© 2014 IJIRT | Volume 1 Issue 6 | ISSN : 2349-6002

Enhancing the Security and Reliability of the Data over Computer Networks using RSA cryptosystem

Pavan Kumar Pagadala[1], Jasmine Sabeena[2]
[1]Department of IT, S V College of Engineering, Tirupathi, Andhra Pradesh, India
[2]Department of CSE, S V College of Engineering ,Tirupathi, Andhra Pradesh, India

***Abstract*- One of the principal challenges of resource sharing on data communication network is its security. This paper presents a design of data encryption and decryption using RSA algorithm with a specific message block size. RSA is based on several mathematical principles in number theory. This paper proposed an implementation of a complete RSA encrypt/decrypt solution based on the study of RSA public key cryptosystem. A cryptosystem is simply an algorithm that can convert input data into something unrecognizable (encryption), and convert the unrecognizable data back to its original form (decryption), security depends on the algorithm while the internal structure of the rigor of mathematics. In RSA algorithm what kind of data you choose to be a key, how to distribute the private key, and how to save both data transmission keys are very important issues in the encryption and decryption algorithm.**

***Index Terms*- Network, Data Security, RSA algorithm, Encryption, Decryption**

I. INTRODUCTION

It is our goal to protect the network and the data that is transmitted over it. Every organization should be concerned about protecting data against intruders, for the organization's ability to survive depends on the availability, comprehensiveness and reliability of its financial and organizational data. Security has become more complicated with the expanded use and networking of personal computers. At present, the local networks and the connections between the large and small computers are such that each of them takes part in the application. The application as a whole appears to be located on the user's computer, but in fact each user and each application has access to, and sometimes even control over, organizational data on various computers and storage facilities.

Unfortunately, many companies do not deal with data security and network management problems until there is a crack in the network. To protect vital information; the companies must set up a sound security system before the network is intruded. This involves identification of the security risks, applying sufficient means of security, and teaching the users data security awareness.

1.1 DATA SECURITY:

In simple terms, the Data security is a practice of keeping data protected from unauthorized access & corruption. The focus behind data security is to ensure privacy while protecting personal or corporate data. Security system components in Distributed computer systems. Distributed computer systems pose four main securities components: security authentication, authorization, access control and encryption.

Authentication – Usually authentication is realized by a "smart token" which is a hardware device in the size of a pocket computer or credit card that creates a password and transfers it to the authentication server that is linked up to the network.

Authorization - The aim here is to supply one secured access point enabling the users to link up to the network once and allow them access to authorized resources.

The authorization is examined via software servers enabling the client, acting in the name of the user, to prove his identity to the authentication server, without sending information over the network that would reveal the client or the party rendering the service.

Encryption - Implemented using intricate algorithms such as RSA, PGP, DES based on the use of public and private key systems (Fleeter, 1997).

Access control - Implemented via access matrices, access lists, capabilities list. These lists define access authorization to the computer resources for the user.

1.2 Basic Concepts:

These security concepts important to information on the internet are confidentiality, integrity, and availability.

Concepts relating to the people who use that information are authentication, authorization, and non-repudiation. When information is read or copied by someone not authorized to do so, the result is known as loss of confidentiality. For some types of information, confidentiality is a very important attribute. Examples include research data, medical and insurance records, new product specifications, and corporate investment strategies. In some locations, there may be a legal obligation to protect the privacy of individuals. This is particularly true for banks and loan companies; debt collectors; businesses that extend credit to their customers or issue credit cards; hospitals, doctors' offices, and medical testing laboratories; individuals or agencies that offer services such as psychological counseling or drug treatment; and agencies that collect taxes.

Information can be corrupted when it is available on an insecure network. When information is modified in unexpected ways, the result is known as loss of integrity. This means that unauthorized changes are made to information, whether by human error or intentional tampering. Integrity is particularly important for critical safety and financial data used for activities such as electronic funds transfers, air traffic control, and financial accounting.

Information can be erased or become inaccessible, resulting in loss of availability. This means that people who are authorized to get information cannot get what they need. Availability is often the most important attribute in service-oriented businesses that depend on information. Availability of the network itself is important to anyone whose business or education relies on a network connection. When users cannot access the network or specific services provided on the network, they experience a denial of service.

