Experimental Analysis of Attacks on RSA & Rabin Cryptosystems using Quantum Shor's Algorithm

Ritu Thombre[1]* and Babita Jajodia[2]

[1] Department of Computer Science and Engineering, Visvesvaraya National Institute of Technology, Nagpur, India

[2] Department of Electronics and Communication Engineering, Indian Institute of Information Technology Guwahati, Guwahati, India

* Corresponding author

doi:https://doi.org/10.21467/proceedings.114.74

ABSTRACT

In this world of massive communication networks, data security and confidentiality are of crucial importance for maintaining secured private communication and protecting information against eavesdropping attacks. Existing cryptosystems provide data security and confidentiality by the use of encryption and signature algorithms for secured communication. Classical computers use cryptographic algorithms that use the product of two large prime numbers for generating public and private keys. These classical algorithms are based on the fact that integer factorization is a non-deterministic polynomial-time (NP) problem and requires super-polynomial time making it impossible for large enough integers. Shor's algorithm is a well-known algorithm for factoring large integers in polynomial time and takes only $O(b^3)$ time and $O(b)$ space on b-bit number inputs. Shor's algorithm poses a potential threat to the current security system with the ongoing advancements of Quantum computers. This paper discusses how Shor's algorithm will be able to break integer factorization based cryptographic algorithms, for example, Rivest–Shamir–Adleman (RSA) and Rabin Algorithms. As a proof of concept, experimental analysis of Quantum Shor's algorithm on existing public-key cryptosystems using IBM Quantum Experience is performed for factorizing integers of moderate length (seven bits) due to limitations of thirty-two qubits in present IBM quantum computers. In a nutshell, this work will demonstrate how Shor's algorithm poses threat to confidentiality and authentication services.

Keywords: Asymmetric Cryptography, Digital Signature, Encryption, IBM, Rabin, Rivest–Shamir–Adleman (RSA), Shor's Algorithm, Qiskit, Quantum Computing, Quantum Cryptography.

I. Introduction

For establishment of secured private communication and protecting information against eavesdropping attacks, there is a requirement of cryptosystems providing information security services of confidentiality [1] and message authentication [2]. Existing cryptosystems uses encryption and signature algorithms that provides data confidentiality and authentication between Alice (the sender) and Bob (the receiver), respectively.

Basically, there are two types of cryptosystems: symmetric cryptosystems and asymmetric cryptosystems. Symmetric cryptosystems employ the same private key for each of the operations (e.g., encryption and decryption). Data Encryption Standard (DES) and Advanced Encryption Standard (AES) [3] are the most commonly used symmetric cryptosystems. Unlike symmetric-key, asymmetric cryptosystems (known as public-key cryptosystems) have two distinctive keys: a private key and a public key without compromising the secrecy of the private key [4]. The most common used asymmetric crypto-systems are: Rivest–Shamir–Adleman (RSA) [5] and Rabin [6] cryptosystems. Asymmetric cryptosystems use the concept of product of two large prime integers for key generation; that will be discussed in details in Section II. Large integer factorization might not be possible using classical computers

Published by AIJR Publisher in the "Proceedings of International Conference on Women Researchers in Electronics and Computing" (WREC 2021) April 22–24, 2021. Organized by the Department of Electronics and Communication Engineering, Dr. B. R. Ambedkar National Institute of Technology, Jalandhar, Punjab, INDIA

Proceedings DOI: 10.21467/proceedings.114; Series: AIJR Proceedings; ISSN: 2582-3922; ISBN: 978-81-947843-8-8

Algorithm 1: RSA Encryption Algorithm [13]

```
Variables:
    Public Key: Tuple $(e, n)$
    Private Key: Integer $d$
    Plaintext: $P$
    Ciphertext: $C$
Functions:
    Carmichael's Totient function: $\phi(n)$
Key Generation:
    Choose two large primes $p$ and $q$ such that $p \neq q$
    Calculate $n = p \times q$
    Calculate $\phi(n) = (p - 1) \times (q - 1)$,
    Choose $e$ such that $gcd(e, \phi(n)) = 1$
    Calculate $d = e^{-1} mod(\phi(n)))$
Encryption: $C = P^e \ mod(n)$
Decryption: $P = C^d \ mod(n)$
```

in a feasible time; but the present asymmetric cryptosystems are still safe against eavesdropping or cryptographic attackers till the advent of quantum computers.

