VOL. 10, NO. 16, SEPTEMBER 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

SDTOR: SECURE DATA TRANSMISSION OF OPTIMUM ROUTING PROTOCOL IN WIRELESS SENSOR NETWORKS FOR SURVEILLANCE APPLICATIONS

E. Vishnupriya[1], T. Jayasankar[2] and P. Maheswara Venkatesh[2]
[1]M. E, Communication Systems, Anna University (BIT Campus), Trichy, India
[2]Department of Electronics and Communication Engineering, Anna University (BIT Campus), Trichy, India
E-Mail: evishnupriyaa@gmail.com

ABSTRACT

Sensor networks are used sometime in very sensitive applications such as health care and military. Wireless sensor networks require the need for effective security mechanism. In a secure data transmission scenario need to decrypt the encrypted data to perform aggregation. In this paper, ECC (Elliptic Curve Cryptography) algorithm is using for encryption and decryption method. It protects all data against malicious modification and information forgery. The optimum routing protocols are such as LEACH, PEGASIS, APTEEN and AOMDV routing protocols. Sensor network routing protocols were very simple and not developed as security in mind. So the adversary can launch various attacks in the network. The ECC algorithm is used to safe guard from different attacks by building a secure route from source to sink node. The routing protocol suffers from many attacks like spoofing or altering the route information, selective forwarding, sinkhole attack, worm hole attack, Sybil attack, flooding attack. Encryption and decryption has been evaluated in terms of data delivery ratio and level of security. Data delivery ratio can be achieved 85% using ECC algorithm. Level of security up to the level compared to other asymmetric and symmetric algorithms. Base station should receive unaltered and fresh data.

Keywords: wireless sensor networks, attacks, ECC algorithm, cryptography techniques, security, encryption, decryption.

INTRODUCTION

Generally, security issue in routing protocol have not given much attention, since most of the routing protocol in WSNs have not been developed with security in mind. Many hierarchical routing protocols have been developed, where energy efficiency is the main goal. In many applications like military and battle field, data is important and have to maintain secrecy in data communication between sensor nodes and BS [1]. Security is a well-established field for general-purpose computing where security mechanisms address computing services like authentication, intrusion detection and provide secure transaction. Since the battery life confines the lifetime of a sensor node, power consumption is normally set as the first priority in developing security solutions. Sensor networks are deployed in a hostile environment, security becomes extremely important as these networks are prone to different types of malicious attacks. To provide security, communication transactions should be encrypted and authenticated. Symmetric key scheme is more appropriate cryptography (SKC) for wireless sensor networks due to its low energy consumption and simple hardware requirements, but most of them cannot provide sufficient security level as public key cryptography like integrity, confidentiality and authentication [2].

Cryptographic primitives are the basis of security solutions and the most frequently executed security operations in sensor networks. Cryptography is the art of achieving security by encoding messages to make them non-readable. Cryptography is the study of hiding information that enables to store sensitive information and also transmit it across insecure networks but it cannot be read by anyone except the intended recipient.

Symmetric algorithms, both parties share the same key for encryption and decryption. The most common types are i) Symmetric Key Cryptography and ii) Public Key Cryptography. A public key cryptography algorithm uses two different keys for encryption and decryption. The key used for decryption kept secret (Private) whereas the encryption key can be distributed openly (Public). Encryption algorithms and their use are essential part of the secure transmission of information. There are extensive studies on using symmetric- key cryptography to achieve various aspects of security in sensor networks. The symmetric key function is used to guarantee secure communications between in-network nodes while the public key function is used to guarantee a secure data delivery between the source node and the sink node [2]. Cryptography can be defined as conversion of data plaintext (ordinary text) into cipher text (known as encryption), then back again (known as decryption) into plain text. Due to the resource constraints, security and cryptography is an open issue for WSNs. A cryptographic algorithm works in combination with a key, a word, number, or phrase to encrypt the plaintext. The same plaintext encrypts to different cipher text with different keys. "Cryptography" derives from the Greek word kryptons, meaning "hidden"[2]. In most of applications, sensor devices are spread over large areas, what difficult a individual control of network components. Moreover, wireless communication allows an attacker can trigger attacks without having physical access to the device, so

according to Shi and Perrig [7] attacks on WSNs can be divided into three main types: (1) Attack of authentication and confidentiality: Consists of attacks change, repetition or modification packages. (2) Availability network Attack: Generally known as DoS attacks or negation of service, this attack involves the application of techniques that make the network unavailable. (3) Attack on integrity: this type of attack the attacker's goal is to inject false data on the network, keeping the network available, but traveling fictitious data.

Table-1. OSI layer attacks.

OSI Layer	Attacks
Physical Layer	Device Tempering, Eves dropping, Jamming
Application Layer	Clock Skewing, Selective message forwarding, Data Aggregation Distortion
Network Layer	False Routing, Packet Replication, Blackhole, Worm Hole, Sink hole
MAC Layer	Traffic Manipulation, Identity Spook

RELATED WORKS

Suraj Sharma *et al*., described the study of different hierarchical routing technique for WSNs. Additionally analyze and compared secure hierarchical routing protocols based on various criteria. Routing protocol affects the performance of the network in the form of energy efficiency, security, resiliency and lifetime. So that secure, robust and efficient routing protocol is the basic requirement. Studied and analyzed a number of secure and energy efficient hierarchical routing protocols for WSNs [1].

Shanta Mandal *et al*., proposed the scheme of overcomes the limitations of public-key and symmetric-key protocols for wireless sensor networks in respect of low energy consumption. The symmetric-key function is used to guarantee secure communications between the nodes in a network while the public-key function is used to guarantee a secure data delivery between the sources to sink. This scheme provides mix of symmetric-key and public-key cryptography functions using the pre-distributed keys to implement data confidentiality service and special attention for data authenticity.

Gustavo S. Quirino *et al*. described the study of symmetric and asymmetric cryptography algorithms. The security of data transmissions from these devices should be improved in a preventative manner to avoid possible attacks. Regarding WSNs, RSA public key algorithm is the most commonly used is standardized, and achieves efficiency relatively good. The algorithm based on elliptic curves is alternative to RSA, and the results achieved good results with smaller keys. The algorithm MQQ is post-quantum, and may even be good solution when quantum computation is standardized. It shows significant results when compared to RSA and ECC, taking as parameters authenticity and digital signature.

Gaurav Sharma *et al*. analyzed all cryptography frame works designed so far. Also a compared the various frame works for different parameters like encryption, ciphering, freshness, key agreement, code requirement, authentication, cost and which mote supports this frame work. WSN suffer from many constraints including lower processing power, low battery life, small memory and wireless communication channel, security becomes the main concern to deal with such kind of networks. Due to these well accepted limitations, WSN is not able to deal with traditional cryptographic algorithms.

ARCHITECTURE DIAGRAM

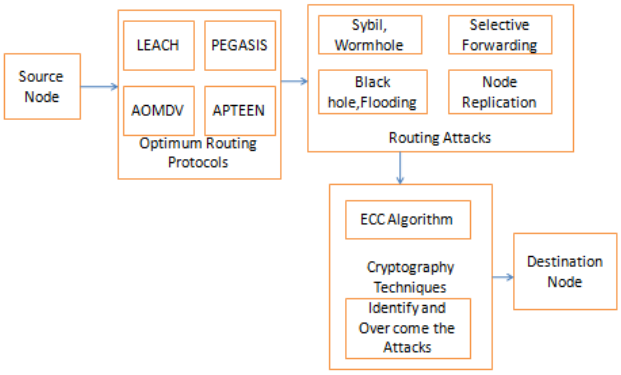


Figure-1. Architecture diagram.

