The Islamic University of Gaza

Deanery of graduate studies

Faculty of Engineering

Computer Engineering Department

Secure and Efficient Connectionless Multicast Scheme for Wireless Sensor Network using IBE

Prepared by

Fuad S. H. Abuowaimer

Supervisor: Professor Hatem Hamad

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master in Computer Engineering

2011 - هـ 1432

Abstract

Despite much research effort key distribution in Wireless Sensor Networks (WSNs) still remains an open problem. As sensor networks edge closer towards wide-spread deployment, security issues become a central concern. The characteristic of WSNs such as power limitations, computation capability and storage resources make the development of efficient security scheme a great challenge especially for multicast applications.

In connectionless multicast, the source explicitly encodes the list of destinations in the connectionless header and then sends the data packet to a router. These connectionless multicast protocols like xcast [51] in MANET and uCast (Unified Connectionless Multicast) [1] in WSNs are designed for small networks. they do not keep any state information relevant to ongoing multicast deliveries at intermediate nodes. All secure multicast scheme are designed for connection originated multicast are based on group creation and management making. This design makes it inapplicable to be applied for connectionless multicast because it does not create groups. This means that there is no secure connectionless multicast scheme designed for WSNs till now.

This thesis presents a secure and efficient connectionless multicast scheme in WSNs using identity based encryption (IBE). In proposed solution, each node in the network can request a secure communication with a group of node from a base station. The base station will be responsible for creating and sending the session key. Only nodes in the multicast group will receive and use the session key to establish a secure communication between them.

نظام آمن وفعال لآليات عدم الارتباط في عملية الإرسال المتعدد في شبكات الاستشعار اللاسلكية باستخدام نظام تشفير الهوية

ملخص الرسالة

على الرغم من الجهود الكبيرة المبذولة في الأبحاث حول شبكات الاستشعار اللاسلكية (Wireless Sensor Networks WSNs) تبقى الأبحاث حول موضوع توزيع المفاتيح(Key Distribution) المستخدمة في عملية تشفير البيانات بين مجسات الاستشعار بحاجة إلى المزيد من الدراسة والتطوير. ومع اتساع مجالات استخدام شبكات الاستشعار اللاسلكية أصبح من الضروري زيادة مستوى الأمان من ناحية تشفير البيانات في هذه الشبكات، بشكل عام، تعتبر عملية الإرسال المتعدد من العمليات الأساسية في تطبيقات شبكات الاستشعار اللاسلكية. وعند استخدام آلية عدم الارتباط في عملية الإرسال المتعدد، يقوم المرسل بإضافة قائمة بعناوين الجهات المستقبلة وفي نفس ترويسة رزمة البيانات ومن ثم يقوم بإرسالها إلى جهاز التوجيه. البروتكولات المستخدمة في آلية عدم الارتباط للإرسال المتعدد مثل بروتكول Xcast و uCast مصممة للشبكات الصغيرة ولا تحتفظ بمعلومات حول حالة عمليات الإرسال المتعدد الجارية بين نقاط الاتصال في الشبكة. وتكمن المشكلة حالياً في عدم وجود انظمة آمنة لآلية عدم الارتباط للإرسال المتعدد وحتى البروتكولات الآمنة التي تم تصميمها للعمل ضمن آلية الارتباط للإرسال المتعدد فإنها لا تتناسب مع آلية عدم الارتباط للإرسال المتعدد.

هذه الاطروحة تعرض نظام آمن لآلية عدم الارتباط للإرسال المتعدد في شبكات الاستشعار اللاسلكية باستخدام نظام تشفير باعتماد الهوية (identity based encryption IBE) . يتيح الحل المقترح لأي جهاز ضمن شبكة الاستشعار اللاسلكية إمكانية الحصول على اتصال آمن مع مجموعة من نقاط شبكة الاستشعار اللاسلكية من خلال الوحدة الرئيسية، بحيث تكون الوحدة الرئيسية هي المسئولة عن إنشاء وإرسال مفاتيح تشفير البيانات. و بذلك ستتمكن الوحدات التابعة لنفس مجموعة الاتصال من استقبال واستخدام مفتاح تشفير البيانات وإنشاء اتصال آمن فيما بينها.

Dedication

To whom I love

Acknowledgment

My thanks to all those who generously contributed their favorite recipes. Without their help, this work would have never been possible.

Table of Contents

List of Figures

Chapter 1 - Introduction

Wireless Sensor Networks (WSN) consists of small devices called sensor nodes with RF radio, processor, memory, battery and sensor hardware. One can precisely monitor the environment with widespread deployment of these devices. Sensor nodes are resource-constrained in terms of the RF radio range, processor speed, memory size and power. Sensors used to monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants, at different locations [10, 12].

The development of wireless sensor networks was originally motivated by military applications such as battlefield surveillance. However, wireless sensor networks are now used in many civilian application areas, including environment and habitat monitoring, healthcare applications, home automation, and traffic control. The cost of sensor nodes is ranging from hundreds of dollars to a few cents, depending on the size of the sensor network and the complexity required of individual sensor nodes. Size and cost constraints on sensor nodes result in corresponding constraints on resources such as energy, memory, security overhead, computational speed and bandwidth.

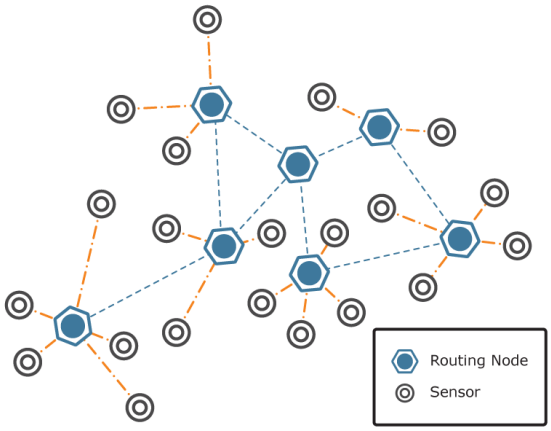

Figure 1.1. Wireless sensor network

A sensor network normally constitutes a wireless ad-hoc network, meaning that each sensor supports a multi-hop routing algorithm, figure 1.1 shows sensor network example.

Multicast communication reduces overhead of the sender as well as the network medium [6]. Existing multicast protocols in WSN are often designed in a P2P pattern or tree pattern, P2P assuming small number of destination nodes and frequent changes on network topologies [5]. There are two categories of multicast in WSN: connection-based and connectionless protocols. Connection-based multicast protocols are efficient for large groups. On the other side, connectionless multicast is efficient for small groups.

A lot of techniques used to secure multicast communication such as Logical Key Hierarchy (LKH), Steiner-based Hierarchical Secure Multicast Routing Protocol (SHSMRP) and other techniques are agreed on providing security based on creating tree take in its consideration of multicast connection group and membership changes such as join and

leave taking place, the rekeying procedure is invoked to update the keys along the path with different behaviors. But these techniques are not suitable for connectionless multicast because there is no group creation and membership changes status [35, 36].