Recent developments in Quantum Computing (QC) over the years [7], [8] show that quantum computing algorithms could outperform existing classical algorithms in terms of time complexity [9]. For example, Quantum Grover's algorithm provide a drastic quadratic speedup in searching an unstructured data [10]. Quantum Shor's algorithm factorize integers in polynomial time [11] over the best classical integer factorization algorithms [12] with exponential time complexity. This work discusses how Quantum Shor's algorithm poses threat to asymmetric cryptosystems and can potentially break RSA and Rabin cryptosystems that use prime products for key generation. The authors have also presented experimental analysis of attacks using open-source IBM quantum systems on existing cryptosystems.

The remainder of the paper is organized as follows. Section II gives a brief on existing asymmetric public-key cryptosystems based on products of large prime integers. Section III discusses about Quantum Shor's Algorithm. Section IV presents experimental analysis of attacks on existing asymmetric cryptosystems using Quantum Shor's Algorithm on IBM Quantum Computers. This is followed by conclusion in Section V.

II. Background on Asymmetric Public-Key Cryptosystems

A cryptosystem is a suite of cryptographic algorithms and their accompanying infrastructure that provides information security services of confidentiality [1] and message authentication [2]. Encryption algorithms provide message confidentiality[1] between the sender and the receiver. [5]. The sender sends a message to the receiver by encrypting message using a public key already present in receiver's certificate provided by Certification Authority [14]. The receiver decrypts the sender's message using his/her own private key. Signature algorithms provide message authentication (i.e. proof of origin of the message) of the sender to the receiver [15]. Suppose, the sender wants to provide authentication of his/her message to the receiver, the sender signs the message using his/her own private key; the receiver verifies signature using the sender's public key present in the sender's certificate provided by Certification Authority [14].

This section discusses about existing asymmetric public-key cryptographic algorithms based on products of large prime integers: (a) RSA Cryptosystem [5] and (b) Rabin Cryptosystem [6].

A. RSA Cryptosystem [5]

RSA cryptosystem uses modular exponentiation for encryption/decryption. Modular exponentiation use fast exponentiation algorithm [5], [13] with two exponents, e and d. Here, e is a part of the public key (e, n) and

[1] To ensure the security of RSA and El-Gamal cryptosystems in algorithms 1,2,3 and 4, p and q must be at least of 512 bits, so that n will be at least of 1024 bits.

Algorithm 2: RSA Signature Algorithm [13]

```
Variables:
    Public Key: Tuple $(e, n)$
    Private Key: Integer $d$
    Message: $M$
    Signature: $S$
    Message retrieved from Signature: $M'$
Functions:
    Carmichael's Totient function: $\phi(n)$
Key Generation:
    Choose two large primes $p$ and $q$ such that $p \neq q$
    Calculate $n = p \times q$
    Calculate $\phi(n) = (p - 1) \times (q - 1)$,
    Choose $e$ such that $gcd(e, \phi(n)) = 1$
    Calculate $d = e^{-1} mod(\phi(n)))$
Signing: $S = M^d\ mod(n)$
Verification: $M' = S^e\ mod(n)$
    The signature is valid if and only if $M' = M$
```

Algorithm 3: Rabin Encryption Algorithm [13]

```
Variables:
    Public Key: $n$
    Private Key: $(p, q)$
    Plaintext: $P$
    Ciphertext: $C$
Key Generation: Choose two large primes $p$ and $q$ in the form $4k + 3$ and $p \neq q$, and calculate $n = p \times q$
Encryption: $C = P^2\ mod(n)$
Decryption:
    Calculate four square roots of $C$:
    $a_1 = (C^{(p+1)/4}) mod(p)$
    $a_2 = (C^{(p-1)/4}) mod(p)$
    $b_1 = (C^{(q+1)/4}) mod(q)$
    $b_2 = (C^{(q-1)/4}) mod(q)$
    Calculate Chinese remainders:
    $r_1$ =Chinese_Remainder($a_1 mod(p), b_1 mod(q)$)
    $r_2$ =Chinese_Remainder($a_2 mod(p), b_1 mod(q)$)
    $r_3$ =Chinese_Remainder($a_1 mod(p), b_2 mod(q)$)
    $r_4$ =Chinese_Remainder($a_2 mod(p), b_2 mod(q)$)
    $P \in (r_1, r_2, r_3, r_4)$
```