Security requirements

Confidentiality - Confidentiality ensures the concealment of the message from an attacker so that any message communicated via the sensor network remains confidential; it shows in Figure-2. In a WSN, the issue of confidentiality should address the following requirements: (i) a sensor node should not allow its readings to be accessed by its neighbors unless they are authorized to do so, (ii) key distribution mechanism should be extremely robust, (iii) public information such as sensor identities, and public keys of the nodes should also be encrypted in certain cases to protect against traffic analysis attacks.

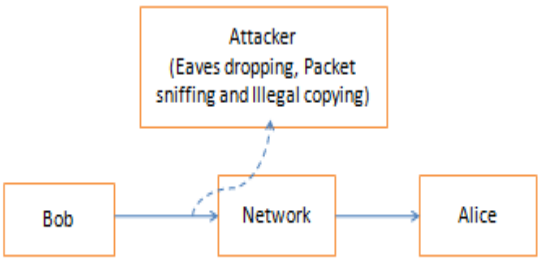


Figure-2. Confidentiality.

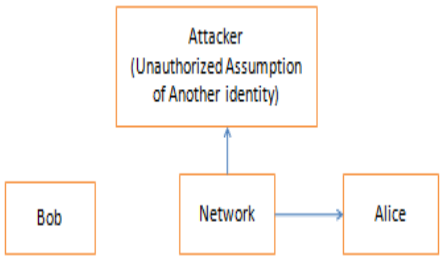


Figure-3. Authentication.

Authentication - Authentication ensures the reliability of the message by identifying its origin. By authenticating other nodes, cluster heads, and base stations before granting a limited resource, or revealing information, it shows in Figure-3. In a WSN, the issue of authentication should address the following requirements: (i) communicating node is the one that it claims to be, (ii) receiver node should verify that the received packets have undeniably come from the actual sender node.

Integrity - Integrity ensures the reliability of the data and refers to the ability to confirm that a message has not been tampered with, altered or changed while on the network, it shows in Figure-4. In a WSN, the issue of integrity should address the following requirements: (i) only the nodes in the network should have access to the keys and only an assigned base station should have the privilege to change the keys. This would effectively thwart unauthorized nodes from obtaining knowledge about the keys used and preclude updates from external sources. (ii) It protects against an active, intelligent attacker who might attempt to disguise his attack as noise.

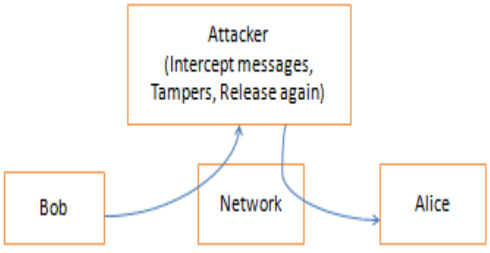


Figure-4. Integrity.

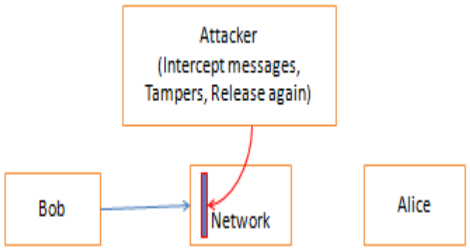


Figure-5. Availability.

Availability - Availability ensures the services of resources offered by the network, or by a single sensor node must be available whenever required, it shows in Figure-5. In a WSN, the issue of availability should address the following requirements: (i) the security mechanisms should be available all the time; a single point of failure should be avoided, (ii) the mechanism is used as a central access control system to ensure successful delivery of every message to its recipient node.

Attacks on routing protocol

The most vulnerable attack in terms of exhaustion of resources in WSN is Denial of Service attacks (DOS). Denials of Service attacks are specific attacks that attempt to prevent legitimate users from accessing networks, servers, services or other resources by sending extra unnecessary packets and thus prevent legitimate network users from accessing services or resources [7] [8] [9].

Black hole attack

Black hole attack is also known as sink holes attack occurring at the network layer. It builds a covenant node that seems to be very attractive in the sense that it promotes zero-cost routes to neighboring nodes with respect to the routing algorithm. This results maximum traffic to flow towards these fake nodes. Nodes adjoining to these harmful nodes collide for immense bandwidth, thus resulting into resource contention and message destruction.

Wormhole attack

In the wormhole attack, pair of awful nodes firstly discovers a wormhole at the network layer. The whole traffic of the network is tunneled in a particular direction at a distant place, which causes deprivation of data receiving in other parts of the network; it shows in Figure-6. These packets are then replayed locally. This creates a fake scenario that the original sender is only one or two nodes away from the remote location. This may cause congestion and retransmission of packets squandering the energy of innocent nodes.

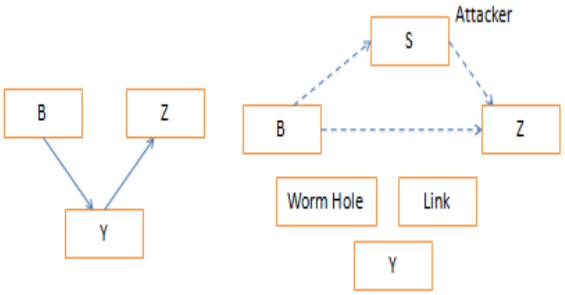


Figure-6. Worm hole attack.

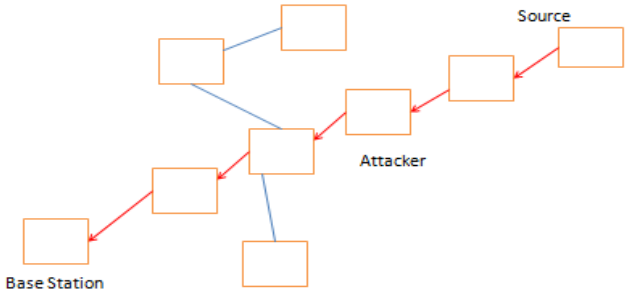


Figure-7. Selective forwarding attack.

Selective forwarding attack

Selective forwarding is a network layer attack. In this, an adversary covenants a node, that it scrupulously forwards some messages and plunge the others. This hampers the quality of service in WSN. If the attacker will drop all the packets then the adjoining nodes will become conscious and may evaluate it to be a flaw. To avoid this, the attacker smartly forwards the selective data. To figure out this type of attack is a very tedious job. It is a situation when certain nodes do not forward many of the messages they receive, it shows in Figure-7. The sensor networks depend on repeated forwarding by broadcast for messages to propagate throughout the network [8].

Flooding

Flooding also occurs at the network layer. An adversary constantly sends requests for connection establishment to the selected node. To hit each request, some resources are allocated to the adversary by the targeted node. This may result into effusion of the memory and energy resources of the node being bombarded.

Sybil attack

This is also a network layer attack. In this, an awful node presents more than one character in a network. It was originally described as an attack able to defeat the redundancy mechanisms of distributed data storage systems in peer-to-peer networks. The Sybil attack is efficient enough to stroke other fault tolerant schemes such as dispersity, multi path routing, routing algorithms, data aggregation, voting, fair resource allocation, topology maintenance and misbehavior detection. The fake node implies various identities to other nodes in the network and thus occurs to be in more than one place at a time, it shows in Figure-8. In this way, it disturbs the geographical routing protocols. It can collide the routing algorithms by constructing many routes from only one node.

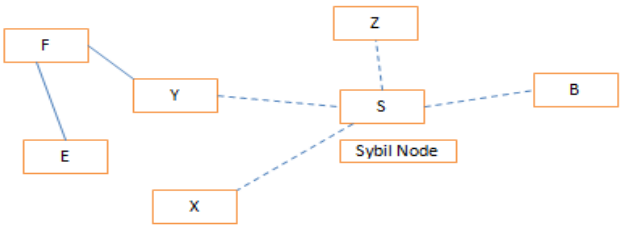


Figure-8. Sybil attack.