1.1. Thesis Statement

This thesis discusses the effective design of security in connectionless multicast protocols in WSNs. The main idea is to provide secure and efficient scheme that is adaptive with connectionless multicast behavior in WSNs. This will enable applications that use connectionless multicast protocols to exchange the sensitive information with each other in secure manner. The ideas presented in this thesis are the author's original works. The implementations and results are also accurate and were obtained solely by the author.

1.2. Background

This thesis builds upon the fact that connectionless multicast protocols are becoming important and accepted solution in both MANET and WSNs as small group communications [1, 51, 14, 43]. Connectionless multicast focuses on small multicast groups and assumes the underlying unicast protocol takes care of forwarding the packets. Source in connectionless multicast explicitly encodes the list of destinations in the header and then sends the data packet to next hop. Each router along the way parses the header and forwards a packet with an appropriate header to each of the next hops. A connectionless multicast also had known as Peer-to-Peer (P2P) protocols multicast patterns. Unlike connection based multicast protocol classifies according to the global

data structure used to forward multicast packets either to tree or mesh-based. Those protocols maintain distributed forwarding states in each node on the multicast path that should be updated via periodic control flooding messages. However, due to the capacity limitations and the control processing overheads, they may not be the most efficient and scalable choices for WSNs. The detailed description of currently efficient multicast protocols in sensor networks can be found in [30, 31, 32, 33, 34].

1.3. Problem Statement

The current research activities for security in multicast protocols in Wireless sensor networks (WSNs) have mostly been concentrating on connection based multicast security using one of three classes of key agreement schemes: public key, trusted server and key pre-distribution schemes. But no one discuss the connectionless multicast security, and all security mechanism used in connection based multicast cannot be used for connectionless because it's based on connection based behavior of creation secure group using tree or other mechanisms to manage key agreement and security, The detailed description of currently secure connection based multicast in sensor networks can be found in [2, 3, 15, 7, 8]. A few of papers discuss the efficiency of using connectionless multicast in ad hoc and WSNs for small group communication [1, 51]. This thesis provide secure and efficient scheme for connectionless multicast protocols in WSNs.

In This thesis we present a secure and efficient connectionless multicast scheme in WSNs. In our solution we use uCast protocol as connectionless protocol for testing our scheme and every sensor node has its own IBE public key and private key. Before sending a multicast message, a node requests the base station to generate a random group

session key $\mathbf{k_G}$ for destination group. The base station generates a random group session key $\mathbf{k_G}$ and sends the session key encrypted with public key of sender with list of destination group encrypted with public key of each node with same requested order. So only nodes in the multicast group will be able to receive and use the session key to establish a secure communication between them. A full description of this scheme is presented in chapter 4.

The rest of this thesis is organized as follows. In chapter 2, the Preliminary Discussions is overviewed. In chapter 3, the related work is overviewed. In chapter 4 our proposed solution is presented. In chapter 5 the analysis of our scheme is presented. In chapter 6 Simulation and Results is presented. Then the conclusion is given in chapter 7.

Chapter 2 – Preliminary Discussions

In this chapter, we review the basic concepts behind this thesis. Mainly speaking, we will discuss the concepts of multicast, security requirement and Key management in WSNs. For each concept, a quick and comprehensive review will be made. This includes the main concepts, the advantages, and the types available that distinguishes their usage in practice.

2.1 Multicast

Multicast is a mechanism of message delivery which reduces the overall overhead traffic in network by allowing sender to send message to group of destination in one delivery packet rather sending multiple unicast messages. There are two categories of multicast protocols: connection-based and connectionless protocols.

2.1.1 Connection-based multicast

The connection based multicast protocol is classified according to the global data structure used to forward multicast packets either to tree- or mesh-based. Many state full multicast routing protocols including MAODV [37], ADMR [38], ODMRP [39], AMRoute [40], AMRIS [41], and PAST-DM [42] are proposed for WSN multicast services. Those protocols maintain distributed forwarding states in each node on the multicast path that should be updated via periodic control flooding messages. They have been originally designed for the traditional wireless ad hoc networks as control-centric approaches focused on solving the mobility issues under the assumption of enough

processing and local storage capacity on each node. However, due to the capacity limitations and the control processing overheads, they may not be the most efficient and scalable choices for WSNs. The detailed description of currently efficient multicast protocols in sensor networks can be found in [30, 31, 32, 33, 34].

2.1.2 Connectionless multicast

Connectionless multicast focus on small multicast groups and assumes the underlying unicast protocol takes care of forwarding the packets. In connectionless multicast, the source explicitly encodes the list of destinations in the connectionless header and then sends the data packet to a router. Each router along the way parses the header, partitions the destinations addresses based on each destination's next hop and forwards a packet with an appropriate connectionless header to each of the next hops. A connectionless multicast also had known as Peer-to-Peer (P2P) protocols multicast patterns.

2.1.2.1 Ad hoc networks

As example in connectionless multicast communication in ad hoc network is explicit multicast. In Xcast [51], the source explicitly encodes the list of destinations in the Xcast header and then sends the data packet to a router. Each router along the way parses the header, partitions the destinations addresses based on each destination's next hop and forwards a packet with an appropriate Xcast header to each of the next hops. The detailed description of each work in connectionless multicast in ad hoc networks can be found in [44, 45, 46].

2.1.2.2 Wireless sensor Networks

Various stateless multicast protocols including the source multicast routing (SMR) approaches such as DSM [47], PBM [48], and AGSMR [51], and the location-based approaches such as LGT [49] and GMR [50] have been proposed for WSNs to perform a centralized membership management on the multicast root instead of having distributed states. One of the best connectionless multicast protocols is uCast. We describe brief mechanism of this protocol in section 2.1.2.3 because we focused on it in our work for testing our scheme.

2.1.2.3 uCast protocol

uCast protocol is a connectionless multicast protocol for energy efficient content distribution in sensor networks and it is designed to support a large number of multicast sessions, especially when the number of destinations in a session is small. uCast does not keep any state information relevant to ongoing multicast deliveries at intermediate nodes. It directly encodes the multicast information in the packet headers and parses these headers at intermediate nodes using a scoreboard algorithm, the detailed description of uCast protocol can be found in [1].

2.2 Security Requirements

security requirement is things must be taking in consideration in designing any secure scheme the more security requirement you satisfy in your solution the more secure you scheme become. We present some security requirements need to be to observe security in our scheme.

2.2.1 Data Confidentiality

Confidentiality requirement is needed to ensure that sensitive information is well protected and not revealed to unauthorized third parties.

2.2.2 Authentication

Authentication techniques verify the identity of the participants in a communication, distinguishing in this way legitimate users from intruders. An adversary is not just limited to modifying the data packet.

2.2.3 Integrity

Data integrity ensures that any received data has not been altered in transit. There is the danger that information could be altered when exchanged over insecure networks.