Algorithm 4: Rabin Signature Algorithm [13]

```
Variables:
    Public Key: $n$
    Private Key: $(p, q)$
    Message: $m$
    Signature: Tuple $(r, u)$
Functions:
    Cryptographic Hash function: H
Key Generation: Choose two large primes $p$ and $q$ in the form $4k + 3$ and $p \neq q$, and calculate $n = p \times q$
Signing:
    1) $c = H(m|u)$, where u is the random number
    2) Decrypt $c$ to produce $(r_1, r_2, r_3, r_4)$
    3) $r \in (r_1, r_2, r_3, r_4)$ such that encryption($r$) = $c$
    4) signature is $(r, u)$.
Verification:
    Signature $(r, u)$ for message $m$ verification $n$:
    1) Compute $c = H(m|u)$ and encrypt $r$
    2) The signature is valid if & only if the encryption of $r$ equals $c$.
```

d is private (the secret key). Suppose Alice send a message to Bob by generating ciphertext $C = P^e mod(n)$ for plaintext P; Bob retrieves plaintext P using $P = C^d mod(n)$ sent by Alice. Alice and Bob can encrypt (e is public) and decrypt (because he knows d) in polynomial time, respectively; but for eavesdropping attack, Eve need to calculate $\sqrt[e]{C}\ mod\ (n)$ using modular arithmetic.

The encryption and signature algorithms of RSA cryptosystems are illustrated in Algorithm 1 and Algorithm 2 respectively.

B. Rabin Cryptosystem [6]

M. Rabin devised the Rabin cryptosystem [6] based on quadratic congruence; whereas RSA is based on exponentiation congruence. This is a special case of RSA cryptosystem where the values e= 2 and d= $1/2$ are considered. Here, encryption and decryption can be given by $C = P^2\ mod\ (n)$ and $P = C^{\frac{1}{2}}\ mod(n)$, respectively. Alice can encrypt a message using n; only Bob can decrypt the message using p and q. In order for an eavesdropper to decrypt the ciphertext, he/she has to solve integer factorization problem to obtain (p, q), which are factors of n. Unlike RSA, Rabin cryptosystem is non-deterministic; decryption of a ciphertext C creates four equally probable plaintexts (P_1, P_2, P_3, P_4) and Bob can choose one out of the four answers as correct answer.

The encryption and signature algorithms of Rabin cryptosystems are illustrated in Algorithm 3 and Algorithm 4 respectively.

III. Quantum Shor's Algorithm

The famous American mathematician Peter Shor invented Quantum Shor's Algorithm in 1994 that factorizes integers in polynomial time [11] over the best classical integer factorization algorithms [12] with sub-exponential time complexity being impossible for large integers.

Quantum Shor's algorithm solves the problem of *period finding* in polynomial time which is an efficient way for factorization of integers into prime factors. This algorithm is based on (a) finding period of a modulus function $f(x) = a^x\ mod(N)$ using period-finding problem and then, (b) find the prime factors of $f(x)$ using Quantum Phase Estimation (QPE) and Quantum Fourier Transform (QFT) [9].

For better illustration, considering two non-negative integers a and N, $a < N$ and $gcd\ (a, N) = 1$ for the periodic function $f(x) = a^x\ mod(N)$, $x \in 0, \ldots, N-1$, the period r of $f(x)$ can be written as

$$a^r \equiv 1\ mod(N) \tag{1}$$

for some integer $r \geq 1$ and r is the smallest (non-zero) integer; $f(x)$ is periodic with $period = r$ and obeys $a^{x+r} \equiv 1 mod(N)$. The prime factors of N can be calculated as

$$\begin{aligned} a^r &\equiv 1\ mod(N) \\ \therefore a^r - 1 &\equiv 0\ mod(N) \\ \therefore (a^{r/2}+1)(a^{r/2}-1) &\equiv 0\ mod(N) \end{aligned} \tag{2}$$

Based on equation (2), it can be stated that either of the $(a^{r/2}+1)$ or $(a^{r/2}-1)$ have a common factor with N. Hence, $gcd(a^{r/2}+1, N)$ or $gcd(N, a^{r/2}-1)$ are the possible factors of N.

QPE helps in finding the period r of the function $f(x)$; the factors are found out using classical computation repeating QPE for different values of a, till an even value of r can be found out. But, in case of the period r to be odd, QPE is repeated for different values of a since $a^{r/2}$ will not be an integer for odd value of r. The following sub-sections provides a brief details of QFT, QPE and period finding solution.