Node replication attack

Every sensor node in a network has a unique ID. This ID can be duplicated by an attacker and is assigned to a new added malicious node in the network. This assures that the node is in the network and it can lead to various calamitous effects to the sensor network. By using the replicated node, packets passing through malicious node can be missed, misrouted or modified. This results in wrong information of packet, loss of connection, data loss and high end-to-end latency. Malicious node can get authority to the sensitive information and thus can harm the network.

Types of cryptographic functions

The Figure-9 shows that there are three kinds of cryptographic functions: (1) hash functions, (2) secret key functions, and (3) public key functions. Public key cryptography involves the use of two keys. Secret key cryptography involves the use of one key. Hash functions involve the use of zero keys.

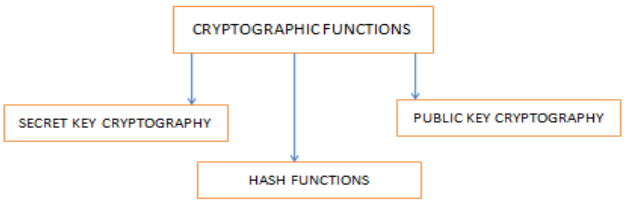


Figure-9. Types of cryptography functions.

Secret key cryptography

Secret key cryptography involves the use of a single key; it shows in Figure-10. Given a message (called plaintext) and the key, encryption produces unintelligible data which is about the same length as the plaintext was. Decryption is the reverse of encryption, and uses the same key as encryption. Secret key cryptography is sometimes

referred to as conventional cryptography or symmetric cryptography.

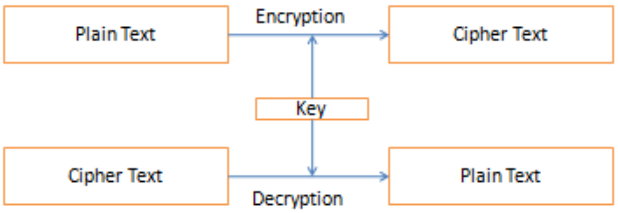


Figure-10. Secret key cryptography.

Suppose Alice and Bob share a key K_{AB} and they want to verify they are speaking to each other. They each pick a random number, which is known as a challenge. In Figure-11, Alice picks r_A. Bob picks r_B. The value x encrypted with the key K_{AB} is known as the response to the challenge x.

Figure-11. Encrypted process.

Public key cryptography

Public key cryptography is sometimes also referred to as asymmetric cryptography. Public key cryptography is a relatively new field, invented in 1975. In Figure-12 shows that unlike secret key cryptography, keys are not shared. Instead, each individual has two keys: a private key that need not be revealed to anyone, and a public key that is prefer-ably known to the entire world. There is something unfortunate about the terminology public and private. It is that both words begin with p. Sometimes want a single letter to refer to one of the keys. Use the letter e to refer to the public key, since the public key is used when encrypting a message. Use the letter d to refer to the private key, because the private key is used to decrypt a message. Encryption and decryption are two mathematical functions that are inverses of each other.

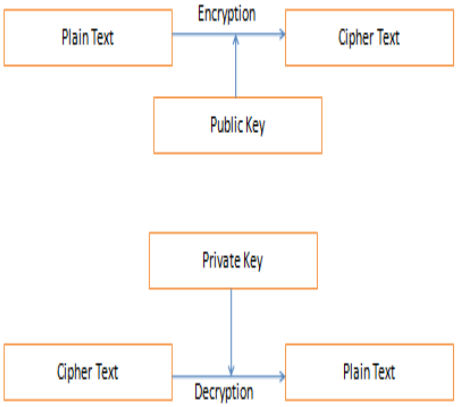


Figure-12. Public key cryptography.

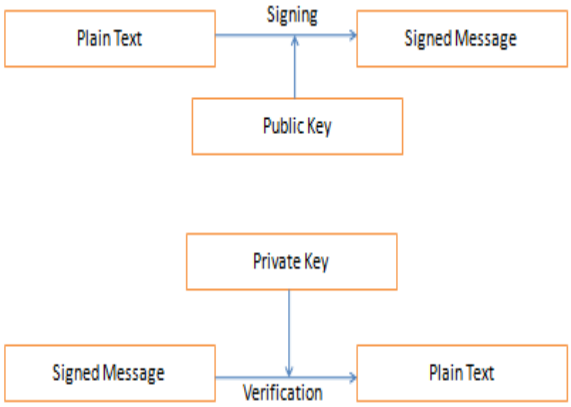


Figure-13. MIC process.

There is an additional thing one can do with public key technology, which is to generate a digital signature on a message as in Figure-13. A digital signature is a number associated with a message, like a checksum or the MIC (message integrity code). However, unlike a checksum, which can be generated by anyone, a digital signature can only be generated by someone knowing the private key. A public key signature differs from a secret key MIC because verification of a MIC requires knowledge of the same secret as was used to create it. Therefore any-one who can verify a MIC can also generate one, and so be able to substitute a different message and corresponding MIC. In contrast, verification of the signature only requires knowledge of the public key. So Alice can sign a message by generating a signature only she can generate, and other people can verify that it is Alice's signature, but cannot forget her signature. This is called a signature because it shares with handwritten signatures the property that it is possible to be able to recognize a signature as authentic without being able to forge it.

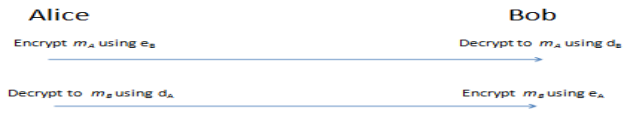


Figure-14. Encryption and decryption process.

Suppose Alice's ⟨public key, private key⟩ pair is $\langle e_A, d_A \rangle$. Suppose Bob's key pair is $\langle e_B, d_B \rangle$. Assume Alice knows Bob's public key, and Bob knows Alice's public key, shows in Figure-14. Actually, accurately learning other people's public keys is one of the biggest challenges in using public key cryptography.

Figure-15. Pair key process.

Another advantage of public key authentication is that Alice does not need to keep any secret information; it shows in Figure-15. For instance, Alice might be a computer system in which backup tapes are unencrypted and easily stolen. With secret key based authentication, if Carol stole a backup tape and read the key that Alice shares with Bob, it could then trick Bob into thinking she was Alice. In contrast, with public key based authentication, the only information on Alice's backup tapes is public key information, and that cannot be used to impersonate Bob.

Hash algorithms

Hash algorithms are used as components by other cryptographic algorithms and processes to provide information security services. Hash functions are often utilized with digital signature algorithms, keyed-hash message authentication codes, key derivation functions, and random number generators. A hash algorithm converts a variable length message into a condensed representation of the electronic data in the message. This representation, or message digest, can then be used for digital signatures, message authentication, and other secure applications. When employed in a digital signature application, the hash value of the message is signed instead of the message itself; the receiver can use the signature to verify the signer of the message and to authenticate the integrity of the signed message.

Types of cryptographic techniques

It is important to select the most appropriate cryptographic method because all the security requirements are ensured by cryptography. Cryptographic methods used in WSNs should meet the constraints of sensor nodes and be evaluated by code size, data size, processing time, and power consumption. However, sensor nodes are limited in their computational and memory capabilities, so the traditional cryptographic techniques cannot be simply transferred to WSNs. Consequently, to meet the above mentioned security requirements, either the existing techniques have to be adapted or novel techniques have to be developed. Based on the existing cryptographic techniques, it can classify them into three classes; it shows in Figure-16: symmetric cryptographic techniques, asymmetric cryptographic techniques and hybrid cryptographic techniques. Asymmetric cryptographic techniques can further be classified into three classes: RSA based techniques, ECC based techniques and pairing based techniques.