1.3 Types of Attacks

Classes of attack might include passive monitoring of communications, active network attacks, close-in attacks, exploitation by insiders, and attacks through the service provider. Information systems and networks offer attractive targets and should be resistant to attack from the full range of threat agents, from hackers to nation-states. A system must be able to limit damage and recover rapidly when attacks occur. There are five types of attack:

Passive Attack

A passive attack monitors unencrypted traffic and looks for clear-text passwords and sensitive information that can be used in other types of attacks. Passive attacks include traffic analysis, monitoring of unprotected communications, decrypting weakly encrypted traffic, and capturing authentication information such as passwords.

Passive interception of network operations enables adversaries to see upcoming actions. Passive attacks result in the disclosure of information or data files to an attacker without the consent or knowledge of the user.

Active Attack

In an **active attack,** the attacker tries to bypass or break into secured systems. This can be done through stealth, viruses, worms, or Trojan horses. Active attacks include attempts to circumvent or break protection features, to introduce malicious code, and to steal or modify information. These attacks are mounted against a network backbone, exploit information in transit, electronically penetrate an enclave, or attack an authorized remote user during an attempt to connect to an enclave. Active attacks result in the disclosure or dissemination of data files, DoS, or modification of data.

Distributed Attack

A distributed attack requires that the adversary introduce code, such as a Trojan horse or back-door program, to a "trusted" component or software that will later be distributed to many other companies and users Distribution attacks focus on the malicious modification of hardware or software at the factory or during distribution. These attacks introduce malicious code such as a back door to a product to gain unauthorized access to information or to a system function at a later date.

Insider Attack

An insider attack involves someone from the inside, such as a disgruntled employee, attacking the network Insider attacks can be malicious or no malicious. Malicious insiders intentionally eavesdrop, steal, or damage information; use information in a fraudulent manner; or deny access to other authorized users. No malicious attacks typically result from carelessness, lack of knowledge, or intentional circumvention of security for such reasons as performing a task

Phishing Attack

In phishing attack the hacker creates a fake web site that looks exactly like a popular site such as the SBI bank or paypal. The phishing part of the attack is that the hacker then sends an e-mail message trying to trick the user into clicking a link that leads to the fake site. When the user attempts to log on with their account information, the hacker records the

username and password and then tries that information on the real site.

Hijack attack

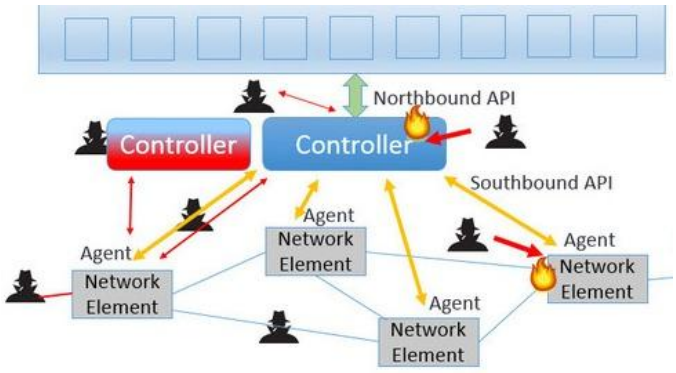
In a hijack attack, a hacker takes over a session between you and another individual and disconnects the other individual from the communication. You still believe that you are talking to the original party and may send private information to the hacker by accident.

Spoof attack

Spoof attack In a spoof attack, the hacker modifies the source address of the packets he or she is sending so that they appear to be coming from someone else. This may be an attempt to bypass your firewall rules.

Exploit attack

Exploit attack In this type of attack, the attacker knows of a security problem within an operating system or a piece of software and leverages that knowledge by exploiting the vulnerability.



Password attack

Password attack An attacker tries to crack the passwords stored in a network account database or a password-protected file. There are three major types of password attacks: a dictionary attack, a brute-force attack, and a hybrid attack. A dictionary attack uses a word list file, which is a list of potential passwords. A brute-force attack is when the attacker tries every possible combination of characters.

