RESEARCH ARTICLE OPEN ACCESS

Integer Factorization in RSA Encryption: Challenge for Cloud Attackers

Janaki Sivakumar [1], Hameetha Begum [2]
Department of Computing
Muscat College

ABSTRACT
Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources such as networks, servers, storage, applications, and services. Cloud computing moves application software and databases to large data centres where management of users' data and services provided to them may not be fully secured. Because data transfer is between user and provider is done remotely; data security comes into concern as it opens the door to attacks such as intrusion. RSA is a most popular security algorithm using asymmetric encryption and decryption method. This paper discusses about strength of RSA in security attacks by integer factorization. A prime number generation technique of RSA makes the middle-man in attack to failure

Keywords :— Cloud Computing, Asymmetric Encryption, Integer Factorization, Prime Number, RSA.

I. INTRODUCTION

Cloud Computing is the key driving power in many small, medium and large sized companies and as many cloud users seeks the services of cloud computing, the major concern is the security of their data in the cloud. Securing data is always of vital importance and because of the critical nature of cloud computing and the large amounts of complex data it carries, the need is even more important. This paper discusses about cloud types, Asymmetric encryption, RSA Algorithm, Integer Factorization Problem and Security in RSA.

II. CLOUD TYPES

Cloud computing involves cloud users to remotely communicate with cloud servers via internet. With the ease of such technology, data security is of major concern for the both cloud providers and cloud users [1]. Cloud computing is usually described in one of two ways, either based on the cloud location or on the service that the cloud is offering. Based on a cloud location, cloud can be classified as:

- public
- private
- hybrid
- community cloud

(a) **Public Cloud**: The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services.

(b) **Private cloud**: The cloud infrastructure is operated solely for an organization. It may be managed by the organization or a third party and may exist on premise or off premise.

(c) **Hybrid cloud**: The cloud infrastructure is a composition of two or more clouds that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting).

(d) **Community cloud**: The cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be managed by the organizations or a third party and may exist on premise or off premise.

For enterprises most important problem is security. So, concentration on USER_CLOUD security of cloud computing using encryption algorithm is much important.

A number of security threats are associated with cloud data services such as network eavesdropping, illegal invasion, and denial of service attacks. Also specific cloud computing threats, such as side channel attacks, virtualization vulnerabilities, and abuse of cloud services [2]. Confidentiality, Access Controllability, Privacy and Integrity are some of the security requirements useful to limit such threats. Encryption algorithms have been applied into the cloud computing to increase the protection. Symmetric (Classical) and Asymmetric are two types of encryption algorithms for secure data. Key distribution through public channel becomes weaker in Symmetric Encryption. So Asymmetric encryption is recommended worldwide for security of data in cloud, where Symmetric Encryption is considered for faster data.

Your paper must be in two column format with a space of 4.22mm (0.17") between columns.

III. ASYMMETRIC ENCRYPTION

Asymmetric encryption algorithm uses two keys instead of one. One is a private key only known to the recipient of the message and the other is a public key known to everyone and can be freely distributed. Either key can be used to encrypt and decrypt the message. However if only key A is used to encrypt the message then only key B can be used to decrypt it[3]. Conversely, if key B is used to encrypt the message then only key A can be used to decrypt it. Asymmetric algorithms are slower than symmetric algorithms. But it has better key distribution than symmetric algorithm. It has better scalability and also provides authenticity and non-repudiation.

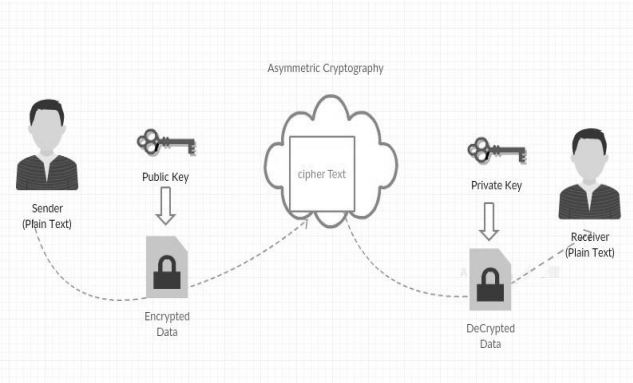

Fig. 1 Asymmetric Encryption

IV. RSA ALGORITHM

RSA was first described in 1977 by Ron Rivest, Adi Shamir and Leonard Adelman of the Massachusetts Institute of Technology. In RSA cryptography, both the public and the private keys can encrypt a message; the opposite key from the one used to encrypt a message is used to decrypt it. This attribute is one reason why RSA has become the most widely used asymmetric algorithm. It provides a method of assuring the confidentiality, integrity, authenticity and non-reputability of electronic communications and data storage [4]. The RSA is most commonly used for providing privacy and ensuring authenticity of digital data. RSA is used by many commercial systems. It is used to secure web traffic, to ensure privacy and authenticity of Email, to secure remote login sessions, and it is at the heart of electronic credit-card payment systems.