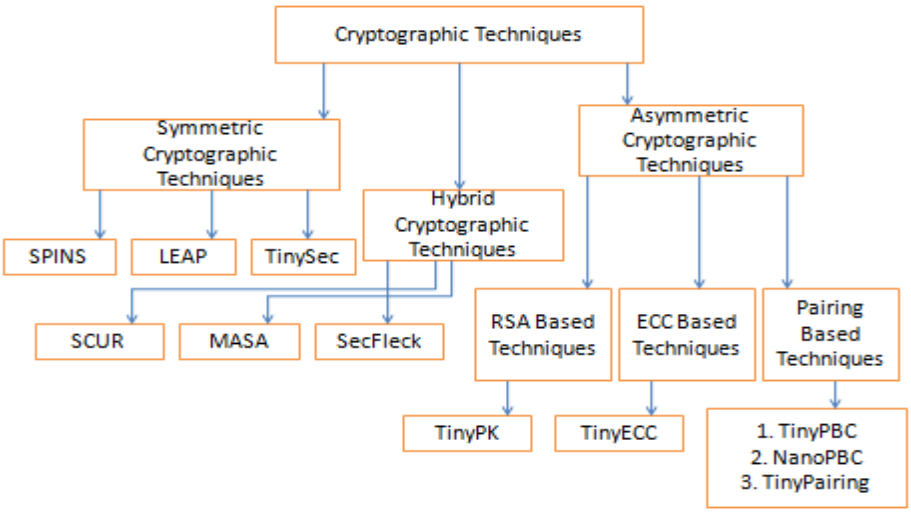


Figure-16. Types of cryptography techniques.

Symmetric cryptographic techniques

In symmetric cryptographic techniques, a single shared key is used between the two communicating nodes both for encryption and decryption; it shows in Figure-17. This key has to be kept secret in the network, which can be quite hard in the exposed environment where WSNs are used. Most security schemes for WSN use only symmetric cryptography, due to its ease of implementation on limited hardware and small energy demands, especially if the implementation is done in hardware to minimize performance loss. Two types of symmetric ciphers are used: block ciphers that work on blocks of a specific length and stream ciphers that work bitwise on the data. A steam cipher can be seen as a block cipher with a block length of 1 bit.

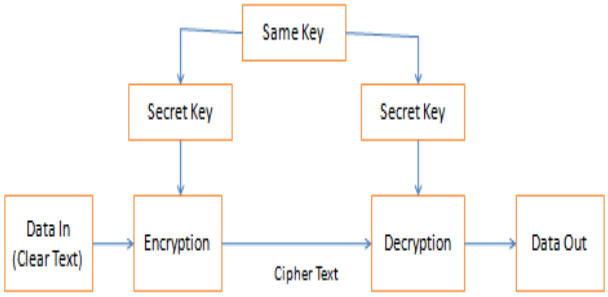


Figure-17. Symmetric cryptography.

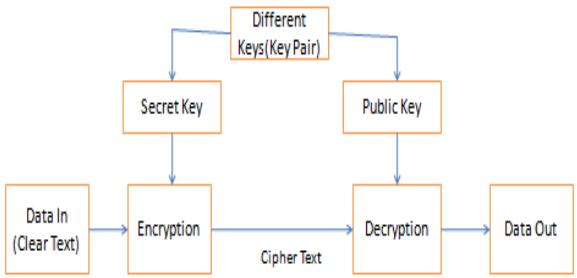


Figure-18. Asymmetric cryptography.

Asymmetric cryptographic techniques

In asymmetric cryptography, a private key can be used to decrypt and sign data while a public key can be used to encrypt and verify data. The private key needs to be kept confidential while the public key can be published freely. Asymmetric cryptography is also known as Public key cryptography; it shows in Figure-18. Public key cryptography tends to be resource intensive, as most systems are based on large integer arithmetic. For a number of years many researchers discarded public key cryptography as infeasible in the limited hardware used in WSN. The code size, data size, processing time, and power consumption make it undesirable for public key algorithm techniques, such as the Diffie-Hellman key agreement protocol or RSA signatures, to be employed in WSNs. There are various public key algorithms include Rabin's Scheme, Ntru-Encrypt, RSA, Elliptic Curve Cryptography (ECC), Pairing Based Cryptography (PBC) and Identity Based Encryption.

Differences between symmetric and asymmetric encryption algorithms

Symmetric encryption algorithms encrypt and decrypt with the same key. Main advantages of symmetric encryption algorithms are its security and high speed. Asymmetric encryption algorithms encrypt and decrypt with different keys. Data is encrypted with a public key, and decrypted with a private key. Asymmetric encryption algorithms (also known as public-key algorithms) need at least a 3, 000-bit key to achieve the same level of security of a 128-bit symmetric algorithm. Asymmetric algorithms are incredibly slow and it is impractical to use them to encrypt large amounts of data. Generally, symmetric encryption algorithms are much faster to execute on a computer than asymmetric ones. In practice they are often used together, so that a public-key algorithm is used to encrypt a randomly generated encryption key, and the random key is used to encrypt the actual message using a symmetric algorithm. This is sometimes called hybrid encryption.

Hybrid cryptographic techniques

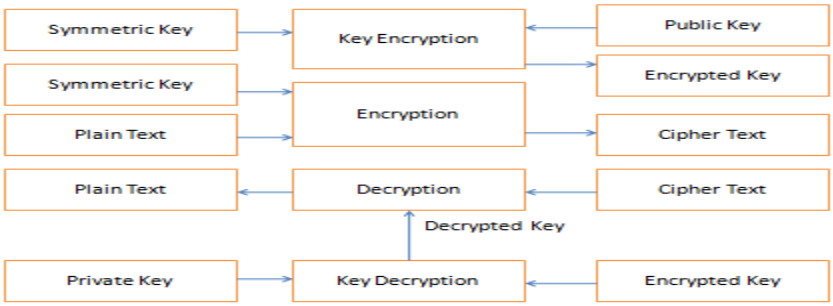


Figure-19. Hybrid cryptography.

Symmetric and asymmetric cryptography can be applied in combination to join the advantages of both approaches; it shows in Figure-19. Hybrid cryptographic scheme for the generation of pairwise network topology authenticated keys (TAK) in WSNs, which is based on vector algebra in $GF_{(q)}$. Symmetric is used for ciphering and authentication, while asymmetric is used for key generation.

Cryptography algorithms

Cryptographic frameworks for wireless sensor networks

In this section, discussed frameworks which are specifically designed and implemented to provide security to wireless sensor networks. We classify the existing frameworks according to the nature of the key material i.e. the shared key is private or public. We further classify asymmetric cryptographic frameworks into three classes as RSA based cryptographic frameworks, ECC based cryptographic frameworks and pairing based cryptographic frameworks [3].

Symmetric cryptographic frameworks

Cryptographic frameworks which are based on single shared key both for encryption and decryption. Such frameworks are: SPINS, Localized Encryption and Authentication Protocol (LEAP) and TinySec.

- SPINS - Perrig *et al*., 2001 proposed a security building block, which is optimized for resource-constrained environments and wireless communication. It based on two secure building blocks: SNEP (Secure Network Encryption Protocol).
- Localized Encryption and Authentication Protocol (LEAP) - Zhu *et al*., 2003 proposed a protocol that was designed to support in-network processing. The design of the protocol was based on the principle that different types of messages exchanged between sensor nodes have different security requirements; a single keying mechanism is not suitable for meeting these different security requirements
- TinySec - Karlof *et al*., 2004 introduced a lightweight, generic security package that developers can easily integrate into sensor network applications. It is the first fully-implemented protocol for link-layer cryptography in sensor networks. The implementation of TinySec is incorporated into the official TinyOS release. It includes some of the trade-offs between performance, transparency, and cryptographic security and a design is based on the needs of applications in the sensor network space. Bandwidth, latency, and energy costs of TinySec are low for sensor network applications. TinySec is easily extensible and has been incorporated into higher level protocols.