A. Quantum Fourier Transform (QFT) [9]

The quantum implementation of the Discrete Fourier transform (DFT) over the amplitudes of a wave function is Quantum Fourier Transform (QFT) [16]. DFT acts on vectors $(x_0, x_1, x_2, \ldots, x_{N-1})$ and maps it to the vector $(y_0, y_1, y_2, \ldots, y_{N-1})$ as

$$y_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n \omega_N^{-kn} \tag{3}$$

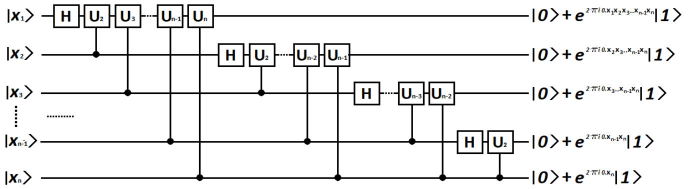
$|x_1\rangle$
$|x_2\rangle$
$|x_3\rangle$
$|x_{n-1}\rangle$
$|x_n\rangle$
H
U_2
U_3
U_{n-1}
U_n
H
U_2
U_{n-2}
U_{n-1}
H
U_{n-3}
U_{n-2}
H
U_2
$|0\rangle + e^{2\pi i\, 0.x_1x_2x_3..x_{n-1}x_n}|1\rangle$
$|0\rangle + e^{2\pi i\, 0.x_2x_3..x_{n-1}x_n}|1\rangle$
$|0\rangle + e^{2\pi i\, 0.x_3..x_{n-1}x_n}|1\rangle$
$|0\rangle + e^{2\pi i\, 0.x_{n-1}x_n}|1\rangle$
$|0\rangle + e^{2\pi i\, 0.x_n}|1\rangle$

Fig. 1. Circuit for Quantum Fourier Transform (QFT)

where, $k = 0, 1, 2, \ldots, N-1$ and $\omega = e^{2\pi i/N}$. Similarly, QFT acts on a quantum state $|x\rangle = \sum_{i=0}^{N-1} |i\rangle$ and maps it to the quantum state $|y\rangle = \sum_{i=0}^{N-1} |i\rangle$ as:

$$y_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n \omega_N^{kn}, \tag{4}$$

where, $k = 0, 1, 2, ..., N-1$ and $\omega = e^{2\pi i/N}$. The inverse Quantum Fourier Transform (IQFT) can be represented as:

$$y_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n \omega_N^{-kn}, \tag{5}$$

where, $k = 0, 1, 2, \ldots, N-1$, $\omega = e^{2\pi i/N}$. Fig. 1 shows the quantum circuit implementation of QFT consisting of Hadamard gates and Controlled U1 gates as the circuit components. A Hadamard gate when applied to a qubit in a circuit it puts it in to a superposition of states such that when it is measured it could be 0 or 1 with equal probability. A Controlled U1 gate is a gate which is used to implement a single rotation around the Z-axis (phase) of the target qubit if the control qubit is 1. A Controlled NOT (CNOT) gate is a multi-qubit gate that operates on a qubit based upon the state of another qubit. If the control qubit is 1, the target qubit will be flipped from 0 to 1 or vice versa. Else, if the control qubit is 0, the target qubit won't be flipped. Mathematically, the Hadamard gate and the controlled U1 gate can be written in matrix form as

$$U_k = \begin{bmatrix} 1 & 0 \\ 0 & e^{2\pi i/2^k} \end{bmatrix} \tag{6}$$

$$H = \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix} \tag{7}$$

It can also be stated that, QFT transforms states in the computational (Z) basis to the Fourier basis. Applying QFT on $|0\rangle$ and $|1\rangle$ in Z basis, the obtained states are $|+\rangle$ and $|-\rangle$ respectively in X basis.

B. Quantum Phase Estimation (QPE) [9]

Any quantum state $|\psi\rangle$ can be represented by a point on the Bloch Sphere as

$$|\psi\rangle = \cos(\theta/2) + e^{i\phi}\sin(\theta/2) \tag{8}$$

where ϕ is the phase angle of $|\psi\rangle$ and this can be determined by using QPE algorithm and the unitary operation:

$$U|\psi\rangle = e^{2\pi i\phi}|\psi\rangle, 0 \leq \phi < 1 \tag{9}$$

Here, $|\psi\rangle$ is an eigenvector of operator U, and $e^{2\pi i\phi}$ is its corresponding eigenvalue. QPE algorithm uses phase-kickback for writing the phase of U into t counting qubits in Fourier basis, which can then be converted to computational basis by applying IQFT. Fig. 2 shows the circuit for QPE algorithm using t-bit precision. Applying