II. METHODOLOGY

There are many ways of classifying data cryptographic algorithms but for the purpose of this paper, they will be classified based on the number of keys that are employed for encryption and decryption. The three common types of algorithms are:

a. Secret Key Cryptography (SKC):

The SKC method uses only a single key for both encryption and decryption. The schemes are generally categorized as being either stream ciphers or block ciphers. Stream ciphers operate on a single bit (byte or computer word) at a time and implement some form of feedback mechanism so that the key is constantly changing while block cipher scheme encrypts one block of data at a time using the same key on each block.

The main drawback of this method is propagation error because a distorted bit in transmission will result in n distorted bits at the receiving side. Though stream ciphers do not propagate transmission errors, they are periodic therefore the key-stream will eventually repeat. This normally results in the use of digital signature mechanisms with either large keys for the public verification function or the use of a TTP.

b. Public Key Cryptography (PKC):

PKC scheme uses one key for encryption and a different key for decryption. Modern PKC was first described using a two-key crypto system in which two parties could engage in a secure communication over a non-secure communications channel without having to share a secret key [5]. In PKC, one of the keys is designated the public key and may be advertised as widely as the owner wants. The other key is designated the private key and is never revealed to another party. RSA is one of the first and still most common PKC implementation that is in use today for key exchange or digital signatures.

The cardinal advantage of this method is that administration of keys on a network requires the presence of only a functionally trusted TTP, as opposed to an unconditionally trusted TTP. Depending on the mode of usage, the TTP might only be required in an “off-line” manner, as opposed to in real time. Many public-key schemes yield relatively efficient signature mechanisms. The key used to describe the public verification function is typically much smaller than for the symmetric-key counterpart.

c. Hash Functions (HF):

The HF uses a mathematical transformation to irreversibly "encrypt" information. This algorithm does not use keys for encryption and decryption of data. It rather uses a fixed-length hash value which computed based on a plaintext that makes it impossible for either the contents or length of the plaintext to be recovered. These algorithms are typically used

to provide a digital fingerprint of a file's content, often used to ensure that the file has not been altered by an intruder or virus. Hash functions are also commonly employed by many operating systems to encrypt passwords to provide some measure of the integrity of a file.

III. Proposed Work

RSA is widely used Public-Key algorithm. RSA stands for Ron Rivest, Adi Shamir and Len Adleman, who first publicly described it in 1977. In our proposed work, we are using RSA algorithm to encrypt the data to provide security so that only the concerned user can access it. By securing the data, we are not allowing unauthorized access to it. User data is encrypted first and then it is stored in the buffers. When required, user places a request for the data for the service provider; service provider authenticates the user and delivers the data. RSA is a block cipher, in which every message is mapped to an integer. RSA consists of Public-Key and Private-Key. In our networks environment, Pubic-Key is known to all, whereas Private-Key is known only to the user who originally owns the data. Thus, encryption is done by the service provider and decryption is done by the user or consumer. Once the data is encrypted with the Public-Key, it can be decrypted with the corresponding Private-Key only.
RSA algorithm involves three steps:
1. Key Generation
2. Encryption
3. Decryption

Key Generation:

Before the data is encrypted, Key generation should be done. This process is done between the service provider and the user.

Steps:

1. Choose two distinct prime numbers a and b. For security purposes, the integers a and b should be chosen at random and should be of similar bit length.

2. Compute n = a * b.

3. Compute Euler's totient function,
Ø(n) = (a-1) * (b-1).

4. Chose an integer e, such that 1 < e < Ø(n) and greatest common divisor of e , Ø(n) is 1. Now e is released as Public-Key exponent.

5. Now determine d as follows: d = e-1(mod Ø(n)) i.e., d is multiplicate inverse of e mod Ø(n).

6. d is kept as Private-Key component,
so that d * e = 1 mod Ø(n).

7. The Public-Key consists of modulus n and the public exponent e i.e, (e, n).

8. The Private-Key consists of modulus n and the private exponent d, which must be kept secret i.e, (d, n).

Encryption:

Encryption is the process of converting original plain text (data) into cipher text (data).

Steps:

1. Network service provider should give or transmit the Public- Key (n, e) to the user who wants to store the data with him or her.