Asymmetric cryptographic frameworks

Cryptographic frameworks, which are based on two shared keys, private key for encryption and public key for decryption. Asymmetric cryptographic frameworks are further classified as RSA based, ECC based and Pairing based cryptographic frameworks.

RSA based cryptographic frameworks

RSA is computationally intensive and usually execute thousands or even millions of multiplication instructions to perform a single-security operation. The number of clock cycles required to perform a multiplication instruction primarily determines a microprocessor's public key algorithm efficiency.

- TinyPK - Watro *et al*., 2004 have described the design and implementation of public-key-based protocols; that allow authentication and key agreement between a sensor network and a third party as well as between two sensor networks.

ECC based cryptographic frameworks

In this subsection of asymmetric cryptographic frameworks, we discuss cryptographic frameworks, which uses ECC (Elliptical Curve Cryptography) algorithm for the security measures. The security of ECC is based on the elliptic curve discrete logarithm problem, which the cryptographic community regards as much more difficult than the integer factorization and discrete logarithm problems that underlie the conventional Rivest-Shamir-Adelman (RSA) and Diffie Hellman public-key algorithms. ECC has two main advantages: (1) ECC public keys are smaller for the same level of security as RSA or Diffic Hellman-based solutions, thus reducing the number of bits that need to be exchanged; and (2) ECC public-key operations require fewer computations than conventional public-key methods. The benefit of smaller key is that they need less storage, less bandwidth and, therefore, less energy, thereby reducing processing and communication overhead, which is ideal for energy-constrained sensor nodes.

- TinyECC - Liu and Ning, 2008 presented the design, implementation, and evaluation of TinyECC, a configurable library for ECC operations in wireless sensor networks. The primary objective of TinyECC is to provide a ready-to-use, publicly available software package for ECC-based PKC operations that can be flexibly configured and integrated into sensor network applications.

Pairing based cryptographic frameworks

Cryptographic framework which uses pairing based cryptography for the security measures. Cryptography using Pairings (PBC) is an emerging field related to ECC, which has been attracting the interest of international cryptography community, since it enables the design of original cryptographic schemes and makes well-known cryptographic protocols more efficient.

- TinyPBC - Oliveira *et al*., 2008 proposed TinyPBC, which is based on Multi-precision Integer and Rational Arithmetic C/C++ Library (MIRACL) which is a publicly available and open source library written in C.
- NanoPBC - Aranha *et al*., 2009 proposed a cryptographic library for resource -constrained devices. The authors implemented all big number, finite field, and elliptic curve arithmetic from scratch therefore it would allow to extract the most from the platform.
- TinyPairing - Xiong *et al*., 2010 proposed an efficient and lightweight pairing-based cryptographic library for sensors. It provides a better way to compute quickly as it consumes low memory for both the cases RAM and ROM by deliberately choosing some super-singular elliptic curve as the pairing group and some specific finite field, which defines the elliptic-curve pairing group.

Hybrid cryptographic frameworks

Cryptographic frameworks, which are based on the combination of two approaches; symmetric , cryptography and asymmetric cryptography.

- SCUR - The objective of the SCUR is to minimize cost effect of the following while maintaining required levels of security: (1) Communication overhead, in case of communicating the encrypted packet.(2) Computation over head in securing the network in order to save sensor's lifetime. (3) Utilized key space.
- MASA - Alzaid *et al*., 2008 proposed a security system known as MASA (Mixture of Asymmetric and Symmetric Approaches) to provide end-to-end data security for wireless sensor networks. It is based on the concept of virtual geographic grid wherein the entire terrain is broken down into smaller regions called cells. Each sensor carries two types of keys, asymmetric and symmetric. MASA uses the private key to sign a hashed event notification to provide confidentiality, authenticity, and data integrity. The symmetric key is used to authenticate the event notification within its cell.
- SecFleck - Hu *et al*., 2009 described the design and implementation of a public- key platform. It is based on a commodity Trusted Platform Module (TPM) chip that extends the capability of a standard node.

ENCRYPTION AND DECRYPTION ALGORITHM:

ECC algorithm

Elliptic curve cryptography (ECC) is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields. One of the main benefits in comparison with non-ECC cryptography (with plain Galois fields as a basis) is the same level of security provided by keys of smaller size. Elliptic curves are applicable for encryption, digital signatures, pseudo-random generators and other tasks. They are also used in several integer factorization algorithms that have applications in cryptography, such as Lenstra elliptic curve factorization.

Public-key cryptography is based on the intractability of certain mathematical problems. Early public-key systems are secure assuming that it is difficult to factor a large integer composed of two or more large prime factors. For elliptic-curve-based protocols, it is assumed that finding the discrete logarithm of a random elliptic curve element with respect to a publicly known base point is infeasible: this is the "elliptic curve discrete logarithm problem" or ECDLP. The entire security of ECC depends on the ability to compute a point multiplication and the inability to compute the multiplicand given the original and product points. The size of the elliptic curve determines the difficulty of the problem.

The primary benefit promised by ECC is a smaller key size, reducing storage and transmission requirements, i.e. that an elliptic curve group could provide the same level of security afforded by an RSA-based system with a large modulus and correspondingly larger key: for example, a 256-bit ECC public key should provide comparable security to a 3072-bit RSA public key. For current cryptographic purposes, an elliptic curve is a plane curve over a finite field (rather than the real numbers) which consists of the points satisfying the equation

$$y^2=x^3+ax+b$$

along with a distinguished point at infinity, denoted ∞.

At the RSA Conference 2005, the National Security Agency (NSA) announced Suite B which exclusively uses ECC for digital signature generation and key exchange. The suite is intended to protect both classified and unclassified national security systems and information. Recently, a large number of cryptographic primitives based on bilinear mappings on various elliptic curve groups, such as the Weil and Tate pairings, have been introduced. Schemes based on these primitives provide efficient identity-based encryption as well as pairing-based signatures, signcryption, key agreement, and proxy re-encryption.

Security

Side-channel attack

Unlike most other DLP systems (where it is possible to use the same procedure for squaring and multiplication) the EC addition is significantly different for doubling (P=Q) and general addition (P is not equal to Q) depending on the coordinate system used. Consequently, it is important to counteract side channel attacks (e.g., timing or simple/differential power analysis attacks) using, for example, fixed pattern window (a.k.a. comb) methods (note that this does not increase the computation time). Another concern for ECC-systems is the danger of fault attacks, especially when running on smart cards.

Cryptographic experts have also expressed concerns that the National Security Agency has inserted a backdoor into at least one elliptic curve-based pseudo random generator. One analysis of the possible backdoor concluded that an adversary in possession of the algorithm's secret key could obtain encryption keys given only 32 bytes of ciphertext.

Quantum computing attack

Elliptic curve cryptography is vulnerable to a modified Shor's algorithm for solving the discrete logarithm problem on elliptic curves. A quantum computer to attack elliptic curve cryptography can be less than half the size of a quantum computer to break an equivalently classically secure version of RSA. This is due to the fact that smaller key sizes of elliptic curves are needed to match the classical security of RSA. The work of Proos and Zalka show how a quantum computer to break 2048-bit RSA requires roughly 4096 qubits while a quantum computer to break the equivalently secure 224-bit Elliptic Curve Cryptography requires between 1300 and 1600 qubits. Depending on the growth rate of quantum computers in the future, elliptic curve cryptosystems may become attackable by a quantum computer many years before an equivalently secure RSA scheme.

