Providing Security in Data Aggregation using RSA Algorithm

Er. Kumar Saurabh
Assistsnt Professor
Ramgharia Institute of Eng. & Technology
Phagwara

Sukhpreet Singh (Research fellow)
Mtech 4th sem CSE
Ramgarhia Institute of Engg. & Tech.
Phagwara

Abstract

The RSA algorithm proposed by Rivest, Adi Shamir and Leonard Adleman is cryptography technique. The current problem with wireless sensor network is how to protect the security of the sensor data. RSA algoritm is used as a digital signature authentication in the field of security, basically works on deciding encryption variable. In this also the basic concept is to decide a description variable and then decide the description variable using same encryption variable. It is a secure and fast cryptographic system. The major effort will be applied on the RSA encryption technique in order to make node authenticated as well as to secure data while dealing with aggregation.

Keywords- RSA; Wireless Sensor Network

I. INTRODUCTION

A Wireless Sensor Network [1] is a self-configuring network of small sensor nodes communicating among themselves using radio signals, and deployed in quantity to sense, monitor and understand the physical world. **W**ireless Sensor Network are used to collect data from the environmet. Wireless Sensor Network consist of number of sensor nodes and one or more base station.The nodes in the network are connected via wireless communication channels.Each node has capability to sense data,process the data and send it to rest of nodes or to base station.These network are limited by the node battery lifetime.The sensor nodes are small, they are able to detect events, gather information , process it and then send the processed data back to sink node.The sensor nodes have limited range and as they are battery operated ,they have limited power.As steted earlier the source of these sensor nodes depend upon battery, so enhancing the working period of a wireless sensor network is an important issue. [2,3]

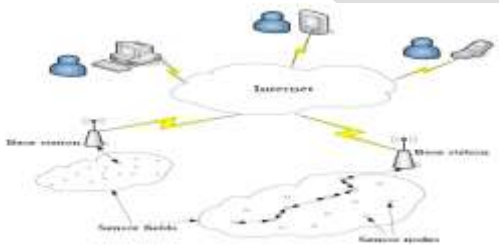

Figure1: Basic Wireless Sensors Network

The main design goal of wireless sensor networks is to transmit data by increasing the lifetime of the network and by employing energy efficient routing protocols. Depending on the applications used, different architectures and designs have been applied in sensor networks. Again, the performance of a routing protocol depends on the architecture and design of the network, so the architecture and design of the network is very important features in WSNs. The design of the wireless sensor network is affected by many challenging factors which must be overcome before an efficient network can be achieved in WSNs.

Node distribution [11] in WSNs is either deterministic or self-organizing and application dependant. The uniformity of the node distribution directly affects the performance of the routing protocol used for this network. In the case of deterministic node distribution, the sensor nodes are mutually placed and gathered data is transmitted through pre-determined paths. In the other case, the sensor nodes are spread over the area of interest randomly thus creating an infrastructure in an *ad hoc* manner. Network Dynamicity: Since the nodes in WSNs may be static or dynamic, dynamicity of the network is a challenging issue. Most of the routing protocols assume that the sensor nodes and the base stations are fixed *i.e.*, they are static, but in the case of dynamic BS or nodes routes from one node to another must be reported periodically within the network so that all nodes can transmit data via the reported route. Again depending on the application, the sensed event can be dynamic or static. For example, in target detection/tracking applications, the event is dynamic, whereas forest monitoring for early fire prevention is an example of a static event. Monitoring static events works in reactive mode. On the other hand, dynamic events work in proactive mode.

The sensor nodes in WSNs have limited energy and they use their energy for computation, communication and sensing, so energy consumption is an important issue in WSNs. According to some routing protocols nodes take part in data fusion and expend more energy. Since the transmission power is proportional to distance squared, multi-hop routing consumes less energy than direct communication, but it has some route management overhead. In this regard, direct communication is efficient. Since most of the times sensor nodes are distributed randomly, multi-hop routing is preferable. In some applications nodes sense environment periodically and lose more energy than the nodes used in some applications where they sense environment when some event occurs.

Data Transmission: Data transmission in WSNs is application specific. It may be continuous or event driven or query-based or hybrid. In case of continuous data transmission, sensor nodes send data to the base station periodically. In event driven and query-based transmission they send data to the base station when some event occurs or a specific query is generated by the base station. Hybrid transmission uses a combination of continuous, event driven and query-based transmission, so for architecture and design of WSNs data transmission is a very significant issue.
Scalability: A WSN consists of hundreds to thousands of sensor nodes. Routing protocols must be workable with this huge number of nodes *i.e.*, these protocols can be able to handle all of the functionalities of the sensor nodes so that the lifetime of the network can be stable.