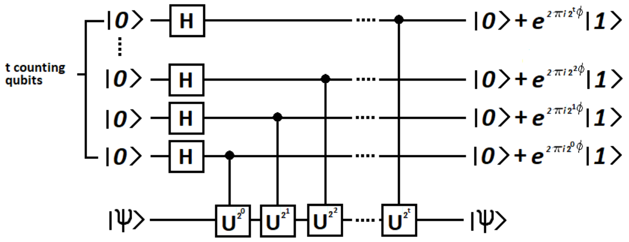

Fig. 2. Circuit for Quantum Phase Estimation (QPE) using t-bit precision

controlled-U, controlled-U^2, controlled-U^4, and so on to the initial state with t qubits $|0_1 0_2 \dots 0_n\rangle$, produces the quantum state: $(|0\rangle + e^{2\pi i 2^{n-1}\phi}|1\rangle)(|0\rangle + e^{2\pi i 2^{n-2}\phi}|1\rangle \dots (|0\rangle + e^{2\pi i\phi}|1\rangle)$ which is equal to $\sum_{y=0}^{2^n-1} e^{2\pi i\phi y}|y\rangle$. This quantum state is exactly the state when QFT is applied in the Fourier basis. Therefore, the phase angle $\phi = 0.x_1 x_2 \dots x_n$ can be computed by applying IQFT.

Note: ϕ may not always be a rational number; hence, it turns out that applying IQFT produces the best n-bit ($n = t$ counting bits in this case) approximation of ϕ with probability at least of $4/\pi^2 = 0.405$. But, by using a higher number of counting bits t, ϕ leads to a better approximation of ϕ.

C. Period Finding and Integer Factorization

For factorization of integer N, Shor's solution uses QPE on the unitary operator [16]

$$U^x|\psi\rangle = |a^x \psi mod(N)\rangle \tag{10}$$

for periodic function $f(x) = a^x mod(N)$ with some integer a. QPE is applied on the qubits starting with counting qubits (input register) and ancilla qubits (output register) initialized to $|0\rangle$, using equation (10) so that period r of $f(x) = a^x mod(N)$ is stored in counting bits in Fourier basis, which later, can be converted into computational basis by applying IQFT.

Since f is periodic, the measurement probabilities of possible outcomes y in the input register is stated by $(|\sum_{x:f(x)=f(x_0)} e^{2\pi ixy/N}|^2)/N$, i.e. $(|\sum_b e^{2\pi i(x_0+rb)y/N}|^2)/N$ with $f(x_0)$ in the output register.

It is observed that as the value of yr/N converges to an integer, probability outcome is higher. Thus, turning y/N to an irreducible fraction, the denominator of this fraction r', is a probable candidate for r. If $f(x) \neq f(x + r$'$)$, it will be terminated and different values of r closer to y, or multiples of r' are tested; this process need to be repeated for different values of a till a particular r is found out.

Fig. 3 shows the quantum circuit for period finding solution of Shor's Algorithm. Later, the prime factors of N can be found out using period finding solution (Fig. 3) and (2) on classical computations and can be illustrated in the following steps:

1) Pick a random number a and calculate $gcd(a, N)$.
2) There is a probability that the random number picked (a) could be a non-trivial factor of N i.e $gcd(a, N) \neq 1$.If so, then terminate the algorithm.
3) If random number picked (a) is a trivial factor of N i.e $gcd(a, N) = 1$, then use the quantum period-finding algorithm to find the period r.
4) If r is odd or $a^{r/2}$ is a trivial factor of N (i.e $a^{r/2} \equiv -1(\text{mod} N)$), then go back to step 1. Else, either of the $gcd(a^{r/2} + 1, N)$ and $gcd(a^{r/2} - 1, N)$ are non-trivial factors of N.