Features

There are some widely used cryptographic algorithms which need a finite, cyclic group (a finite set of element with a composition law which fulfils a few characteristics), e.g. DSA or Diffie-Hellman. The group must have the following characteristics: Group elements must be representable with relatively little memory. The group size must be known and be a prime number (or a multiple of a known prime number) of appropriate size (at least 160 bits for the traditional security level of "80-bit security"). The group law must be easy to compute. It shall be hard (i.e. computationally infeasible, up to at least the targeted security level) to solve discrete logarithm in the group. Elliptic curve are another kind of group, appropriate for group-based cryptographic algorithm. An elliptic curve is defined with: A finite field, usually consisting in integers modulo some prime p (there are also other fields which can be used). A curve equation, usually y2=x3+ax+b, where a and b are constant values from the finite field, it shows in figure-20. The curve is the set of pairs of values (x, y) which match the equation, along with a conventional extra element called "the point at infinity". Since elliptic curves initially come from a graphical representations (when the field consists in the real numbers R), the curve elements are called "points" and the two values x and y are their "coordinates". Compared to the traditional multiplicative group modulo a big prime, elliptic curve variants of cryptographic algorithms have the following practical features:

- **They are small and fast**. There is no known efficient discrete-logarithm solving algorithm for elliptic curves, beyond the generic algorithms which work on every group. So we get appropriate security as soon as p is close to 160 bits. Computing the group law costs ten field operations, but on a field which is 6 times smaller; since multiplications in a finite field have quadratic cost, we end up with an appreciable speedup.

- **Some elliptic curves allow for pairings**. A pairing is a bilinear operation which can link elements from two groups into elements of a third group. A pairing for cryptography requires all three groups to be "appropriate" (in particular with a hard-to-solve discrete logarithm).

Elliptic curves are usually said to be the next generation of cryptographic algorithms, in order to replace RSA. Performance of EC computations is the main interest of these algorithms, especially on small embedded systems such as smartcards (in particular Koblitz curves over binary fields); the biggest remaining issue is that public-key operations with group-based algorithms are a bit slow (RSA signature verification or asymmetric encryption, as opposed to signature generation and asymmetric decryption, respectively, is extremely fast, whereas analogous operations in the group-based algorithms are just fast). Also, involved mathematics is a bit harder than with RSA.

The equation of an elliptic curve is given as,

$$y^2 = x^3 + ax + b$$

Few terms that will be used,

E -> Elliptic Curve P -> Point on the curve n -> Maximum limit (This should be a prime number)

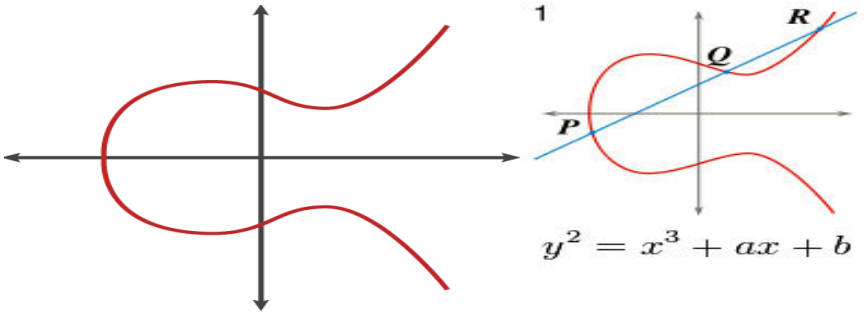


Figure-20. ECC curves.

Key generation

Key generation is an important part where we have to generate both public key and private key. The sender will be encrypting the message with receiver's public key and the receiver will decrypt its private key.

Q = d * P

d = The random number that we have selected within the range of (1 to n-1). Pis the point on the curve.

'Q' is the public key and 'd' is the private key.

Encryption

Let 'm' be the message that we are sending. We have to represent this message on the curve. This has in-depth implementation details. All the advance research on ECC is done by a company called certicom.

Conside 'm' has the point 'M' on the curve 'E'. Randomly select 'k' from [1 - (n-1)].

Two cipher texts will be generated let it be **C1** and **C2**.

C1 = k*P

C2 = M + k*Q

C1 and C2 will be send.

Decryption:

We have to get back the message 'm' that was send to us,

M = C2 – d * C1

M is the original message that we have send.

Proof:

How does we get back the message,

M = C2 – d * C1

'M' can be represented as 'C2 – d * C1'

C2 – d * C1 = (M + k * Q) – d * (k * P) (C2 = M + k * Q and C1 = k * P)

= M + k * d * P – d * k *P (canceling out k * d * P)

= M (Original Message)

OPTIMUM ROUTING PROTOCOLS

LEACH (Low energy adaptive clustering hierarchy)

LEACH is the first network protocol that uses hierarchical routing for wireless sensor networks to increase the life time of network. All the nodes in a network organize themselves into local clusters, with one node acting as the cluster-head; it shows in Figure-21. All non-cluster-head nodes transmit their data to the cluster-head, while the cluster-head node receive data from all the cluster members, perform signal processing functions on the data (e.g., data aggregation), and transmit data to the remote base station. Therefore, being a cluster-head node is much more energy-intensive than being a non-cluster-head node. Thus, when a cluster-head node dies all the nodes that belong to the cluster lose communication ability [6].

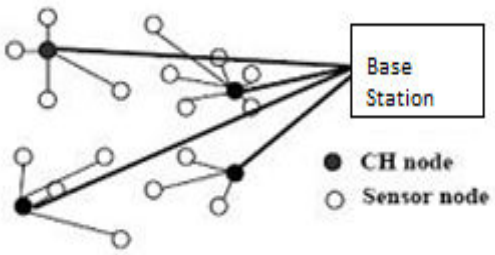


Figure-21. LEACH protocol.

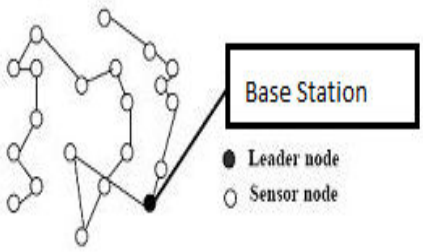


Figure-22. PEGASIS.

PEGASIS (Power efficient gathering sensor information system)

The main idea in PEGASIS is for each node to receive from and transmit to close neighbors and take turns being the leader for transmission to the BS, it shows in Figure-22. In constructing the chain, it is possible that some nodes may have relatively distant neighbors along the chain. Such nodes will dissipate more energy in each round compared to other sensors. Whenever a node dies, the chain will be reconstructed and the threshold can be changed to determine which nodes can be leaders [6].

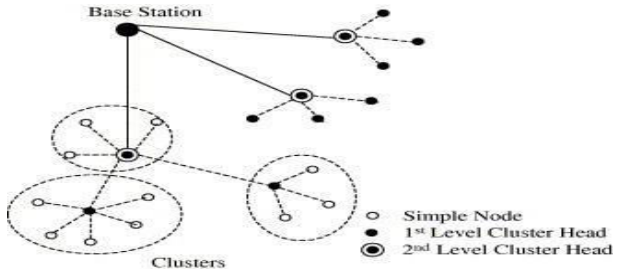


Figure-23. Hierarchical clustering of TEEN and APTEEN.

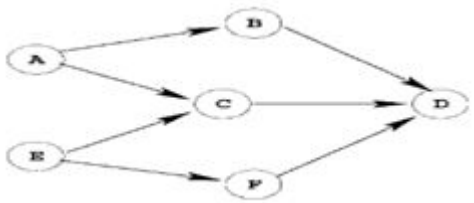


Figure-24. Disjoint paths.

PEGASIS improves on LEACH by saving energy in several stages. First, in the local gathering, the distances that most of the nodes transmit are much less compared to transmitting to a cluster-head in LEACH. Second, the amount of data for the leader to receive is at most two messages instead of 20 (20 nodes per cluster in LEACH for a 100-node network). Finally, only one node transmits to the BS in each round of communication. Data aggregation occurs at all node in the sensor network to pervade all important information across the network. Distributing the energy load among the nodes increases the lifetime and quality of the network. PEGASIS performs better than LEACH by about 100 to 300% when 1%, 20%, 50%, and 100% of nodes die for different network sizes and topologies.

