithm for Secure Hashing-160 ‘ASH-160’. The
proposed scheme has been implemented and the results analyzed and compared with RSA digital signature
scheme using SHA1 and RIPEMD160 [11]. Hongjie Zhu, Daxing Li were proposed a kind of digital signature
based on public key. They are effectively realized both digital signature and defending illegal interpolation and
replication of digital products [13]. Dindayal Mahto, Danish Ali Khan, Dilip Kumar Yadav were analyzed the
security strength of the RSA and ECC. The security of the RSA cryptosystem is based on the Integer
Factorization Problem and the security of ECC is based on elliptic curve discrete logarithm problem [14].

5. Proposed Scheme

RSA digital signature is the most popular public key cryptosystem, and the security of RSA algorithm is
depending on the difficulty of solving the prime numbers factorization problem. There are many efforts have
been done in past to solve the prime factorization problem. In this paper we propose a new RSA digital
signature scheme based on the concept of a strong prime numbers, such that $p = 2p_0 + 1$, $q = 2q_0 + 1$ and $p_0, q_0$
are prime numbers. To create the public and the private key, the generating of a strong prime numbers,
calculated by using the Gordon’s algorithm. Then we analyze the new proposed scheme among the normal key
generation of the RSA Digital Signature using different keys length of 256, 512, 1024, 2048, 4096, and 8192. In
order to show and find out the analysis variations of the improvement and getting a good results that optimize
the key generation strategy, and solving the prime factorization. The new proposed scheme of RSA digital
signature moves throw three phases, as follows:

5.1. Key generation phase

In this phase, the message signer generates the private key, $d$, and public key $(n, e)$. The details of this phase for
the signer are as following steps:

- Generate two large random primes $p_0$ and $q_0$.
- Compute $n = p \times q$ and $\varnothing = (p - 1) \times (q - 1)$.
- Choose random integer $e$, satisfying $1 < e < \varnothing$, such that $\gcd(e, \varnothing) = 1$.
- Compute the secret signature key $d$, satisfying $1 < d < \varnothing$, such that $e \times d \mod \varnothing = 1$.
- Send The public key is $(n, e)$ to the recipient.

5.2. Signing Generating phase

In this phase, the signer inputs a message $(m)$ and his or her private key $(d)$, to make an output of a digital
signature $(s)$. The details of this phase for the signer are as follows:
5.3. Signature verification phase

In this phase, for a given message \( m \), a signer’s public key \((n, e)\) and a digital signature \(s\), the recipient decide whether to accept or reject the signature. The details of this phase for the receiver are as follows:

- Obtains the public key \((n, e)\).
- Receives the message \( m \) and its signature \( s \) from the signer.
- Applies hash function to the received message \( H(m) \) to compute the message digest.
- Compute integer \( v = s^e \mod n \). \( v \) be the message digests calculated by sender.
- Verifies that \( v = H(m) \), if not, then rejects the signature. Otherwise accepts the signature.

6. Results

The proposed scheme was tested on a different keys length of 256, 512, 1024, 2048, 4096, and 8192, in order to represent the performance of the private and public key generation time, and analyzed by a different statistical method. The analysis show that how much time it taken by the algorithm to create the Private and Public keys using different key lengths.

The mean time generated by normal prime key generation recorded a little time variation due to its small up to medium key sizes compared with strong prime key generation.

But it can be seen that the time for key generation of Strong Prime is slightly less than that of Normal prime for keys larger than 4096. The results of analysis shown as in following figures and tables:
Figure 2: Key Generation Time, staked line with Marker chart (in seconds)

Figure 3: Key Generation Time, Clustered Column Chart (in seconds)

Table 1: The Result of the Key Size vs Prime Number

<table>
<thead>
<tr>
<th>Prime Number</th>
<th>Mean</th>
<th>N</th>
<th>Std. Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal Prime</td>
<td>139.63</td>
<td>30</td>
<td>352.811</td>
</tr>
<tr>
<td>Strong Prime</td>
<td>173.67</td>
<td>30</td>
<td>386.785</td>
</tr>
<tr>
<td>Total</td>
<td>156.65</td>
<td>60</td>
<td>367.439</td>
</tr>
</tbody>
</table>

Table 2: The Result of Key Generation Time IN SECONDS using Nova Analysis

<table>
<thead>
<tr>
<th></th>
<th>Sum of Squares</th>
<th>df</th>
<th>Mean Square</th>
<th>Sig.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Between Groups .000</td>
<td>1 .000</td>
<td>7.534.008</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Within Groups .000</td>
<td>58.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sum of Squares</td>
<td>df</td>
<td>Mean Square</td>
<td>F</td>
<td>Sig.</td>
</tr>
<tr>
<td>----------------</td>
<td>----</td>
<td>------------</td>
<td>------</td>
<td>-------</td>
</tr>
<tr>
<td>Between Groups</td>
<td>.000</td>
<td>7.534.008</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Within Groups</td>
<td>.000</td>
<td>58.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>.000</td>
<td>59</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table 3: The Case Processing Summery of Key Size vs Prime Number

<table>
<thead>
<tr>
<th>Cases</th>
<th>Included</th>
<th>Excluded</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>Percent</td>
<td>N Percent</td>
<td>N Percent</td>
</tr>
<tr>
<td>KeySize * Results</td>
<td>60</td>
<td>100.0%</td>
<td>0</td>
</tr>
<tr>
<td>KeySize * PrimeNumber</td>
<td>60</td>
<td>100.0%</td>
<td>0</td>
</tr>
</tbody>
</table>

### 7. Conclusion

The proposed work has been implemented and the results are analyzed and compared with the normal key generation of RSA digital signature scheme using different keys length of 256, 512, 1024, 2048, 4096, and 8192. From the analysis of experimental results above we show that the computation time taken to generate the private and public key using a normal prime key generation recorded a little time variation for small up to medium key sizes, compared with strong prime key generation. But the time for key generation of Strong Prime significantly less than the traditional Prime for keys larger than 4096. The decreasing time of largest key generation slightly leads to the improvement and efficiency of RSA digital signature. Thus it seems there is not much overhead or burden on the system.

### References

6. M. Thangavel, P. Varalakshmi, Mukund Murrali, K. Nithya,” An Enhanced and Secured RSA Key Generation Scheme (ESRKGS)”, journal of information security and applications 20,
www.elsevier.com/locate/jisa, 2015, p. 3-10.