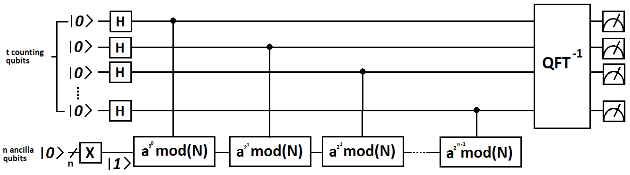

Fig. 3. Quantum Circuit for Period Finding using t-bit precision

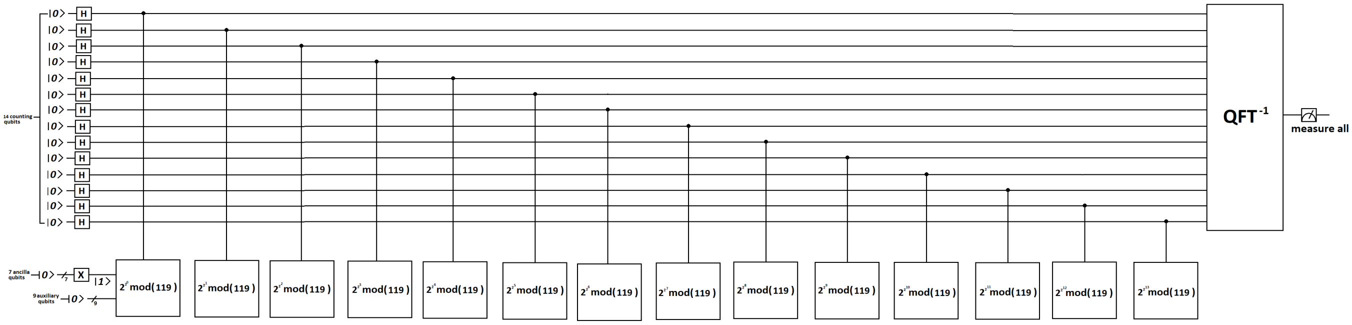

Fig. 4. Quantum Circuit for Shor's algorithm of integer factorization $n = p \times q$ ($119 = 17 \times 7$)

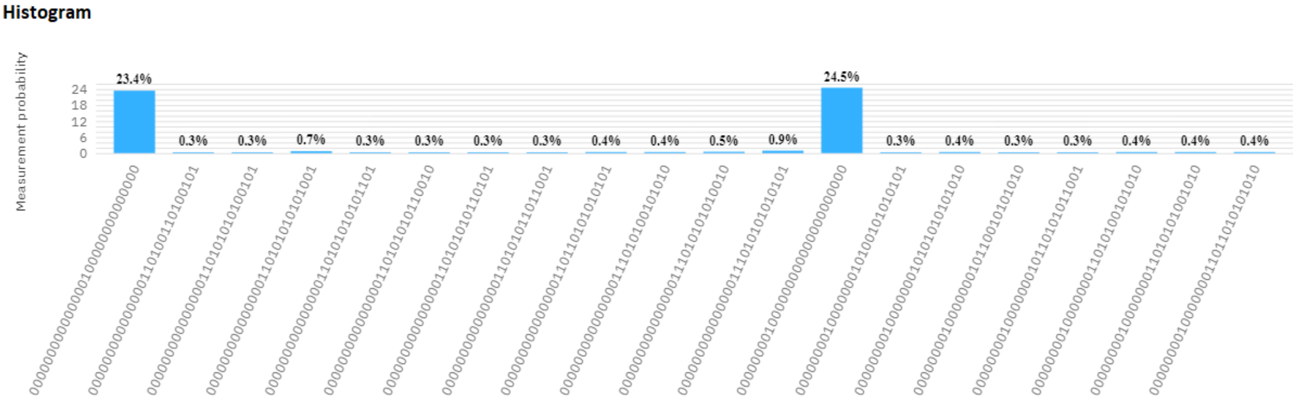

Fig. 5. Measurement Results obtained from IBM Quantum Experience for integer factorization $n = p \times q$ ($119 = 17 \times 7$) using Quantum Shor's Algorithm

IV. Analysis of Attacks using Quantum Shor's Algorithm

Shor's algorithm can be used to attack assymmetric cryptosystems based encryption algorithms [17] and signature algorithms [18], [19]: (a) RSA Cryptosystem (b) Rabin Cryptosystem.

A. *Attacks on RSA Cryptosystems*

The RSA key generation process is the same for both encryption and signature algorithms discussed in Algorithm 1 and Algorithm 2 respectively. Large number n can be easily factorized into p and q using Quantum Shor's algorithm; and after integer factorization, $\phi(n)$ and d can be calculated using p and q as e is public. Since d is a private key of some entity, ciphertext C sent to this entity encrypted by entity's public key (e, n) is no longer secure as it can be decrypted using d. Further, d can be used to forge the signature of that entity which poses threat to authentication of this entity, leading to total break [19] of the entire RSA cryptosystems.

B. Attacks on Rabin Cryptosystems

The key generation process involved in Rabin cryptosystem is the same for both the encryption and signature schemes already discussed in Algorithm 3 and Algorithm 4 respectively.

Quantum Shor's algorithm can easily factorize n into prime factors: p and q since n is public. Next, (p, q) is a private key of some entity, ciphertext C sent to this entity encrypted by entity's public key n is no longer secure as it can be decrypted using (p, q). Further, (p, q) can be used to forge the signature of that entity which poses threat to authentication of this entity, leading to total break [19] of the entire Rabin cryptosystems.