APTEEN (Adaptive threshold sensitive energy efficient sensor network protocol)

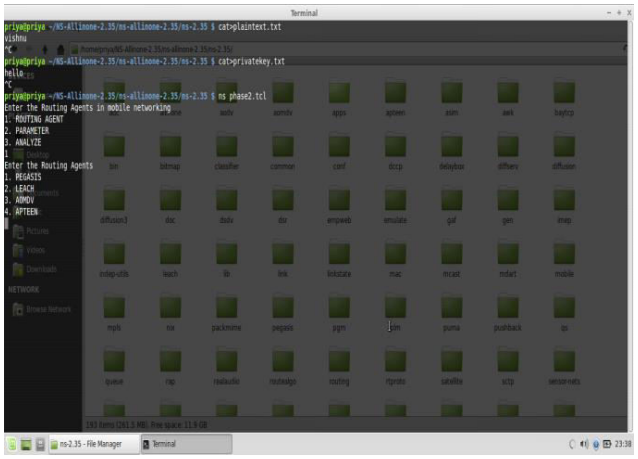
APTEEN is an improvement to TEEN to overcome its short comings and aims at both capturing periodic data collections (LEACH) and reacting to time-critical events (TEEN); it shows in Figure-23. Thus, APTEEN is a hybrid clustering-based routing protocol. APTEEN allows the sensor to send their sensed data periodically and react to any sudden change in the value of the sensed attribute by reporting the corresponding values to their CHs [7]. A TDMA schedule is used and each node in the cluster is assigned a transmission slot. So APTEEN is a hybrid protocol that is both proactive and reactive.

AOMDV (Ad-Hoc on-demand multipath distance vector routing protocol)

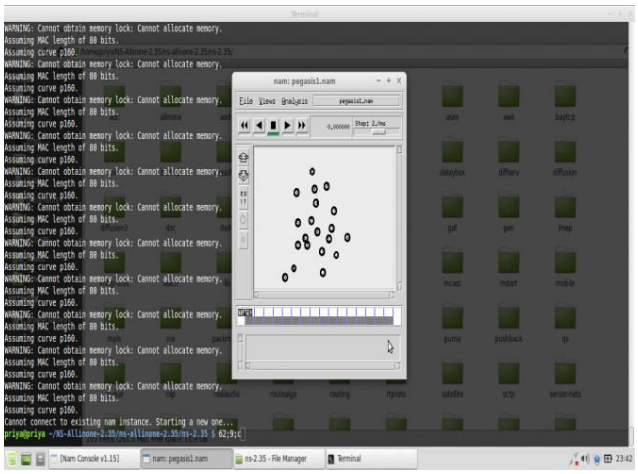
AOMDV (Ad-Hoc On-Demand Multipath Distance Vector Routing Protocol) extends the AODV protocol to discover multiple paths between the source and the destination in every route discovery. Multiple paths so computed are guaranteed to be loop free and link disjoints. AOMDV also finds routes on-demand using a route discovery procedure. AOMDV can be used to find node-disjoint or link-disjoint routes. To find node-disjoint routes, each node does not immediately reject duplicate RREQs [8]. Each RREQs arrive in via a different neighbor of the source defines a node disjoint path. This is because nodes cannot be broadcast duplicate RREQs, so any two RREQs arriving at an intermediate node vie a different neighbor of source could not have traversed the same node. Paths maintained at different nodes to a destination may not be mutually disjoint [9]. Here D is the destination. In Figure-24, Node A has two disjoint paths to D: A - B - D and A - C- D. Similarly, node E has two disjoint paths to D: E - C - D and E - F - D. But the paths A - C - D and E - C - D are not disjoint; they share a common link C - D. AOMDV protocol describe in four components: routing table, route discovery, route maintenance and data packet forwarding.

EXPERIMENT RESULT AND ANALYSIS

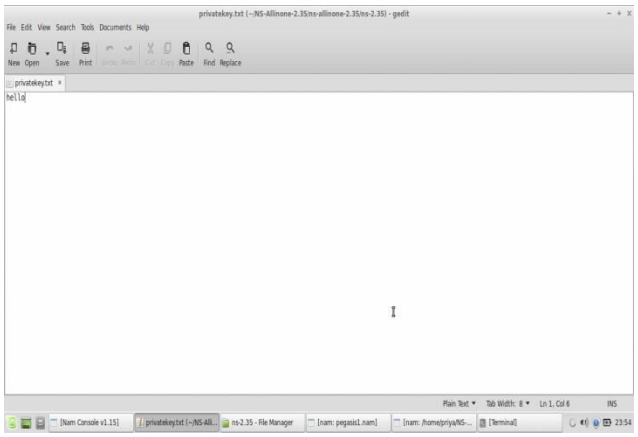
In this work data has encrypted and decrypted using ECC asymmetric algorithm.



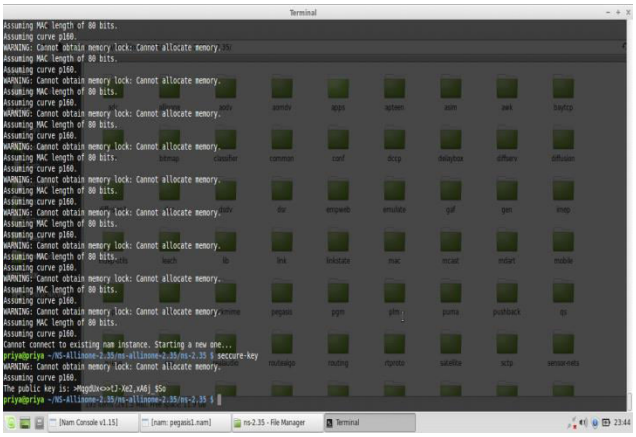
The plain text and private key has given initially.



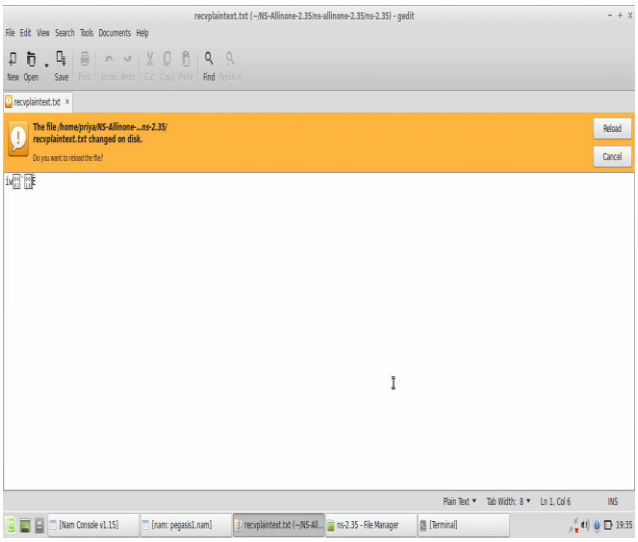
Data has sent with keys to the network.



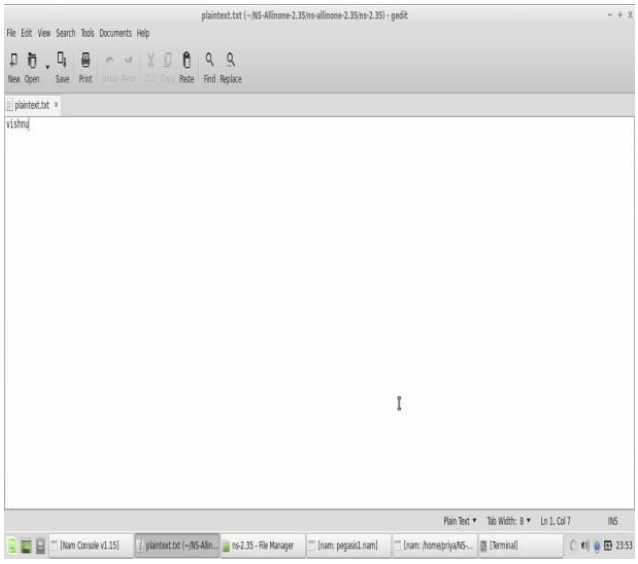
Given secret key.