2.2.4 Data Freshness

Data freshness implies that the data is recent, and it ensures that no adversary replayed old messages.

2.2.5 Robustness and Survivability

The sensor network should be robust against various security attacks, and if an attack succeeds, its impact should be minimized. The compromise of a single node should not break the security of the entire network.

2.3 Basics of IBE

The concept of identity-based cryptography was first proposed in 1984 by Adi Shamir [20]. In his paper, Shamir presented a new model of asymmetric cryptography in which the public key of any user is a characteristic that uniquely identifies himself/herself, like an e-mail address. In such a scheme there are four steps: (1) **setup** generates global system parameters and a master-key, (2) **extract** uses the master-key to generate the private key corresponding to an arbitrary public key string ID $\in \{0, 1\}^*$ (3) **encrypt** encrypts messages using the public key ID, and (4) **decrypt** decrypts messages using the corresponding private key.

Shamir's original motivation for identity-based encryption was to simplify certificate management in e-mail systems. When Alice sends mail to Bob at bob@company.com she simply encrypts her message using the public key string “bob@company.com”. There is no need for Alice to obtain Bob's public key certificate. When Bob receives the encrypted mail he contacts a third party, which we call the Private Key Generator (PKG). Bob authenticates himself to the PKG in the same way he would authenticate himself to

a Center of Authentication (CA) and obtains his private key from the PKG. Bob can then read his e-mail. Note that unlike the existing secure e-mail infrastructure, Alice can send encrypted mail to Bob even if Bob has not yet setup his public key certificate. Also note that key escrow is inherent in identity-based e-mail systems: the PKG knows Bob's private key.

The distinguishing characteristic of identity-based encryption is the ability to use any string as a public key. The functions that compose a generic IBE are thus specified as follows.

Setup: takes security parameter t_s and returns t_g (system parameters) and master-key. The system parameters include a description of a finite message space M, and a description of a finite cipher text space C. Intuitively, the system parameters will be publicly known, while the master-key will be known only to the Private Key Generator (**PKG**).

Extract: takes as input t_g, master-key, and an arbitrary ID $\in \{0, 1\}^*$, and returns a private key K. Here ID is an arbitrary string that will be used as a public key, and K is the corresponding private decryption key. The Extract algorithm extracts a private key from the given public key.

Encrypt: takes as input t_g, ID, and $m \in M$. It returns a cipher text $c \in C$.

Decrypt: takes as input t_g, $c \in C$, and a private key K. It returns $m \in M$. These algorithms must satisfy the standard consistency constraint, namely when K is the

private key generated by algorithm Extract when it is given ID as the public key, then ∀ m ∈ M: Decrypt(t_g, c, K) = m where c = Encrypt(t_g, ID, c)

2.3.1 The Boneh-Franklin IBE scheme

The scheme is based on IBE technique and proposed by Boneh and Franklin [23]. We use $\mathbf{Z_q}$ to denote the group {0, …, q-1} under addition modulo q. For a group G of prime order we use G^* to denote the set G* = G\O where O is the identity element in the group G. We use Z^+ to denote the set of positive integers. We describe first some definitions and then the Boneh-Franklin IBE scheme.

Definition 1 An map **ê:** $G_1 \times G_1 \rightarrow G_2$ is called a bilinear pairing if, for all x, y ∈ G_1 and all a, b ∈ Z, we have ê (x^a, y^b) = ê $(x, y)^{ab}$.

Definition 2 The Bilinear-Diffie-Hellman problem (BDH) for a bilinear map ê: $G_1 \times G_1 \rightarrow$G2 such that $|G_1|=|G_2|$=q is prime is defined as follows: given g, g^a, g^b, g^c ∈ G_1, compute ê$(g, g)^{abc}$, where g is a generator and a, b, c ∈ Z. An algorithm A is said to solve the BDH problem with advantage ε if Pr [A (g, g^a, g^b, g^c)= ê$(g, g)^{abc}$] ≥ ε ,

Where the probability is over the random choice of a, b, c, g, and the random bits of *A*.

Definition 3 A randomized algorithm G that takes as input a security parameter k ∈ Z^+ is a BDH parameter generator if it turns in time polynomial in k and outputs the description of two groups G_1, G_2 and a bilinear function ê: $G_1 \times G_1 \rightarrow G_2$, with $|G_1|=|G_2|$=q for some prime q. Denote the output of the algorithm by G(1^k)=< G_1, G_2, ê, q>.

Definition 4 we say that G satisfies the BDH assumption if no probabilistic polynomial algorithm *A* can solve BDH with non-negligible advantage.

We now give the Boneh-Franklin IBE algorithm for identity-based encryption based on bilinear pairings on elliptic curves.

2.4 Key management in wireless sensor networks

Key management is the process by which cryptographic keys are generated, stored, protected, transferred, loaded, used, and destroyed. The general key distribution problem refers to the task of distributing secret keys between communicating parties to provide security properties such as secrecy and authentication. In sensor networks, key distribution is usually combined with initial communication establishment to bootstrap a secure communication infrastructure from a collection of deployed sensor nodes. In this chapter we will discuss and evaluate several well-known methods of key distribution. Besides these, we present an in-depth study of the trusted base station distribution, a method that we have worked on.

2.4.1 Using a Single Network-Wide Key

The simplest method of key distribution is to pre-load a single network-wide key onto all nodes before deployment. After deployment, nodes establish communications with any neighboring nodes that also possess the shared network key. This can be achieved simply by encrypting all communications in the shared network-wide key and

appending a message authentication code (MAC) to ensure integrity. Many advantages in using single key because minimal memory storage required and no additional protocol steps are necessary. The main drawback of the network-wide key approach is that the compromise of a single node causes the compromise of the entire network, since the network-wide key is now known to the adversary.

2.4.2 Using Asymmetric Cryptography

The favored method of key distribution in most modern computer systems is via asymmetric cryptography, also known as public key methods. If sensor node hardware is able to support the computationally intensive asymmetric cryptographic operations, then this is a potentially viable method of key distribution. A brief outline of a possible public-key method for sensor networks is as follows.

Prior to deployment, a master public/private key-pair, (K_M, K_M^{-1}) is first generated. Then, for every node A, its public/private key-pair (K_A, K_A^{-1}) is generated. This key-pair is stored in node A's memory along with the master public key K_M and the master key's signature on A's public key. Once all the nodes are initialized in this fashion, they are ready for deployment. Once the nodes have been deployed, they perform key exchange. Nodes exchange their respective public keys and master key signatures. Each node's public key is verified as legitimate by verifying the master key's signature using the master public key. Once the public key of a node has been received,

a symmetric link key can be generated and sent to it, encrypted by its public key. Upon reception of the session key, key establishment is complete and the two nodes can communicate using the symmetric link key [4]. The advantage of this method is that, it is perfectly resilient against node capture and it is possible to revoke known compromised key-pairs. However, using asymmetric cryptography has its disadvantages because the dependence on asymmetric key cryptographic hardware or software and it is vulnerability to denial-of-service and it has no resistance against node replication [9].