Data Fusion: Data fusion [4] is a process of combining of data from different sources according to some function. This is

achieved by signal processing methods. This technique is used by some routing protocols for energy efficiency and data transfer optimization. Since sensor nodes get data from multiple nodes, similar packets may be fused generating redundant data. In data fusion or data aggregation process awareness is needed to avoid this redundant data.

II. ATTACKS IN WIRELESS SENSOR NETWORK

HELLO Flood Attacks

HELLO messages are used in many protocols by nodes that want to announce their presence and proximity to their neighbors. Most of these protocols rely on the assumption that a node *A is within the radio transmission range of another node B* if *A is able to receive messages from B. In a HELLO flood* attack, a malicious node may try to transmit a message with an abnormally high power so as to make all nodes believe that it is their neighbor .[5]

Wormhole Attacks

A wormhole attack: where an attacker receives packets at one location in the network, tunnels and then replays them at another remote location in the network. The route request can be tunneled to the target area by the attacker through wormholes. Thus, the sensor nodes in the target area build the route through the attacker. Later, the attacker can tamper the data, messages, or selectively forward data messages to disrupt the functions of the network

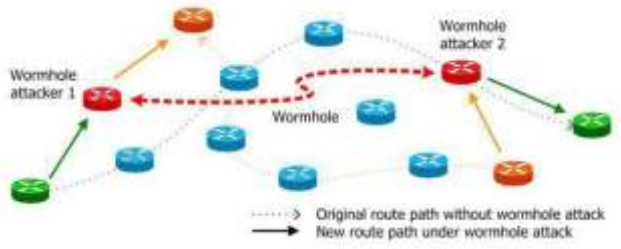

Figure2 : Wormhole Attack In WSN

Jamming Attacks

Jamming, a well-known attack on wireless communication is simply interference with the radio frequencies used by a device‘s transceiver. It represents an attack on the availability of a network. Jamming is only different from normal radio propagation in that it is unwanted and disruptive, thus creating a denial-of-service condition

Back hole Attacks

Selective forwarding is a more subtle attack in which some packets are correctly forwarded but others are silently and intentionally dropped. A compromised node could be configured to drop all packets, creating a so-called black hole

Sinkhole Attacks

In the sinkhole attack, a node spuriously advertises a very good route to a sink node in order to lure all nearby traffic to itself. Thus all traffic within some sphere of influence is drawn into the sinkhole centered at the compromised node. This attack enables the selective forwarding attack along with other attacks.[6,7]

Sybil Attack

The Sybil attack occurs when a single node claims to be other nodes in the network. Karloff and Wagner claim that this attack significantly reduces the effectiveness of ―fault-tolerant schemes‖ such as distributed storage, multipath routing, and topology maintenance

III. INTRODUCTION TO DIGITAL ENCRYPTION

RSA is an algorithm for public-key cryptography that is based on the presumed difficulty of factoring large integers, the factoring problem. RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman, who first publicly described it in 1978. A user of RSA creates and then publishes the product of two large prime numbers, along with an auxiliary value, as their public key. The prime factors must be kept secret. Anyone can use the public key to encrypt a message, but with currently published methods, if the public key is large enough, only someone with knowledge of the prime factors can feasibly decode the message.[1] Whether breaking RSA encryption is as hard as factoring is an open question known as the RSA problem.[8]

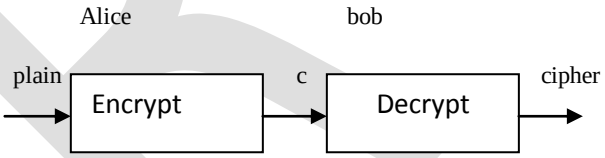

Figure3: Plain to Cipher conversion

Alice encrypts message to Bob using Bob’s Private Key. Only Bob knows Bob’s Private Key ⇒ only Bob can decrypt message.

A. RSA OPERATION

The RSA algorithm involves three steps: key generation, encryption and decryption.

a) Key generation

RSA involves a **public key** and a **private key.[9,10]** The public key can be known to everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted using the private key. The keys for the RSA algorithm are generated the following way:

1. Choose two distinct prime numbers p and q.
   - For security purposes, the integers p and q should be chosen at random, and should be of similar bit-length. Prime integers can be efficiently found using a primality test.
2. Compute $n = pq$.
   - n is used as the modulus for both the public and private keys
3. Compute $\varphi(n) = (p - 1)(q - 1)$, where φ is Euler's totient function.
4. Choose an integer e such that $1 < e < \varphi(n)$ and greatest common divisor of $(e, \varphi(n)) = 1$; i.e., e and $\varphi(n)$ are coprime.
   - e is released as the public key exponent.

- *e* having a short bit-length and small Hamming weight results in more efficient encryption - most commonly 0x10001 = 65,537. However, small values of *e* (such as 3) have been shown to be less secure in some settings.[4]

5. Determine *d* as:

$$d \equiv e^{-1} \pmod{\varphi(n)}$$

i.e., *d* is the multiplicative inverse of *e* mod φ(*n*).

