Key Distributed Cryptography using Key Algorithm in MANET

Rajesh Kumar Dangi, Rachna Singh Thakur, Neha Rahinj, Sankalp Rajora & Dinesh Thakur
Malwa Institute of Technology, Indore India
E-mail: rajesh.dangi127@gmail.com, rachnasingh.30nov@gmail.com, nrahinj22@gmail.com, Sankalprajora2006@gmail.com, d9229116722@gmail.com

Abstract – Mobile Ad hoc Network (MANET for short) is a kind of communication network which is different from the traditional wireless network due to its characteristic of no central administration. It is a self-configurable and autonomous network, and consists of several independent nodes. MANET in recent years not only seen widespread use in domestic and commercial application areas but also becomes the focus of intensive research. Security aspects play an important role in almost all of these application scenarios given the vulnerabilities inherent in wireless ad hoc networking.

In this paper, we study the security goals and solutions to be achieved and also use cryptographic schemes such as distributed cryptography, to build a highly available and secure key management service and also provide accurate routing information in a timely manner. The efficiency of a distributed cryptosystem strongly depends on the characteristics of the public key cryptosystem on which it is based.

We propose an optimized encryption method which may be related with the RSA key generation and Diffie Hellman key exchange mechanism, to provide more security for data exchange. In this we perform splitting both, plaintext before encryption and ciphertext after encryption and also here we try to provide more security by combining both algorithms.

Keywords – MANET, Distributed Cryptography, Data Transmission, RSA and Diffie Hellman algorithm.

I. INTRODUCTION

Mobile Ad-Hoc Networks – MANET, an English translation of ad-hoc is “For a particular purpose and a MANET consists of mobile platforms (e.g., a router with multiple hosts and wireless communications devices)--herein simply referred to as 'nodes'--which are free to move about arbitrarily

Cryptography is the art and science of keeping messages secure. Basically, the point of cryptography is to allow any user to keep his data secure and not readable from not desired individuals. Distributed Cryptography is the method of key distribution or distribution of trust to among nodes. Distributed Cryptography ensure the availability of nodes, even if some of the information is lost still the actual message reaches the intended receiver without compromising the security. Distributed Cryptography can be implemented using RSA, Elliptic Curve Cryptography (ECC) and Digital signature. Distributed cryptography involves sharing of a key by multiple individuals called shareholders engaged in encryption or decryption [2].

Distributed cryptography spreads the operation of a cryptosystem among a group of servers (parties) in a fault-tolerant way. We consider the threshold failure model with n servers, of which up to t are faulty; such distributed cryptosystems are called threshold cryptosystems. Distributed cryptosystems are based on secret sharing and are typically known only for public-key cryptosystems because of their “nice” algebraic properties. Here we consider a public-key cryptosystem and a digital signature scheme.

II. OBJECTIVES

Communicating nodes in a MANET usually seek the help of other intermediate nodes to establish communication channels. In such an environment, malicious intermediate nodes can be a threat to the security of conversation between mobile nodes. The security experience from the Wired Network world is of little use in Wireless Mobile Ad hoc networks, due to some basic differences between the two Networks. Therefore, some novel solutions are required to make MANET secure. Some Solutions are:

• Secure Routing.
• Dynamic Source Routing.
• Adhoc On-demand Distance Vector Routing.
• Threshold/ Distributed Cryptography.
• Proactive Security.
Communication over the network using Cryptography Mechanism for providing security. Our main aim is to implement the Distributed Cryptography in the Distributed network using security algorithms. The significance of our project is to study the effectiveness of various network security algorithms.

1) Study of various cryptography algorithms.
2) How to provide key shares to the servers through Cryptography mechanism.
3) Evaluating the trustworthy node.
4) Preparing comparison chart of various algorithm.
5) Finding new cryptography algorithm with better security if possible.

III. METHODOLOGY

Methodology defines about data collection of the project work. The data collection as such is divided into

i. Primary data collection: Data collected through questionnaire survey, brainstorming, opinion poll, discussion etc.

ii. Secondary data collection: Data collected through literature survey, journals, Internet search, company records/bulletin, company reports, CD-ROM search etc.

Research will be more of descriptive research. The most important purpose of the research is to bring out facts that will prove that it can be used to improve the services in many ways. In methodology we divide our project in two phases.

In the first phase we analyze various papers of cryptography security algorithms on the basis of time complexity, efficiency, data integrity, consistency, security, confidentiality etc. In second phase firstly on the basis of our previous phase analysis we are proposing a new algorithm on the basis of their different functional characteristics like tim complexity, efficiency, data integrity, consistency, security, confidentiality etc of RSA and Diffie.