2.4.3 Using Pairwise-shared Keys

In this approach, every node in the sensor network shares a unique symmetric key with every other node in the network. Hence, in a network of n nodes, there are a total of $\binom{n}{2}$ unique keys. Every node stores n-1 keys, one for each of the other nodes in the network. After deployment, nodes must perform key discovery to verify the identity of the node that they are communicating with. The advantage of this method is that, it is perfect resilience to node capture and it is compromised keys can be revoked [4].

The main problem with the pair-wise keys scheme is poor scalability. The number of keys that must be stored in each node is proportional to the total number of nodes in the network. With an 80 bit key, a network with 100 nodes will require almost 1kB of storage on each node for keys alone. Assuming memory-constrained sensor nodes, the pair-wise keys scheme would not scale to large sensor networks. In addition, adding new nodes may also be a challenge in this setting [13].

2.4.4 Using Trusted Base Station

This method of key distribution uses a trusted, secure base station as an arbiter to provide link keys to sensor nodes. The sensor nodes authenticate themselves to the base station, after which the base station generates a link key and sends it securely to both parties. Prior to deployment, a unique symmetric key is generated for each node in the network. This node key is stored in the node's memory and will serve as the authenticator for the node as well as facilitate encrypted communications between the node and the base station. The base station has access to all the node keys either directly (they are stored in its memory) or indirectly (the base station relays all communications to a secured workstation off site). This method, unlike the other methods mentioned previously, assumes some level of reliable transport is available between the node and the base station before any key establishment has taken place. Since this transport occurs before any security primitives are in place, it will necessarily have to be assumed as insecure, however, as long as it is reliable in a way such that a small number of malicious nodes are unable to prevent the transmission of messages to and from the base station then the protocol presented here is viable. Now assume that after deployment, node A wants to establish a shared secret session key SK_{AB} with node B. Since A and B do not share any secrets, they need to use a trusted third party S, which is the base station in our case [4]. The properties of this method of key establishment are as follows.

- Small memory requirement.

For every node, a single secret symmetric key shared with the base station is needed, as well as one unique link key for each one of its neighbors.

- Perfect resilience to node capture.

Any node that is captured divulges no secret information about the rest of the network.

- Revocation of nodes is simple.

Since no link keys can be established without the direct involvement of the base station, the base station has a record of all nodes that have established a link key with any given node. If a node is to be revoked, the base station securely transmits the revocation message to all the nodes that may be in communication with the revoked node.

However, key establishment through a base station has its disadvantages, as follows.

- Significant communication overhead.

If any two nodes wish to establish a secure communications, they must first communicate directly with the base station. In a large network, the base station may be many hops away, thus incurring a significant cost in communication.

- The base station becomes a target for compromise.

Since the base station has access to all the secret node keys in the sensor network, compromise of the base station's key store will expose the secrecy of all links that are established after the time of the compromise. This may not be a problem if the

communications base station merely acts as a gateway to a workstation at a remote, secured site, since the adversaries would have to successfully attack the secure workstation in order to gain the node keys. Since that problem a lot of research study conducted to find out a practical way to use Public-Key Cryptography (PKC) in sensor networks [16, 17, 18, 19]. Their studies focus mostly on optimization of PKC. Though computing cost is still a crucial problem for PKC system, results in [17] indicate that Elliptic Curve Cryptography (ECC) has some advantages in memory requirement and computing cost and that it is suitable for sensor networks. In 1984 Shamir proposed the idea of Identity-Based Encryption (IBE) [20]. The idea of an identity-based encryption is that the public key can be an arbitrary string, for example, an email address, a name or a role. Soon after, various identity-based techniques were proposed [21, 22] but a fully-functional identity-based encryption scheme was not found until recently by Boneh and Franklin [23]. Since then the ideas of IBE have been used to design several other identity-based schemes for different purposes [24,25,26,27]. Note that IBE-based algorithms are types of ECC. According to the studies about public key system, therefore, it is interesting to investigate the possibility to apply IBE in wireless sensor networks. Table 2.1 summarizes the main advantages of IBC when compared with other security schemes. It also shows our main motivations behind choosing an ID-based mechanism to ensure secure communication in sensor networks.

Table 2.1 IDENTITY BASED CRYPTOGRAPHY VS OTHER SECURITY SCHEMES FOR WSNS

	Symmetric Key Cryptography	**Public key cryptography**	**Identity based cryptography**
Computational complexity	**Low**	**High**	**High**
Communication overhead	**Low**	**High**	**Low**
Key distribution	**problematic**	**complex**	**Simple**
Key directory	**$O(n^2)$**	**O(n)**	**O(n)**
Non-repudiation	**No**	**Yes**	**Yes**
Forward encryption	**No**	**No**	**Yes**

Chapter 3 – Related Work

In this chapter, we review some of the significant and recent research papers in the field of multicast protocols and secure mechanism used in it. We present these activities and discuss their advantages and the disadvantages.

Security in sensor networks is very important issue and a lot of researches focus on how to secure a sensor network. Such as Ghosh, S.K. et. Al. , in ” **secure group communication for wireless sensor networks (WSNs)**” [15] address the problem of formation secure group in WSNs with low communication complexity and providing an efficient solution to maintain such multicast group the main goal of this paper is how to form secure groups by mechanisms of group formation and discovery with little overhead and maintain such groups, so only the intended recipients of the group can receive and send data. The disadvantage is that we cannot apply this solution to connectionless multicast protocol to secure it because the connectionless multicast protocol behavior prevents group creation nor discovery, so this solution fails in securing connectionless multicast protocols.

Rong Fan, et. Al. in “**A Steiner-Based Secure Multicast Routing Protocol for Wireless Sensor Network**” [36] propose Secure Multicast Routing Protocol for wireless sensor network, which is an energy-efficient and secure protocol for multicast in the WSNs based on a Steiner tree, partitioned Steiner sub-trees, and clusters to minimize

the number of multicast packets transmitted in WSN for reducing the overall cost of the transmission to all destinations. Then they design a new logical key hierarchy based on LKHW which based on group creation and management. The disadvantage is that we cannot apply this solution to connectionless multicast protocol to secure it because the connectionless multicast protocol behavior prevents group creation nor discovery, so this solution fails in securing connectionless multicast protocols.

Other Multicast Encryption Schemes: In [29], GKMPAN was proposed to address secure multicast in ad hoc networks. GKMPAN assumes that all nodes in an ad hoc network are pre-distributed with a certain number of keys m randomly out of a big pool of l keys, which are used to update group keys. If a node is compromised, the key server first determines a non-compromised key, which is the most common among the remaining members of the group. Then, the key server broadcasts a new group key encrypted with the chosen non compromised key. Consequently, nodes that have this key can independently decrypt the group key. These nodes further re-encrypt the new group key with another non compromised key and forward it to those neighbors yet to obtain it. In this way, the new group key is propagated to all the members in a hop-by-hop fashion. However, GKMPAN is vulnerable to the selective node compromise attack. The disadvantage is that creating pool of keys or using symmetric infrastructure in key exchange is weak compare with using Asymmetric key.