C. Experimental Analysis

Solving large integer factorization in polynomial time complexity poses direct threat to confidentiality and authentication services of a cryptographic system that uses prime products for key generation. Experiments on integer factorization using Shor's Algorithm were performed on IBM Quantum Experience: $ibm\_qasm\_simulator$ as backend having 32 qubits. Basically, Shor's algorithm uses $4n_{bit} + 4$ qubits for factorizing an large integer with n_{bit} bits leading to the maximum value of integer $n < 128$ and this can be calculated as:

$$\begin{aligned} 4n_{bit} + 4 &\leq 32 \\ n_{bit} &\leq 7 \\ \therefore n &< (2^{n_{bit}} = 128) \end{aligned} \tag{11}$$

Quantum Experiments were easily carried out for demonstrating successful attacks on RSA and Rabin cryptosystems by integer factorization for integers n less than or equal to 127 due to limitations of number of qubits presently on IBM quantum computers. Fig. 4 shows the circuit for Quantum Shor's Algorithm for factorization of $n = 119$ into two prime products: $p = 17$ and $q = 7$ (Note: This is considered as an example for better illustration).

Fig. 5 shows the measurement probabilities (in percentage) of the Quantum Shor's circuit (Fig. 4). From Fig. 5 it can be observed that the maximum possible counts (in %) are 23.4% and 24.5% for the quantum states 00000000000000000000100000000 and 00000000000001000000000000000, respectively. (Note: Little-endian format is followed for representation of quantum states in IBM system).The states can be represented in integers as 256_{10} and 32768_{10}. The possible phases for the quantum states are: $256/2^{14} = 1/64$ and $32768/2^{14} = 2/1$; assuming the total number of counting qubits is 14. Both $1/64$ and $2/1$ are in the form of irreducible fractions, the possible values of the period r are the denominators of the fractions, i.e. 64 and 1, respectively. $r = 64$ is the period from the period finding algorithm using the relation: $f(x)$= 2^x $mod(119)$. Next, the possible prime factors of 119 can be either one of the $gcd(2^{64/2} - 1, 119) = 17$ or $gcd(2^{64/2} + 1, 119) = 1$. Therefore, one of the prime factor is p = 17. (Note: 1 is not considered as the prime factor of n). The other prime factor q is division of n= 119 by prime factor p (obtained from Shor's algorithm), i.e., 119/17= 7.

Presently, there are few practical limitations of running existing RSA and Rabin cryptosystems with prime factors (p and q) up to length of 512-1024 bits and hence n of 1024-2048 bits [13] on IBM Quantum computers due to availability of only up to 32 qubits [20]. Future works will focus on experimental analysis of 2048-bit integer factorization using 8196 ($4 * 2048 + 4 = 8196$) qubits for RSA and Rabin cryptosystems with the advent of IBM systems using noisy qubits [21], [22].

V. Conclusions

This paper successfully demonstrated how Quantum Shor's algorithm will be able to break integer factorization based asymmetric cryptographic algorithms of Rivest–Shamir–Adleman (RSA) and Rabin cryptosystems. Experimental analysis on integer factorisation were performed for integers factorizing integers moderate in length (seven bits) due to limitations of thirty-two qubits in present IBM quantum computers. Implementation of Quantum Shor's algorithm on large scale poses potential threat to to confidentiality and authentication services of currently used security systems. This necessitates the development of Quantum Cryptography [23] using quantum superposition and quantum entanglement for encrypting data; making it virtually unhackable and preventing it from eavesdropping attacks, such as, Quantum Key Distribution (QKD) [24] and BB84 [25] protocols that are secured against cryptographic attacks [26], [27].

References

[1] Z. Chen, J. Wang, Z. Zhang, and X. Song, "A fully homomorphic encryption scheme with better key size," *China Communications*, vol. 11, no. 9, pp. 82–92, 2014.

[2] L. Venkatraman and D. P. Agrawal, "A novel authentication scheme for ad hoc networks," in *2000 IEEE Wireless Communications and Networking Conf. Conf. Record (Cat. No.00TH8540)*, vol. 3, 2000, pp. 1268–1273 vol.3.

[3] M.-L. Akkar and C. Giraud, "An implementation of DES and AES, secure against some attacks," in *Int. Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2001, pp. 309–318.

[4] Y. Kumar, R. Munjal, and H. Sharma, "Comparison of Symmetric and Asymmetric Cryptography with Existing Vulnerabilities and Countermeasures," *Int. J. of Computer Science and Management Studies*, vol. 11, 10 2011.

[5] A. S. Dr. Prerna Mahajan, "A Study of Encryption Algorithms AES, DES and RSA for Security," *Global Journal of Computer Science and Technology*, 2013. [Online]. Available: https://computerresearch.org/index.php/computer/article/view/272

[6] M. Elia, M. Piva, and D. Schipani, "The Rabin cryptosystem revisited," *Applicable Algebra in Engineering, Communication and Computing*, vol. 26, no. 3, pp. 251–275, Jun 2015. [Online]. Available: https://doi.org/10.1007/s00200-014-0237-0

[7] A. Steane, "Quantum computing," *Reports on Progress in Physics*, vol. 61, no. 2, pp. 117–173, Feb 1998. [Online]. Available: https://doi.org/10.1088/0034-4885/61/2/002

[8] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, "Variational Quantum Algorithms," 2020.