A Collect and Examine information:

The initial step is collect all possible information about the cryptography security algorithm and then we analyze various papers of cryptography security algorithms on the basis of time complexity, efficiency, data integrity, consistency, security, confidentiality etc. We are more concerned with analyzing various asymmetric algorithms like RSA, Diffie-Hellman etc.s. Review the collected information. This step allows the person(s) doing the recovery to become familiar with the system and its components. This step permitted a plan to be created for dissecting the program and entering its information into the development environment.

1) Extract the structure:

The second step was to studying asymmetric algorithms among them we implement two security algorithms RSA and Diffie-Hellman on the basis of their mechanism (formula). For implementing both of these algorithms we use the java technology. By studying both the algorithms we find out that yet the RSA and Diffie-Hellman algorithm is used for single point to point security (encryption/decryption) where there is only one sender and one receiver. So here our approach is to implement it in distributed environment. So we come to a new solution that is a combination of both RSA and DIFFIE is proposed in this paper.

2) Record Accuracy, Integrity, Efficiency of both RSA and DIFFIE HELLMAN:

Here we record the accuracy, integrity, efficiency of both the algorithms(RSA and DIFFIE HELLMAN), to analyze various shortcomings that are present in that algorithm and to provide a better proposed solution. We found about RSA that if it compromises its private key then whole message will be recovered, and also it is slower than DES(DATA ENCRYPTION STANDARD) and other symmetric key algorithm.

3) Record Accuracy, Integrity, Efficiency of our proposed algorithm:

Here we analyze our new proposed algorithm in terms of Accuracy, Integrity, Efficiency etc., and then compared it with RSA and we found that our algorithm is more secure than RSA because in our algorithm encryption and decryption is done with using more secure cipher text generated after XOR between cipher text and Difie key.

4) Identify Class and class dependencies:

Recognize all the classes that are present in the source code along with the dependencies between various classes as well.

5) Generate documentation:

The final step is to generate design documentation. Information explaining the purpose of the program, program-overview, history, etc, will need to be recorded. This information will almost probably not be contained in the source code and must be recovered from other sources.
IV. RSA ALGORITHM

A. What is RSA?

**Rivest-Shamir-Adleman (RSA).** A public key encryption algorithm developed by Ronald Rivest, Adi Shamir, and Leonard Adleman in 1978 that became a de facto standard. RSA formed the basis for a number of encryption programs, including Pretty Good Privacy (PGP). RSA is an algorithm for public key encryption. It was the first algorithm known to be suitable for signing as well as encryption, and one of the first great advances in public key encryption. It is still widely used in electronic commerce protocols, and is believed to be secure given sufficiently long keys. The RSA scheme is a block cipher. Each plaintext block is an integer between 0 and n−1 for some n, which leads to a block size \( \leq \log_2 (n) \). The typical block size for RSA is 1024 bits [5].

1) World’s most popular Asymmetric Key Encryption algorithm
2) Based on the theory of Prime Numbers
3) Algorithm is based on the fact that it is easy to find and multiply large prime numbers together, but it is extremely difficult to factor their product.
4) The private and public keys in RSA are based on very large (100 or more digits) prime numbers.
5) Real challenge in the case of RSA is the selection and generation of public and private keys.

B. Steps of this algorithm are as:

1) Choose two large prime numbers \( P \) and \( Q \).
2) Calculate \( N = P \times Q \).
3) Select the public key (i.e. encryption key) \( E \) such that it is not a factor of \((P−1)\) and \((Q−1)\).
4) Select the private key (i.e. the decryption key) \( D \) such that the following equation is true: \((D \times E) \mod (P−1) = 1\).
5) For encryption, calculate the cipher text \( CT \) from the plain text \( PT \) as follows: \( CT = PT^E \mod N \).
6) Then send \( CT \) as the cipher text to the receiver.
7) For decryption, calculate the plain text \( PT \) from the cipher text \( CT \) as follows: \( PT = CT^D \mod N \).

C. Limitations of RSA are as:

1) Every RSA initialization process requires the random selection of two very large prime numbers \( p \) and \( q \).
2) In the real world the encryption capabilities of RSA are rarely used for one simple reason: the length of plain text that can be encrypted is limited to the size of \( n = p^aq \).
3) RSA is much slower than DES and other symmetric cryptosystems.

4) If any one of \( p, q, e, d \) is known, then the other values can be calculated. So secrecy is important.
5) To protect the encryption, the minimum number of bits in \( n \) should be 1024.