2. User data is now mapped to an integer by using an agreed upon reversible protocol, known as padding scheme.

3. Data is encrypted and the resultant cipher text(data) C is C = me (mod n).

4. This cipher text or encrypted data is now stored with the network service provider.

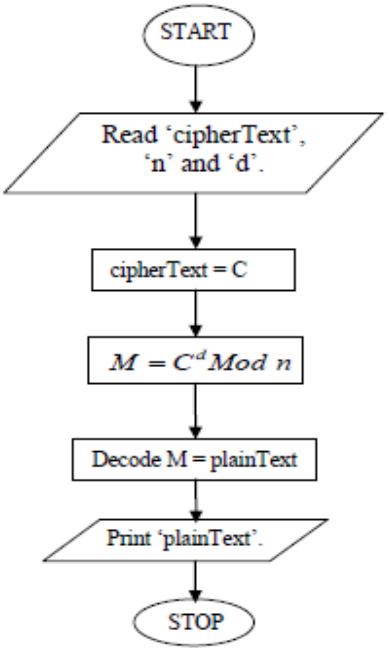


Figure: Flowchart the process of Encryption

Decryption:

Decryption is the process of converting the cipher text (data) to the original plain text (data).

Steps:

1. The network user requests the network service provider for the data.

2. Network service provider verity's the authenticity of the user and gives the encrypted data i.e., C.

3. The network user then decrypts the data by computing, m = Cd (mod n).

4. Once m is obtained, the user can get back the original data by reversing the padding scheme.

Now RSA security enhanced measures have been proposed to improve the security of data transmission over computer networks. Now easily we can transfer our data with high efficiency and reliability using the above algorithm.

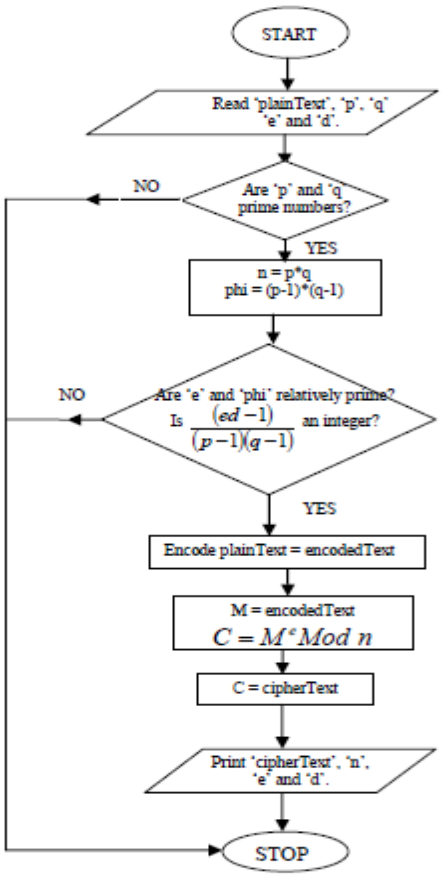


Figure: Flowchart the process of Decryption

3.1 The RSA Algorithm for Creating RSA Public and Private Key Pair

RSA makes use of an expression with exponentials. Plaintext is encrypted in blocks, with each block having a binary value less than some integer n. That is, the block size must be less or equal to $\log_2 n$; in practice, the block size is k bits, where $2^k < n \leq 2^{k+1}$. Encryption and decryption are of the following form, for some plaintext block M and cipher text block C:

$$C = M^e \bmod n$$

$$M = C^d \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n$$

Both sender and receiver must know the value of n. The sender knows the value of e, and only receiver knows the value of d. Thus, this is a public-key encryption algorithm with a public key of KU={e,n}, and a private key of KR={d,n}. For this algorithm to be satisfactory for public-key encryption, the following requirements must be met:

It is possible to find values of e,d,n such that $M^{ed} = M \bmod n$ for all M<n

It is relatively easy to calculate M^e and C^d for all values of M<n

It is infeasible to determine d given e and d.