4.1. Design and use of RSA cryptosystem using example:

1. Choose two large s-bit primes p,q, s in [512,1024], and denote

$$n = pq \quad (1)$$

Where p and q are two prime numbers

$$\emptyset(n) = (p-1)(q-1) \quad (2)$$

2. Choose a large d such that

$$\gcd(d, \emptyset(n)) = 1 \quad (3)$$

3. Compute

$$e = d^{-1} (\mathrm{mod}\ \emptyset(n)) \quad (4)$$

where e is co prime to $\emptyset(n)$; $1 < e < \emptyset(n)$

4. Public Key $\rightarrow (e, n)$, Private Key $\rightarrow (d, n)$
Trapdoor information: p, q, d (decryption algorithm)

5. Plaintext :w
Encryption : crypto text $c = w^e \bmod n$
Decryption : plaintext $w = c^d \bmod n$

4.2. Sample Work:

Plaintext: KARLSRUHE

Encoding: 100017111817200704

Since 103 < n < 104, the numerical plaintext is divided into blocks of 3 digits

So 6 plaintext integers are obtained

100, 017, 111, 817, 200, 704

By choosing p = 41,q = 61 we get n = 2501, f(n) = 2400

By choosing d = 2087 we get e = 23 (or)

By choosing d = 2069 we get e=29

By choosing other values of d , would get other values of e.

Choose the first pair of encryption/decryption exponents (e=23 and d=2087).

Encryption:

$100^{23} \bmod 2501 = 2306$

$017^{23} \bmod 2501 = 1893$

$111^{23} \bmod 2501 = 621$

$817^{23} \bmod 2501 = 1380$

$200^{23} \bmod 2501 = 490$

$704^{23} \bmod 2501 = 313$

Decryption:

$2306^{2087} \bmod 2501 = 100$

$1893^{2087} \bmod 2501 = 017$

$621^{2087} \bmod 2501 = 111$

$1380^{2087} \bmod 2501 = 817$

$490^{2087} \bmod 2501 = 200$

$313^{2087} \bmod 2501 = 704$

Cipher text:

230618936211380490313

Plain text

:100017111817200704

From the above design of RSA Algorithm, gcd (p, q) is always 1 if p and q are prime. However,

gcd (x, y) can also be 1 if x and y are not prime but just don't have any divisors in common.

For example, gcd (16, 27) = 1 because there is no other number that 16 and 27 are both divisible by. So 16 and 27 are called relatively prime to each other.

V. RSA CHALLENGE

RSA inventors presented the following challenge to hackers in the year 1977[5] as Decrypt the crypto text:

9686 9613 7546 2206 1477 1409 2225 4355 8829 0575 9991 1245 7431 9874 6951 2093 0816 2982 2514 5708 3569 3147 6622 8839 8962 8013 3919 9055 1829 9451 5781 5154

The above is eencrypted using the RSA cryptosystem with 129 digit number, called also RSA129 with e = 9007.The problem was solved in 1994 by first factorizing n into one 64-bit prime and one 65-bit prime, and then computing the plaintext

The Magic Words are **SQUEMISH OSSIFRAGE**

VI. APPROACHES TO ATTACK RSA ALGORITHM

Hackers are using the below three attacking techniques to break the security wall of RSA algorithm

- Brute force key search (infeasible given size of numbers)
- Mathematical attacks (based on difficulty of computing ø(N), by factoring modulus N)
- Timing attacks (on running of decryption)

Under mathematical attacks, Security of RSA depends on the difficulty of factorizing n into its constituent p and q. With the increase in the speed of computers and the improvement of factorization methods, it becomes increasingly feasible to factorize large numbers [6]. From this perspective, choosing a longer key is essential to safeguard against attacks based on factorization. Choosing a longer key may incur higher overhead in performing encryption and decryption.

The security of e-commerce applications based on public key cryptography such as RSA depends on the difficulty of factorizing large integers. Given the progress in the development of new factorization methods, the increase in the computational power of personal computers, and the emergence of well-organized group of users such as distributed.net, organization should deploy public key cryptography with key length long enough to make the factorization attack difficult[7]

For example, multiplying

17477852958781876547 × 15241555427044345769, probably need only a matter of minutes with pencil and paper, or a tiny fraction of a second with a computer, to answer as

17477852958781876547 × 15241555427044345769 =

266389664617004986624
09797818773977964 3

What if we turn the question around and ask what two integers would have to be multiplied together to yield a given number?