![Fig.1 : RSA Symmetry key Algorithm](image)

V. DIFFIE HELLMAN KEY EXCHANGE ALGORITHM

A. What is Diffie Hellman key Exchange?

Whitfield Diffie and Martin Hellman discovered what is now known as the Diffie-Hellman (DH) algorithm in 1976. It is an amazing and ubiquitous algorithm found in many secure Connectivity protocols on the Internet. In an era when the lifetime of “old” technology can sometimes be measured in months, this algorithm is now celebrating its 25th anniversary while it is still playing an active role in important Internet protocols.

DH is a method for securely exchanging a shared secret between two parties, in real-time, over an untrusted network. A shared secret is important between two parties who may not have ever communicated previously, so that they can encrypt their communications. As such, it is used by several protocols, including Secure Sockets Layer (SSL), Secure Shell (SSH), and Internet Protocol Security (IPSec). These protocols will be discussed in terms of the technical use of the DH algorithm and the status of the protocol standards established or still being defined [6].

B. Steps of this algorithm are as:

1) Taking two numbers \( “P” \) and \( “G” \), \( “P” \) is a large prime number \( “G” \) is called the base or generator.
2) Picks a secret number \( “A” \) as first secret number = \( A \), then picks another secret number \( “B” \) as second secret number = \( B \).
3) Computes first public number \( X = G^A \mod P \), and public number = \( X \). Then computes second public number \( Y = G^B \mod P \), and public number = \( Y \).
4) Exchange their public numbers.
5) First knows \( P, G, A, X, Y \), Second knows \( P, G, B, X, Y \).
6) Computes First session key as $K_A = Y^a \mod P$ OR $K_A = (G \mod P)^a \mod P$ OR $K_A = (G^b)^a \mod P$ OR $K_A = G^{ab} \mod P$.

7) Computes second session key as $K_B = X^b \mod P$ OR $K_B = (G^a \mod P)^b \mod P$ OR $K_B = (G^b)^b \mod P$ OR $K_B = G^{ab} \mod P$.

i. Fortunately for Both by the laws of algebra, First session key “$K_A$” is the same as Second session key “$K_B$”, or $K_A = K_B = K$.

ii. Know we have both the secret value as “$K$”.

C. Limitations of Diffie Hellman key Exchange as are:

1) There is no identity of the parties involved in the exchange.

2) It is easily susceptible to man-in-the-middle attacks. A third party C, can exchange keys with both A and B, and can listen to the communication between A and B.

3) The algorithm is computationally intensive. Each multiplication varies as the square of $n$, which must be very large. The number of multiplications required by the exponentiation increases with increasing values of the exponent, x or y in this case.

4) The computational nature of the algorithm could be used in a denial-of-service attack very easily.

5) The algorithm cannot be used to encrypt messages.

6) There is also a lack of authentication.

A Proposed Algorithm From RSA and Diffie Hellman Key Exchange

A Proposed Algorithms:
1. Choose two large prime numbers P and Q.
2. Calculate N = P x Q.

3. Select the public key (i.e. encryption key) $E$ such that it is not a factor of $(P - 1)$ and $(Q - 1)$.

4. Select the private key (i.e. the decryption key) $D$ such that the following equation is true $(D \times E) \mod (P - 1) \times (Q - 1) = 1$

5. Suppose R, S and G is automatic generated prime constants.

6. And put the value of $E$ and $D$ from above as secret number such that A=$E$ and B=D.

7. Now calculate following as public number

1. $X = GA \mod R$
2. $Y = GB \mod R$
3. Calculate session key $K_A = YA \mod R$ or $K_A = (GB \mod R)A \mod R$ or $K_A = (GB) A \mod R$ or $K_A = GBA \mod R$.
4. $K_B = XB \mod R$ or $K_B = (GA \mod R)B \mod R$ or $K_B = (GA) B \mod R$ or $K_B = GAB \mod R$.

3. Such that $K_A = K_B = K$.

1. For Encryption we XOR session key $K$ with Plain text $PT$ that will generate a new Cipher text $CT$ such as

1. $CT = PT \oplus K$

1. Then send $CT$ as the cipher text to the receiver and for decryption, calculate the plain text $PT$ from the cipher text $CT$ as follows:

1. $PT = CT \oplus K$

VI. NEW PROPOSED ALGORITHM FROM RSA AND DIFFIE HELLMAN KEY EXCHANGE

VII. REFERENCES

[1] Self-Organized Public Key Cryptography in Mobile Ad hoc Network