Chapter 4 – secure and efficient connectionless multicast scheme for WSNs using IBE

In this chapter we present our proposed solution of providing a secure and efficient connectionless multicast scheme for WSNs. We design efficient scheme for key management and using Boneh-Franklin IBE algorithm for encryption and decryption. We test this scheme on uCast protocol as connectionless multicast protocol.

4.1 overview

The proposed solution is built upon the fact that connectionless multicast protocol becomes important and accepted solution in wireless network such as MANET and WSNs as small group communication. As that important security issues become main concern in how to provide secure communication between group of nodes to exchange sensitive information. A lot of research focus on how to secure connection based multicast scheme but that solution contract with connectionless multicast protocol behavior that prevent group creation nor discovery.

The proposed solution presents a secure and efficient connectionless multicast scheme in WSNs and is capable of providing secure group communications in connectionless multicast scheme, it presents novel mechanism in key management to distribute session key among nodes in connectionless multicast groups. Before sending a multicast message, a node requests the base station to generate a random group session key $\mathbf{k_G}$ for destination group. The base station generates a random group session key $\mathbf{k_G}$ and sends the session key encrypted with public key of sender with list of destination group

encrypted with public key of each node with same requested order. So only nodes in the multicast group will be able to receive and use the session key to establish a secure communication between them. uCast protocol is used as connectionless protocol for testing this scheme. A full description of the scheme is presented throw this chapter.

4.2 Boneh-Franklin IBE algorithm

In this section we present details of Boneh-Franklin's algorithm steps needed to a node to get its public and private key and how encryption and decryption are done using Boneh- Franklin IBE algorithm.

Algorithm 4.1 shows the steps needed to generate system public parameters and the master key. The master key should be kept in a secure place, but the public parameters π can be distributed to all nodes. This phase should be done prior to the nodes deployment. We use a base station to run the setup function and distribute all the parameters to nodes.

Algorithm 4.1 Boneh-Franklin IBE Setup

INPUT: a security parameter k $\in Z^+$, an elliptic curve E, a plaintext bit length n

OUTPUT: public system parameter π = {q, G_1, G_2, ê, n, α, β, H_1, H_2, H_3, H_4}, The master key is s $\in Z_q^*$.

Step 1: Run G on input k to generate a prime q, two groups G_1, G_2 of order q, and an admissible bilinear map ê : $G_1 \times G_1 \rightarrow G_2$. Choose a random $\alpha \in G_1$.

Step 2: Pick a random s $\in Z_q^*$ and set $\beta = \alpha^s$.

Step 3: Choose cryptographic hash functions for some n, H_1: $\{0, 1\}^* \rightarrow G_1^*$, H_2: $G_2 \rightarrow \{0, 1\}^n$, H_3: $\{0, 1\}^n \times \{0, 1\}^n \rightarrow Z_q^*$, H_4: $\{0, 1\}^n \rightarrow \{0, 1\}^n$. For the security

proof, we view the all hash functions as random oracles. The message space is M= {0, 1}^{n}. The cipher text space is C = $G_1^* \times$ {0, 1}*.

Algorithm 4.2 runs the Extract function of Boneh-Franklin Algorithm to obtain private keys. The inputs are public parameters obtained from algorithm 4.1 and a String ID represents an identity. The public key could be an arbitrary string Id∈{0, 1}*. The private key generated by this algorithm will be distributed to a sensor, this phase should be done before the nodes deployment. The base station is used to perform the calculations, so the private key is only known by the base station and the corresponding sensor. The master key ***s*** is known just for the Private key Generator (**PKG**) which in our scheme is the base station.

From the administration point of view, this step could be performed within a scope of users of the sensor networks (for example a military unit, a fire department, a company, etc.). The master key is only stored in the base station of an organization, When a new sensor has to be added to the network, or to be replaced, the administration system completes the initialization process and puts it into the network, this enhances effectively the security of the sensor networks.

Algorithm 4.2 Boneh-Franklin IBE Private key Extraction

INPUT: A string ID representing an identity and public parameter π = {q, G_1, G_2, ê, n, α, β, H_1, H_2, H_3, H_4}.

OUTPUT: The private key K_{Id}.

For a given string Id∈{0, 1}* the algorithm does:
Step 1: Computes $Q_{Id} = H_1$ (Id) ∈ * G_1^* .

Step 2: Sets the private key K_{Id} to be $K_{Id} = (Q_{Id})^s$ where **s** is the master key.

Algorithm 4.3 shows the steps needed to encrypt a message, the input parameters to the algorithm are the system public parameters and the master key, the output of the encryption algorithm is the encrypted message. Once the initialization is completed and a sensor network is deployed, a node can encrypt a message using the public parameters loaded before the deployment of sensor nodes using algorithm 4.3.

Unlike traditional application of public-key infrastructure, a Certification Authority (CA) will be eliminated in identity-based cryptography for sensor networks, and the problem of impersonation will be solved using an identity-based signcryption scheme [24, 25].

Algorithm 4.3 Boneh-Franklin IBE Encryption

INPUT: A plaintext message M of length n bits, a string ID representing the identity of recipient of ciphertext and a set of public system parameter π = {q, G_1, G_2, ê, n, α, β, H_1, H_2, H_3, H_4}.

OUTPUT: A ciphertext C.

Step 1: Compute $Q_{Id} = H_1(Id) \in G_1^*$.

Step 2: Choose a random $\sigma \in \{0, 1\}^n$.

Step 3: Set $r=H_3(\sigma, m)$.

Step 4: Set the cipher text to be

$$C = \langle r\alpha, \sigma \oplus H_2(g_{Id}^r), m \oplus H_4(\sigma) \rangle$$

$$where \ \ g_{Id} = \hat{e}(Q_{Id}, \beta) \in G_2$$

Finally Algorithm 4.4 shows steps that a node use to decrypt an encryption message, the input parameter to this algorithm is ciphertext, public parameters and node's private key. The output is the plaintext of encrypted message.

Algorithm 4.4 Boneh-Franklin IBE Decryption

INPUT: A ciphertext C, a set of public system parameter π = {q, G_1, G_2, ê, n, α, β, H_1, H_2, H_3, H_4} and The private key K_{Id}.

OUTPUT: A plaintext message M or an error condition .

Step 1: Compute $V \oplus H_2$ (ê (K_{Id}, U)) = σ.

Step 2: Compute $W \oplus H_4$ (σ) = m.

Step 3: Set r = H_3 (σ , m). Test that U = r α . If not, reject the ciphertext.

Step 4: Output m as the decryption of c.