[9] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, "Quantum algorithms revisited," *Proc. of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 454, no. 1969, p. 339–354, Jan 1998. [Online]. Available: http://dx.doi.org/10.1098/rspa.1998.0164

[10] L. K. Grover, "A Fast Quantum Mechanical Algorithm for Database Search," in *Proceedings of the 28th Annual ACM Symp. on Theory of Computing*, ser. STOC '96. New York, NY, USA: Association for Computing Machinery, 1996, p. 212–219. [Online]. Available: https://doi.org/10.1145/237814.237866

[11] P. W. Shor, "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer," *SIAM Journal on Computing*, vol. 26, no. 5, p. 1484–1509, Oct 1997. [Online]. Available: http://dx.doi.org/10.1137/S0097539795293172

[12] C. Burrus and P. Eschenbacher, "An in-place, in-order prime factor FFT algorithm," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 29, no. 4, pp. 806–817, 1981.

[13] B. A. Forouzan and D. Mukhopadhyay, *Cryptography and Network Security*. Mc Graw Hill Education (India) Private Limited, 2015.

[14] L. Zhou, F. B. Schneider, and R. Van Renesse, "COCA: A secure distributed online certification authority," *ACM Trans. on Computer Systems (TOCS)*, vol. 20, no. 4, pp. 329–368, 2002.

[15] W. Stallings, "Digital signature algorithms," *Cryptologia*, vol. 37, no. 4, pp. 311–327, 2013.

[16] "Learn Quantum Computation using Qiskit." [Online]. Available: https://qiskit.org/textbook

[17] R. Cramer and V. Shoup, "Design and Analysis of Practical Public-Key Encryption Schemes Secure against Adaptive Chosen Ciphertext Attack," *SIAM Journal on Computing*, vol. 33, no. 1, pp. 167–226, 2003. [Online]. Available: https://doi.org/10.1137/S0097539702403773

[18] D. Poddebniak, J. Somorovsky, S. Schinzel, M. Lochter, and P. Rösler, "Attacking Deterministic Signature Schemes Using Fault Attacks," in *2018 IEEE European Symp. on Security and Privacy (EuroS P)*, 2018, pp. 338–352.

[19] S. T. Ali, "Provable security for public key cryptosystems: how to prove that the cryptosystem is secure," in *Cryptography: Breakthroughs in Research and Practice*. IGI Global, 2020, pp. 214–238.

[20] "IBM Quantum Backends," accessed on February, 10, 2021. [Online]. Available: https://quantum-computing.ibm.com/docs/manage/backends/

[21] S. Beauregard, "Circuit for Shor's algorithm using 2n+ 3 qubits," *arXiv preprint quant-ph/0205095*, 2002.

[22] C. Gidney and M. Ekerå, "How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits," 2019.

[23] N. Gisin, G. Ribordy, W. Tittel, and H. Zbinden, "Quantum cryptography," *Rev. Mod. Phys.*, vol. 74, pp. 145–195, Mar 2002. [Online]. Available: https://link.aps.org/doi/10.1103/RevModPhys.74.145

[24] V. Scarani, H. Bechmann-Pasquinucci, N. J. Cerf, M. Dušek, N. Lütkenhaus, and M. Peev, "The security of practical quantum key distribution," *Rev. Mod. Phys.*, vol. 81, pp. 1301–1350, Sep 2009. [Online]. Available: https://link.aps.org/doi/10.1103/RevModPhys.81.1301

[25] P. W. Shor and J. Preskill, "Simple Proof of Security of the BB84 Quantum Key Distribution Protocol," *Phys. Rev. Lett.*, vol. 85, pp. 441–444, Jul 2000. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.85.441

[26] A. Mallik, "Man-in-the-middle-attack: Understanding in simple words," *Cyberspace: Jurnal Pendidikan Teknologi Informasi, volume=2, number=2, pages=109–134, year=2019*.

[27] V. Mavroeidis, K. Vishi, M. D., and A. Jøsang, "The Impact of Quantum Computing on Present Cryptography," *Int. J. of Advanced Computer Science and Applications*, vol. 9, no. 3, 2018. [Online]. Available: http://dx.doi.org/10.14569/IJACSA.2018.090354