- This is more clearly stated as solve for d given (de) mod φ(*n*) = 1
- This is often computed using the extended Euclidean algorithm.
- *d* is kept as the private key exponent.

so, d*e= 1 mod φ(*n*) The **public key** consists of the modulus *n* and the public (or encryption) exponent *e*. The **private key** consists of the modulus *n* and the private (or decryption) exponent *d* which must be kept secret. (*p*, *q*, and φ(*n*) must also be kept secret because they can be used to calculate *d*.)

Notes:

- An alternative, used by PKCS#1, is to choose *d* matching $de \equiv 1 \bmod \lambda$ with $\lambda = \text{lcm}(p-1, q-1)$, where lcm is the least common multiple. Using λ instead of φ(*n*) allows more choices for *d*. λ can also be defined using the Carmichael function, λ(*n*).
- The ANSI X9.31 standard prescribes, IEEE 1363 describes, and PKCS#1 allows, that *p* and *q* match additional requirements: be strong primes, and be different enough that Fermat factorization fails.

b)Encryption

Alice transmits her public key (n, e) to Bob and keeps the private key secret. Bob then wishes to send message **M** to Alice.

He first turns **M** into an integer m, such that $0 < m < n$ by using an agreed-upon reversible protocol known as a padding scheme. He then computes the ciphertext c corresponding to

$$c = m^e \pmod{n}$$

This can be done quickly using the method of exponentiation by squaring. Bob then transmits c to Alice.

Note that at least nine values of m could yield a ciphertext c equal to m,[5] but this is very unlikely to occur in practice.

c)Decryption

Alice can recover m from c by using her private key exponent d via computing

$$m = c^d \pmod{n}$$

Given m, she can recover the original message **M** by reversing the padding scheme.

IV. PROPOSED ALGORITHM

Source Node
If (Any Packet sent P)
{
Alter Header add columns _Route‘ and _Signatures‘

Insert id into Route Column
Insert Digital Signature into Signature Column
Forward Packet P
}
If (received A Packet)
{
If (Received Packet==Data_Ack)
{
Note the _Signature‘ in the header
Note Route Noted In header
Verify the Digital Signature
If(Verification Successful)
{
Discard the route noted
Else
{
Drop the packet
}
Repeat the Procedure for next packet;
}
}

Intermediate Node
If (Received a packet P)
{
Insert id into Route Column
Insert Digital Signature into Signature Column
Forward Packet P
}
Destination Node
If (Received a packet P)
{

Note the _Signature‘ in the header
Note Route Noted In header
Verify the Digital Signature
If(Verification Successful)
{
Noted Route=null;
}
Else
{
Noted Route unchanged
}
Create Data_Ack Packet
Insert columns _Route‘ and _Signatures in Data_Ack
Insert id into Route Column
Insert Digital Signature into Signature Column
}

V. ANALYSIS OF THE PROPOSED ALGORITHM

The results produced in data aggregation in wireless sensor networks are much better than any other algorithm. The energy efficiency and timing constraints are also better. It provides graphical clear results which are easy to understand.

VI. RESULT AND CONCLUSION

In this simulation i have implemented security using RSA algorithm. All the results are shown in MATLAB. In the first scenario source to destination communication is performed using shortest path algorithm. The root is shown in Green with arrows.

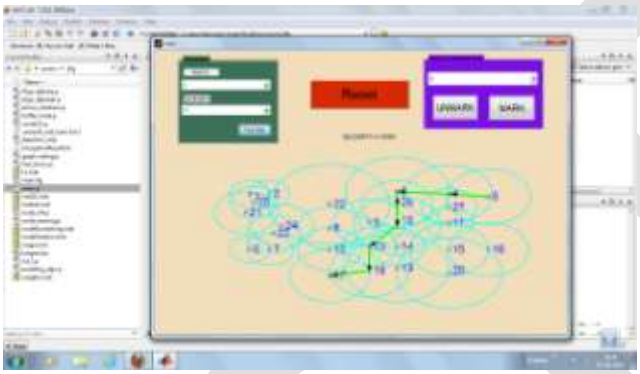

Figure 4: Graph shows the path taken from source to destination

In the Second scenario the malicious attack is performed and detected through security algorithm. The data packets take a new route to the destination.