4.3 key management

The key management in our approach will be as follows: the base station generates a random group session key $\mathbf{k_G}$ when a secure multicast requested by a node. The base station encrypts $\mathbf{k_G}$ by the node's public key of each node in the multicast group to generate encrypted key list by running the function **Encrypt** of Boneh-Franklin algorithm as described in above section, where the public key is the identity of each node in multicast group ($\mathbf{Encrypt_{K1}}$ (K_G)… $\mathbf{Encrypt_{Kn}}$(K_G)). The encrypted key list is sent back to the sender with $\mathbf{Encrypt_{ks}}$(k_G), then the sender will get k_G by running the Decrypt function Boneh-Franklin algorithm as described in above section with the its private key. The sender will encrypt the message by the group session key $\mathbf{K_G}$ ($\mathbf{E_{KG}}$(Message)), then it will multicast the encrypted message with the encrypted key list using the adapted uCast message format hierarchical shown in Figure 4.1 (b).

Figure 4.1 (a) shows the default uCast message format which destination member is listed before message payload, figure 4.1 (b) shows our modified secure uCast (suCast) message format, in our modified secure uCast message format the session key encrypted with destination public key is appended to each destination header format.

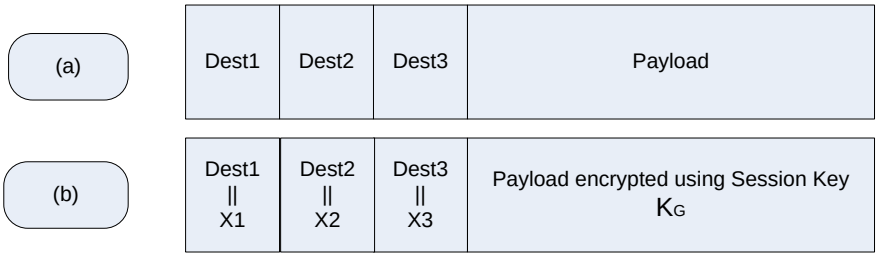

Figure 4.1 (a) uCast, (b) suCast message format.

Secure uCast header contains the list of the multicast group and the session key encrypted by the node's public key of each node in the group. When the secure uCast packet reaches any node member to the multicast group, the receiver node will use its private key to decrypt the session key which was encrypted with its public key, and then the session key is used to decrypt the message payload (data).

Figure 4.2 shows the key management mechanism in our approach, each member node will get k_G by decrypting the received encrypted key by its own secret key. After that, the member node will decrypt the message by k_G. Any node outside the group cannot decrypt the message because it does not have the k_G. If the group has changed, the sender will request a new group key from the base station.

as example of our key management, Figure 4.2 shows that node S wants to send multicast message to group of nodes {1,2,3}, the source node first sends the multicast group list encrypted with the base station's public key, then the base station decrypts the message and generates the group session key K_G , The encrypted key list of multicast group members {1,2,3} is generated by the base station, which uses the public key for each node in multicast group to encrypt the key list. Finally the base station encrypts the session key K_G and send both the encrypted session key and key list to the sender node.

the source node decrypts the message to get the session key by using node's private key, then it encrypts the multicast message with session key K_G and use the adaptive uCast

message format in figure 4.1.b and send the session group key to the group. When the node inside the group receives the message it can decrypt the message and get its session key.

The session key life time of the multicast group will be valid if the multicast group is not changed or the sender requests a new session key from the base station.

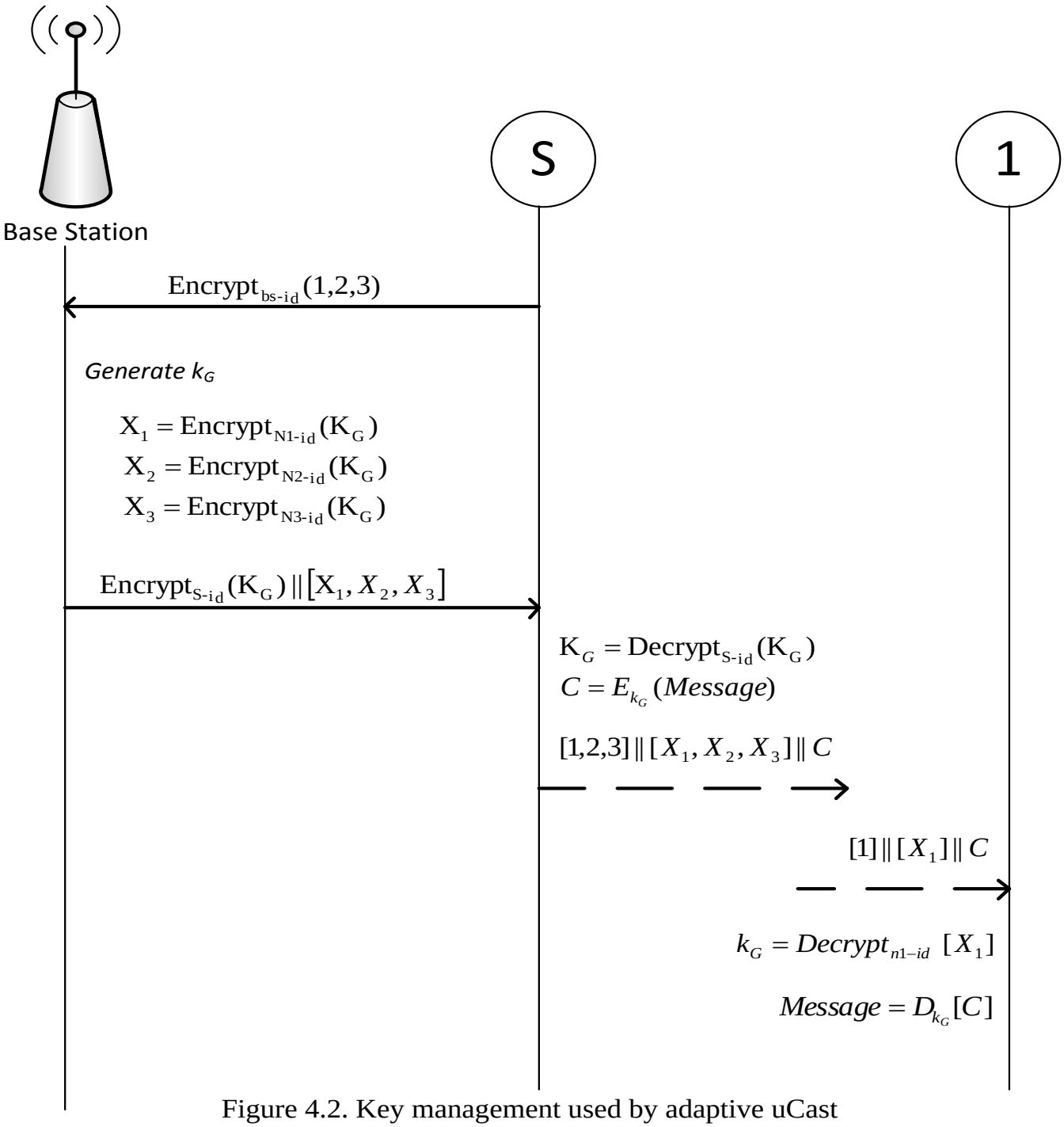

Figure 4.2. Key management used by adaptive uCast

Chapter 5 – Analysis Of Proposed Scheme

In This chapter we focus on analysis of efficiency and security of our scheme. To our knowledge this work is the first secure connectionless multicast protocol in Wireless Sensor Networks (WSNs) so we will direct the analysis to our key management by comparing with other key management schemes and encryption algorithms; we discuss benefits and drawbacks of the scheme in security and efficiency.