For now, we focus on the 1st requirement and consider the other questions later. We need to find a relationship of the form

$$M^{ed} = M \bmod n$$

A corollary to Euler's theorem

(For every a and n that are relatively prime

$$a^{\varphi(n)} \equiv 1 \bmod n$$

Where $\varphi(n)$ is the Euler's totient function – number of positive integers less than n and relatively prime to n), fits the bill:

Given two prime numbers, p and q, and two integers, n and m, such that n=pq and 0<m<n, and arbitrary integer k, the following relationship holds:

$$m^{k\varphi(n)+1} = m^{k(p-1)(q-1)+1} \equiv m \bmod n \quad (*)$$

(as far as for p,q prime, $\varphi(n) = (p-1)(q-1)$)

Thus, we can achieve the desired relationship if $ed = k\varphi(n) + 1$

$$ed \equiv 1 \bmod \varphi(n)$$

This is equivalent to saying: $d \equiv e^{-1} \bmod \varphi(n)$

That is, e and d are multiplicative inverses $\bmod \varphi(n)$. Note that, according to the rules of modular arithmetic, this is true only if d (and therefore e) is relatively prime to $\varphi(n)$. Equivalently, $\gcd(\varphi(n), d) = 1$.

We are now ready to state the RSA scheme. The ingredients are the following:

p,q, two prime numbers (private, chosen)

n=pq (public, calculated)

e, with $\gcd(\varphi(n), e) = 1; 1 < e < \varphi(n)$ (public, chosen)

$d \equiv e^{-1} \bmod \varphi(n)$ (private, calculated)

The private key consists of {d,n}, and the public key consists of {e,n}. Suppose that user A has published its public key and that user B wishes to send message M to A. Then B calculates $C = M^e \bmod n$ and transmits C. On receipt of this ciphertext, user A decrypts by calculating $M = C^d \bmod n$. It is worthwhile to summarize the justification for this algorithm. We have chosen e and d such that $d \equiv e^{-1} \bmod \varphi(n)$ .Therefore $ed \equiv 1 \bmod \varphi(n)$.

Therefore, ed is of the form $k\varphi(n) + 1$. But by the corollary to Euler's theorem (*), given two prime numbers, p and q, and integer n=pq and M, with 0<M<n:

$$M^{k\varphi(n)+1} = M^{k(p-1)(q-1)+1} \equiv M \bmod n$$

So, $M^{ed} \equiv M \bmod n$. Now

$$C = M^e \bmod n$$

$$M = C^d \bmod n \equiv (M^e)^d \bmod n \equiv M^{ed} \bmod n \equiv M \bmod n$$

3.2 EXPERIMENTAL RESULTS

In this section, we are taking some sample data end implementing RSA algorithm over it.

Example 1:

The keys were generated as follows:
Choose p = 3 and q = 11

Compute n = p * q = 3 * 11 = 33
Compute φ(n) = (p - 1) * (q - 1) = 2 * 10 = 20
Choose e such that 1 < e < φ(n) and e and n are coprime. Let e = 7
Compute a value for d such that (d * e) % φ(n) = 1. One solution is d = 3 [(3 * 7) % 20 = 1]
Public key is (e, n) => (7, 33)
Private key is (d, n) => (3, 33)

The encryption of m = 2 is c = 27 % 33 = 29
The decryption of c = 29 is m = 293 % 33 = 2

Assume X and Y interpret each letter in the English alphabet as a number between 1 and 26. That is, a = 1, b = 2, …, n = 14,…, z = 26. Assume person X wishes to send the plaintext "love" to the person Y, after encrypting using RSA.

Example 2:

The keys were generated as follows:
Compute person Y's public key using RSA.
Determine person X's cipher text using RSA algorithm.
Example (with small values for p and q):
Suppose person Y chooses p = 5, and q = 7.

Compute n = pq =35, and z = (p-1)(q-1) = (4)(6) = 24
Person Y chooses e = 5; Note 5 and 24 have no common factors.