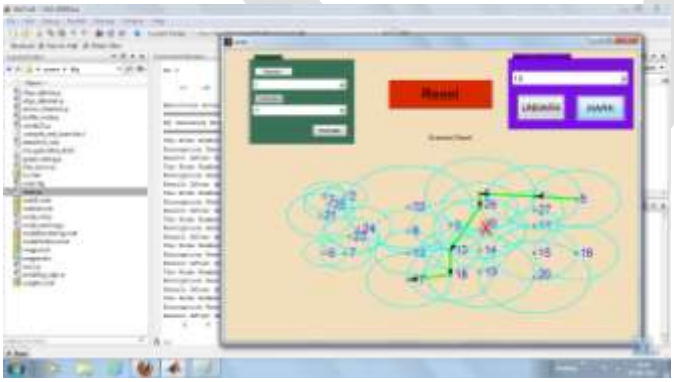

Figure5 : Graph shows marked malicious node and alternative path taken

In this scenario energy of nodes is shown after the attack and the energy level is almost same , so it does not loss any data during aggregation.

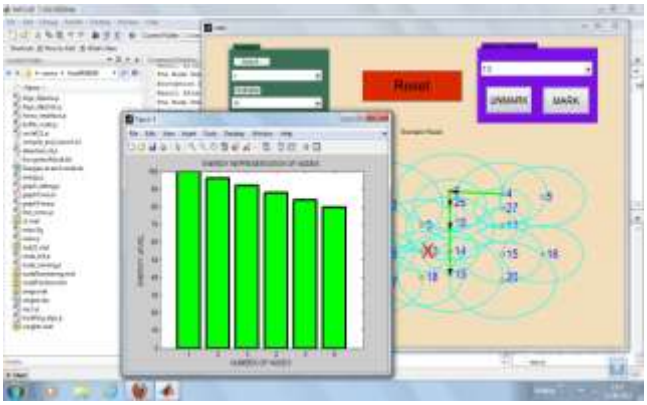

Figure6 :Graph shows energy representation of nodes

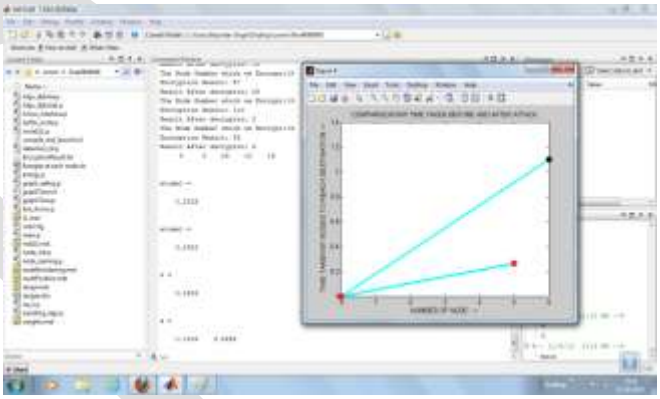

Figure7: Graph shows comparison between time taken before and after attack

In the above scenario the time taken by various nodes during data transmission is shown. After the particular attack the path taken by nodes is the second optimal path.

V. CONCLUSION

Security related issues in WSN have become an important part of research in present scenario. Detecting abnormal/malicious function of nodes and offering efficient counter measures is a difficult task. In the proposed paper a method that‘s Node Authentication Method. In the proposed method there is no need for specific hardware and neither is the need for clock synchronization due to use of the cryptographic concepts like digital signature. In that each node is authenticated using digital signature(RSA). The received node at the destination node is verified and if the digital signature is false the information about that is sent to the sender node using DATA_ACK..In the future there is a plan to simulate Node Authentication method with other wireless sensor protocols.

REFERENCES

[1] K.P. AlSakib, T.D. Tran, S.H. Chong. A Key Management Scheme with Encoding and Improved Security for Wireless Sensor Networks. ICDCIT 2006:102-115.

[2] L. Esehenauer, V. D. Gligor. A key-management scheme for distributed sensor networks. Proceedings of the 9th ACM

[3] R Gennaro. (2000), "Robust and Efficient Sharing of RSA Functions", Journal of Cryptology, Vol 13, No 2, pp 273-300

[4] Y Frankel, PD MacKenzie, M Yung, STOC. (1998), "Robust Efficient Distributed RSA-Key Generation", Proceedings of the thirtieth annual ACM symposium on Theory of computing, pp 663-672.

[5] Capacity of Ad Hoc Wireless Networks, in the proceedings of the 7th ACM International

[6] Laura Marie Feency .Ataxomony for routing protocols in mobile ad hoc networks technical report, institute of computer surden 1999

[7] D Boneh, M Franklin. (2001), "Efficient generation of shared RSA keys", Journal of the ACM, Vol 48, No. 4, pp 702-722.

[8] Y.Lili, W.Lijun, "Integration into The Chinese Remainder Theorem and Montgomery Algorithm for Fast RSA Algorithm",

[9]]R. Ahlswede, N. Cai, S. Y. R. Li, and R. W. Yeung. Network information flow. *Information Theory, IEEE Transactions on*, 46(4):1204–1216, 2000.

[10] Y.Lili , W. Lijun, " Integration into The Chinese Remainder Theorem and Montgomery Algorithm for Fast RSA Algorithm.