5.1 Efficiency Analysis

5.1.1 Comparison with PKI

IBE has some special characteristics and properties compared with PKI. We have

(1) Public keys in IBE are arbitrary strings or “identities”. They can be names, roles, email addresses, etc. This makes it possible for a sender to send a message whenever he wants; while in PKI public keys should be generated and distributed to senders before sending a message. Our key pre-distribution scheme for wireless sensor networks benefits from this property. In fact, we can generate private keys in initialization phase. No key pre-distribution is needed in this case.

(2) Private keys in IBE are derived from the identities by a trusted Private Key Generator (PKG) using a master key, while in PKI both public and private keys are created by users themselves. This gives one reason that why PKI is not considered as a good choice for key agreement and encryption in wireless sensor networks. In a system

with RAS algorithm, an authentication process is executed before establishment of a secure communication, whereas this process is unnecessary in IBE-based algorithms.

(3) the most common criticism on using PKI in sensor networks is its computational complexity and communication overhead. Recently, a number of studies have been conducted to address PKC for sensor networks [18,19,28,29]. For example, Gura et al. show that Elliptic Curve Cryptography (ECC) signature verification takes 1.62s with 160-bit keys on ATmega128 8MHz processor, a processor used for Crossbow motes platform [17]. These results indicate that ECC-based algorithms have some advantages and will soon be available for sensor networks; in despite of comparing with the symmetric key cryptography, PKC is still much more expensive.

As we all known, IBE algorithms are based on ECC. Research results show that the traditional RSA algorithm with 1024-bit key (RSA-1024) provides the currently accepted security level, and is equivalent in strength to ECC with 160 bit keys (ECC-160) and to symmetric key with 80 bit [30]. Therefore, the length of the keys is much shorter than that of the traditional RSA algorithms. As a result, it economizes the storage resources and computing cost.

5.1.2 Comparison with symmetric key encryption

Applications of symmetric key system in wireless sensor networks have been widely investigated. Compared to IBE algorithms, in symmetric key system, an extra key distribution must be performed prior to deployment of a sensor network. Secret keys are stored in nodes after distributing operation. There are two extreme cases in storing secret keys. One is to let each sensor keeps in memory only one secret key (a global master secret key) shared by all nodes in a sensor network. The other is to let each node carry all N-1 secret pair-wise keys, where N is the total number of nodes in a sensor network. Evidently, these two mechanisms are impractical. A random key pre-distribution scheme and its variants are proposed [6,9,10], where at least q keys selected from a key pool are stored in each node. When a node wants to communicate with another node, a key discovery operation should be performed. However, in IBE algorithms, each node stores only public parameters and owner private key. Neither key pre-distribution nor key discovery is needed. At the same time, IBE algorithms with 160 bit keys provide currently a sufficient security level. Therefore, in terms of memory requirement and key discovery in wireless sensor networks, our algorithm has a better performance than symmetric key encryption algorithms. But in encrypting and decrypting operations it seems that symmetric key algorithms offer a better performance in computing cost. A detail comparison could be an interesting future work.

5.2 Security Analysis

5.2.1 Message confidentiality

For getting secret message in this scheme, all messages are encrypted with session key which is encrypted with public key of intended reception so only user have the private key (intended reception) can get the session key of decryption of the message and read the message which satisfy the confidentiality of the message.

5.2.2 Message integrity

Our scheme don't take message integrity in consideration because to satisfy message integrity we need to make calculation of HMAC (Hash-based Message Authentication Code) and embedded it in encrypted payload is a specific construction for calculating a message authentication code (MAC) this calculation makes overhead that we can't deal with it in our case in connectionless multicast system in WSNs.

5.2.3 The Boneh-Franklin IBE algorithm security

In order to add new node in wireless sensor networks with symmetric key technique, some private keys have to be distributed to the new node. Also, some index information has to be changed in case a node is deleted. But in our scheme, based on IBE algorithms, adding or deleting a node does not affect other nodes, because only identities of nodes are used as public keys. The scheme is independent of network size. Moreover, it is easy to reach a time-stamped identity by using "bob@company || 03" as a public key [24].

Chapter 6 – Simulation and Results

In this chapter we present the mechanism used to test our scheme and the implementation of our secure connectionless multicast protocol. The simulator used in this thesis is JiST/SWANS (Java in Simulation Time/Scalable Wireless Ad-hoc Network Simulator) which is a discrete-event simulator [11].

6.1 JiST

We choose JiST simulator, because it is a Java-based simulation platform that executes discrete event simulations efficiently by embedding simulation semantics directly into the Java execution model and transparently performs important optimizations via byte code-level program transformations. Also the system provides standard benefits that the modern Java runtime affords. In addition, JiST is efficient, out-performing existing highly optimized simulation runtimes both in space and time.

JiST transparently introduces simulation time execution semantics to simulation programs written in plain Java and they are executed over an unmodified Java virtual machine. JiST consists of four components:

1- Compiler.

2- Byte code rewriter.

3- Simulation kernel

4- Virtual machine

Figure 6.1 shows the compiler and virtual machine are standard Java language components. Simulation are compiled then dynamically instrumented be rewriter and finally executed.

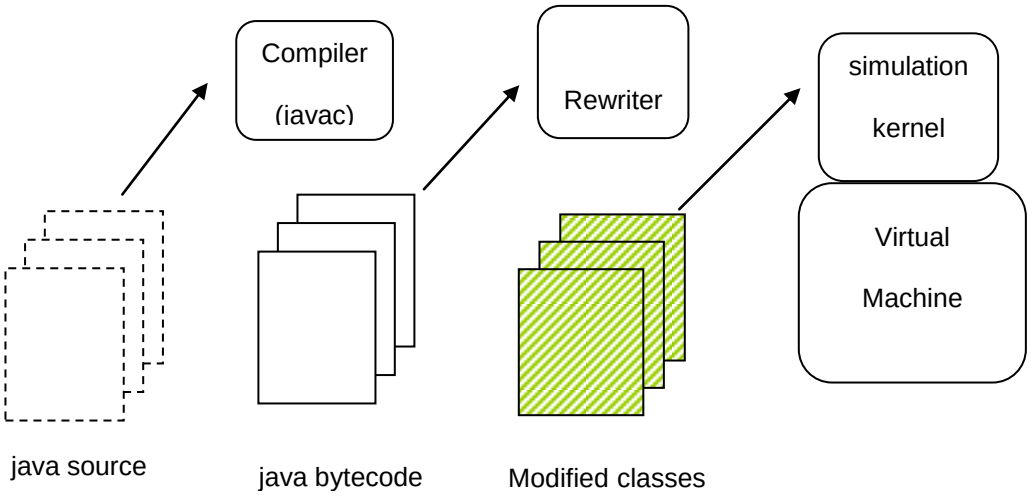

Figure 6.1. JiST components

6.2 SWANS

SWANS SWANS is a Scalable Wireless Ad hoc Network Simulator built atop the JiST platform. The SWANS simulator consists of event-driven components that can be configured and composed to form the desired wireless network simulation as shown in figure 6.2.