Person Y chooses d = 29; Note (ed-1) = (5x29 -1) = (145-1) = 144 is exactly divisible by z =24 (as 24 x 6 = 144)

Person Y's public key is given by KB+ = (n, e) = (35, 5);
Person Y's private key is given by KB- = (n, d) = (35, 29)

Suppose X and Y interpret each letter as a number between 1 and 26.
That is, a = 1, b = 2, c = 3,…, n = 14,…, z = 26.
Assume plain text is: Love

Person X's RSA encryption with e = 5, and n =35 is as follows:

Plaintext letter	m (numeric representation)	m e	c = me mod n
l	12	248832	17
o	15	759375	15
v	22	5153632	22
e	5	3125	10

Cipher text is: 17152210
Y's decryption with d = 29, and n = 35 is as follows:

Cipher text	cd	m = cd mod n	Plaintext letter
17	48196857210675 09150914118252 23071697	12	l
15	12783403948858 93911123275756 8359375	15	o
22	85164331908653 77019561944997 21106030592	22	v
10	10000000000000 00000000000000 00	5	e

3.3 Why does RSA work?

The result from number theory that states the following:

If p and q are prime, and n = pq, then xy mod n = x(ymod(p-1)(q-1)) mod n

This rule implies that (me)d mod n = m because, as per number theory rule

(me)d mod n = m (ed(mod(p-1)(q-1)) mod n = m1 mod n (as the integer remainder when ed is divided by z is 1)= m.

Similarly, (md)e mod n = m.

Providing more security to RSA with the help of biggest primes

Example 3:

The keys were generated as follows:
p = 31 q = 23 (chosen at random)

n = 31*23 = 713
r = 30*22 = 660
e = 223 (chosen at random)
d=367(computed using Euclids algorithm as below)

660 = 223 * 2 + 214 223 = 214 * 1 +9 214 = 9 * 23 +7 9 = 7 * 1 + 2 7 = 2 * 3 +1	1 = 7 – 3(2) 1 = 7 – 3(9-7) = 4(7) – 3(9) 1 = 4(214-9(23))-3(9) = 4(214) – 95(9) 1 = 4(214) – 95(223 – 214) = 99(214) – 95(223) 1 = 99(660 – 2(223))-95(223) 1 = 99(660)-293(223) so the inverse of 223 mod 660 = -293 = 367

Y's private key = 367 Y's public key = (223, 713)

Encryption

X wishes to send Y the message m = 439.
He computes c=439223 mod 713 = 284 (see fast exponent ion table below)
X sends the ciphertext 284 to Y.

Decryption

Y receives the ciphertext 284 from X.
He computes 284367 mod 713 = 439 (see fast exponentiation table below)
He knows the message is 439

439^{223} mod 713		
y	u	n
1	439	223

284^{367} mod 713		
y	u	n
1	284	367

439	211	111	284	87	183
652	315	55	466	439	91
36	118	27	656	211	45
683	377	13	94	315	22
98	242	6	94	118	11
98	98	3	397	377	5
335	335	1	652	242	2
284		0	652	98	1
			439		0

IV. CONCLUSION

Network security is still a new and evolving paradigm where security is regarded as on-demand service. Once the organization takes the decision to move, 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 Network relies on trusted Network and cryptography.

Thus, in our proposed work, only the authorized user can access the data. Even if some intruder (unauthorized user) gets the data accidentally or intentionally if he captures the data also, he can't decrypt it and get back the original data from it. Hence forth, data security is provided by implementing RSA algorithm.

REFERENCES

[1] Amoroso, E. (1994), Fundamentals of Computer Security Technology, chap. 7, Prentice-Hall, Englewood Cliffs, NJ.

[2] Appleton, K., & Elain, L. (1997), Network Security: Is Your LAN Safe? DATAMATION, 39, pp. 45-49.

[3] Hellman, M. and J. Diffie, 1976. New Directions in Cryptography. IEEE transactions on Information theory, vol. IT-22, pp:644-654.

[4] Shinde, G.N. and H.S. Fade War, 2008. Faster RSA algorithm for decryption using Chinese remainder theorem. ICCES, Vol. 5, No. 4.

[5] Yang L. and S.H. Yang. 2007. A frame work of security and safety checking for internet-based control systems. International Journal of Information and Computer security.

[6] R. Rivest, A. Shamir, L. Adleman, \A method for obtaining digital signatures and public-key cryptosystems"

[7] The RSA Algorithm Evgeny Milanov 3 June 2009.