For example,

? × ? =
266389664617004986624097978187739779643

Suddenly, this task is remarkably difficult; you'd have little hope of ever getting the answer by guessing with pencil and paper, and even with a computer you might have to write a new program to find the answer, and it might take a noticeable amount of time to do so [8]. It seems as though this pattern continues, and factoring gets drastically harder as the numbers involved get larger.

VII. SECURITY OF RSA

Security of RSA relies on the computational difficulty of factoring large integers. As computing power increases and more efficient factoring algorithms are discovered, the ability to factor larger and larger numbers also increases. Encryption strength is directly tied to key size, and doubling key length delivers an exponential increase in strength, although it does impair performance. RSA keys are typically 1024- or 2048-bits long, but experts believe that 1024-bit keys could be broken in the near future, which is why government and industry are moving to a minimum key length of 2048-bits.

Barring an unforeseen breakthrough in quantum computing, it should be many years before longer keys are required, Finally, a team of researchers which included Adi Shamir, a co-inventor of RSA, has successfully determined a 4096-bit RSA key using acoustic cryptanalysis, however any encryption algorithm is vulnerable to this type of attack.

Cryptographic algorithms provide various "strengths" of security, depending on the algorithm and the key size used. In this discussion, the algorithms are considered and compared the strength for the given key sizes. If the amount of task needed to "smash the algorithms" or establish the keys is approximately the same using a given resource. The security strength of an algorithm for a given key size is conventionally described in terms of the quantity of work it takes to try all keys for a symmetric algorithm with a key size of "X" that has no short cut attacks (i.e., the most efficient attack is to try all possible keys). In this case, the best attack is said to be the fatigue attack. An algorithm that has a Y-bit key, but whose strength is equivalent to an X-bit key of such a symmetric algorithm is said have a "security strength of X bits" or to provide "X bits of security" based on Integer Factorization Cryptography [9].

In common there are four security levels to secure the data in cloud computing. Each level and the key length are mentioned in the below Table 1.

TABLE I
SECURITY LEVEL VS. KEY BITS

Level	Key size(in Bits)
Low	512
Medium	1024
High	2048
High Security	4096

The security of RSA is analysed by varying the private key length in bits and RSA Modulus size in Table 2.

TABLE II
KEY LENGTH VS. RSA MODULUS

Private Key Length(in Bits)	RSA Modulus Size
Low	512
Medium	1024
High	2048
High Security	4096

VIII. CONCLUSION

Cloud Computing is still a new and evolving paradigm where computing is regarded as on-demand service. Once the organization takes the decision to move to the cloud, it loses control over the data. Thus, the amount of protection needed to secure data is directly proportional to the value of the data. Security of the Cloud relies on trusted computing and cryptography. Thus, by using RSA algorithm, only the authorized users can have access rights to the data. If Hackers captures the data also, he can't decrypt it and get back the plain data from it. RSA algorithm is such strong in Integer factorization with different Key Size even if eve droppers get access to public key, Trapdoor(p,q) information cannot be obtained .By analysing the above discussion, we have concluded that RSA encryption algorithm is a feasible solution for secure communication in cloud computing.

REFERENCES

[1] S.Hemalatha, Dr.R.Manickachezian, "Present and Future of Cloud Computing: A Collaborated Survey Report"., International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-1, Issue-2, July 2012.

[2] Puneet Jai Kaur, Sakshi Kaushal, "Security Concerns in Cloud Computing", Communication in Computer and Information Science Volume 169, pp.103-112, 2011.

[3] Mandeep Kaur and Manish Mahajan, "Implementing Various Encryption Algorithms To Enhance The Data Security Of Cloud In Cloud Computing" VSRD International Journal of Computer Science & Information Technology, Vol. 2, pp.831-835, October 2012.

[4] Khan, Miss Shakeeba S., and Miss Sakshi S. Deshmukh. "Security in cloud computing using cryptographic algorithms." IJCA, 2014.

[5] Kevin Curran, Sean Carlin and Mervyn Adams, "Security issues in cloud computing", Elixir Network Engg.38 (2011), pp.4069-4072, August 2011.

[6] Randeep Kaur, Supriya Kinger, "Analysis of Security Algorithms in Cloud Computing" International Journal of Application or Innovation in Engineering & Management (ISSN 2319 - 4847),Volume 3 Issue 3, pp.171-176, March 2014.

[7] C Rachana, Dr. H S Guruprasad, "Emerging Security Challenges in Cloud Computing ", International Journal of Engineering Science and Innovative Technology (IJESIT), Volume 3 Issue 2, pp.485-490, March 2014.

[8] Shamir, "How to share a secret," Communications of the ACM, vol. 22, pp. 612–613, 1979.

[9] Hemalatha, S., and R. Manickachezian. "Security Strength of RSA and Attribute Based Encryption for Data Security in Cloud Computing." Int. J. Innov. Res. Comput. Commun. Eng 2.9 (2014): 5847-5852.