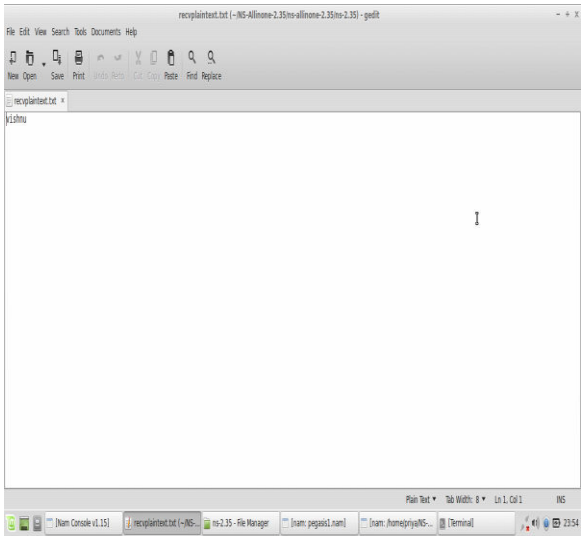
The Secret key has generated using Algorithm.



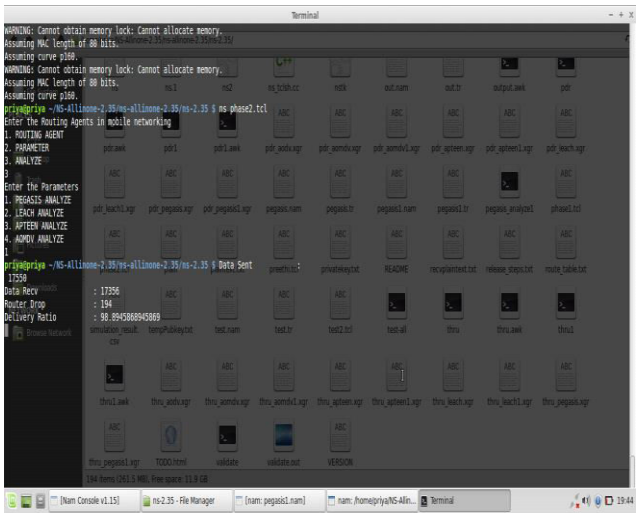
Encrypted data.



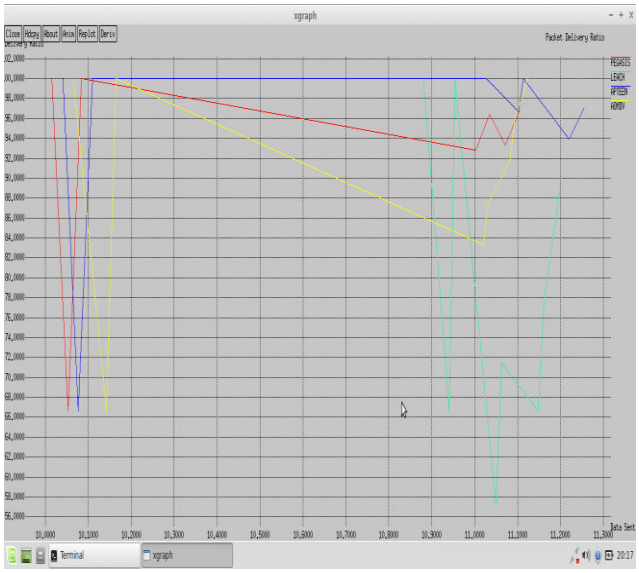
Given plain text.



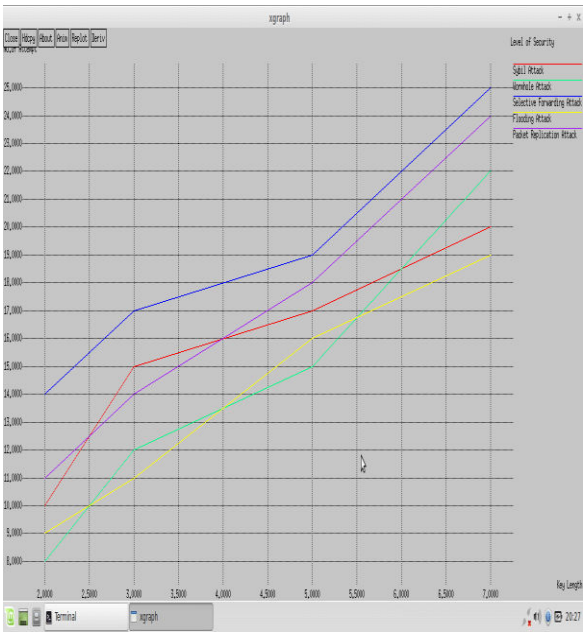
Decrypted plain text.



Data analysis in terms of data sent, data received and delivery ratio.



Packet Delivery Ratio= Total No.of Packets Recieved
Total No.of Packets Sent



Level of security.

CONCLUSIONS

The wireless sensor networks continue to grow and become widely used in many applications. So, the need for security becomes vital. The data transmissions should be improved in a preventative manner to avoid possible attacks. The algorithm based on elliptic curves have been extensively studied in academia as an alternative to RSA, and result show that it is possible to achieve good results with smaller keys. Performance evaluation and simulation is done by using Network Simulation (NS2). From the result it is proved that ECC asymmetric algorithm is the best one for the secure data transmission. Encryption and decryption has been evaluated in terms of data delivery ratio and level of security. Delivery ratio can be achieved 85% using ECC algorithm. Level of security up to the level compared to other asymmetric and symmetric algorithms. In future, the cryptography requirements can be evaluated using some other software.

REFERENCES

[1] Suraj Sharma and Sanjay Kumar Jena. 2011. A Survey on Secure Hierarchical Routing Protocols in Wireless Sensor Networks. ICCCS. ACM 978-1-4503-0464-1.

[2] Shanta Mandal and Rituparna Chaki. 2012. A Secure Encryption Logic for Communication in Wireless Sensor Networks. IJCIS. 2(3).

[3] Gustavo S. Quirino, Admilson R.L. Ribeiro and Edward David Moreno. 2012. Asymmetric Encryption in Wireless Sensor Networks. Licensee in Tech.

[4] A.A.F. Loureiro, J.M.S. Nogueira, L.B. Ruiz, R.A. de Fretas Mini, E.F. Nakamura, and C.M.S. Figueiredo. 2003. Redes de sensores sem fio. SBRC.

[5] Gaurav Sharma, Suman Bala and Anil K. Verma. 2012. Security Frame works for Wireless Sensor Networks-Review. ICCCS. 978-987.

[6] Vishnupriya E, Prof. T. Jayasankar and Prof. P. Maheswara Venkatesh. 2015. Performance Analysis of Optimum Routing Protocols in Wireless Sensor Networks. IJISET. 2(2).

[7] ishal Rathod, Mrudang Mehta. 2011. Security in Wireless Sensor Networks: A Survey. Ganpat University Journal of Engineering and Technology. 1(1).

[8] Kalpana Sharma, M K Ghose. 2010. Wireless Sensor Networks: An Overview on its Security. IJCA.

[9] Himani Chawla. 2014. Some Issues and Challenges of Wireless Sensor Networks. IJARCSSE. ISSN-2277 128X.

[10] Hemanta Kumar Kalita and Avijit Kar. 2009. Wireles Sensor Networks Security Analysis. IJNGN. 1(1).

[11] Idrees S. Kocher, Chee-onn Chow, Hiroshi Ishii and Tanveer A. Zia. 2013. Threat Models and Security Issues in Wireless Sensor Networks. IJCTE. 5(5).