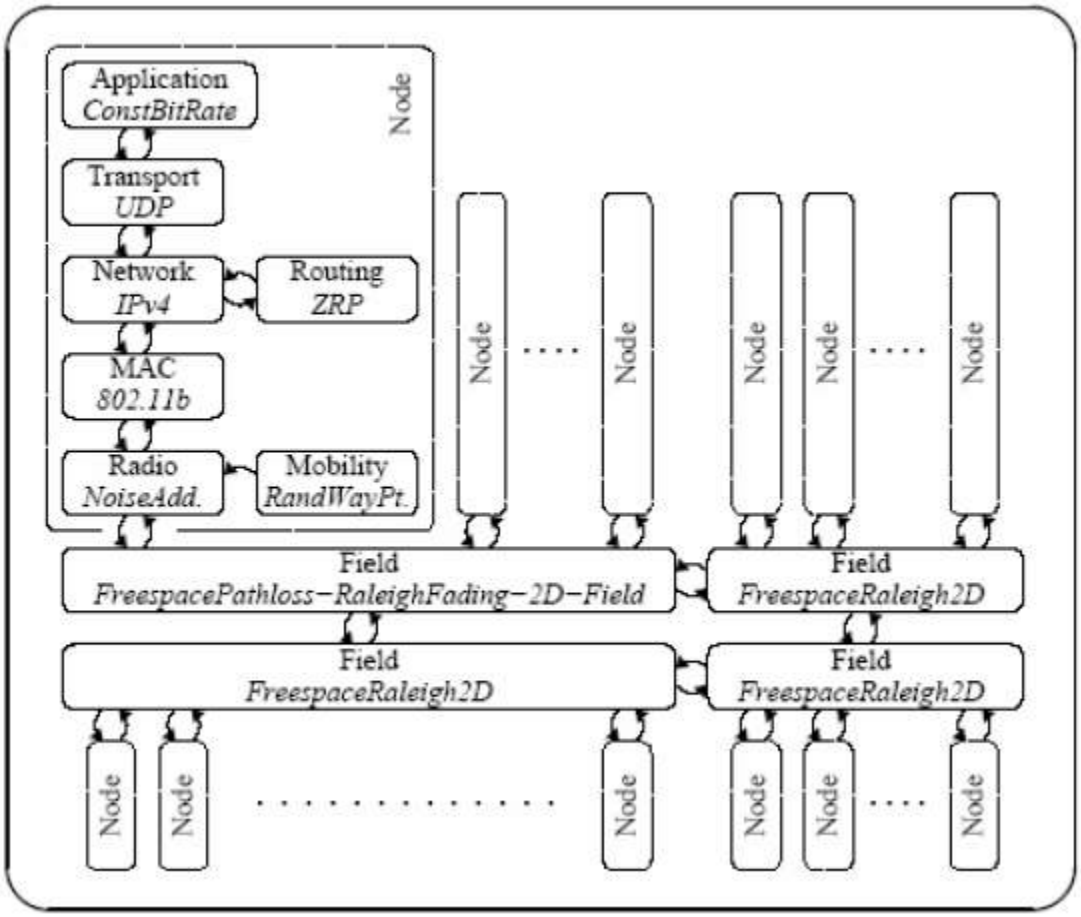

Figure 6.2. SWANS components

we modify swans component to fit our proposed scheme by adding uCast algorithm component in routing components we create full implementation to uCast algorithm and integrate it with SWANS components, figure 6.3 shows the modification to routing component in Swans architecture

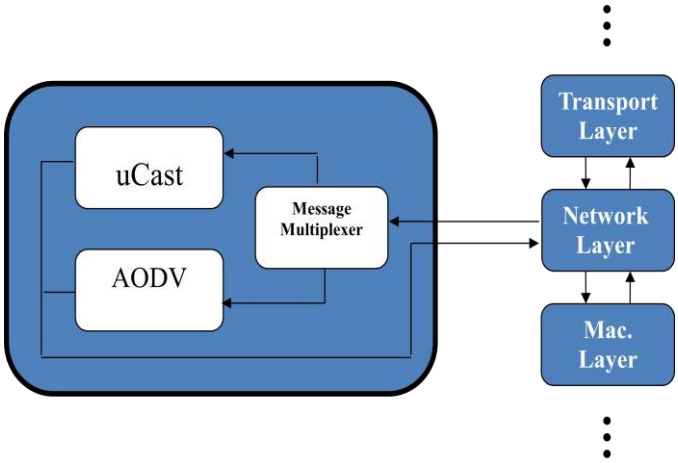

Figure 6.3 modifications to swans network layer component.

6.3 Results

Our evaluation is based on the simulation of 30 sensor nodes in area of 2000x2000 m2. The radio transmission range is assumed to be 635m and the two-ray ground propagation channel is assumed with a data rate of 1 Mbps. A single node multicast to variable numbers of nodes (5, 10, 15, and 20) a message of 128-byte data.

Simulation time is 500 seconds and each simulation scenario is repeated 3 times to obtain steady-state performance metrics. Four cases are assumed: uCast, secure uCast (suCast-AES-128), secure uCast (suCast-RSA-1024) and secure uCast (suCast-IBE-160). The symmetric cipher used is AES cipher with key size of 128 bit, the asymmetric cipher used is RAS with key size 1024 bit and the asymmetric IBE cipher used is Boneh-Franklin with key size 160 bit. The session key is generated using a random key generator.

The first evaluation was to measure the average end-to-end delay. Figure 6.4 shows that the delay of suCast using IBE-160 bit algorithm is the smallest average end-to-end delay in asymmetric security algorithm larger than uCast. The reason is that the packet size increased in suCast-IBE-160 which needs more transmission time and compared with suCast-AES-128 algorithm the symmetric algorithm is not acceptable to use this algorithm as key management techniques as we mention above in this thesis.

Also the time overhead compared between suCast-IBE-160 and suCast-AES-128 just in key session generation and deployed to the group creation members. Then both algorithms will use same end-to-end delay but suCast-IBE-160 will give more security.

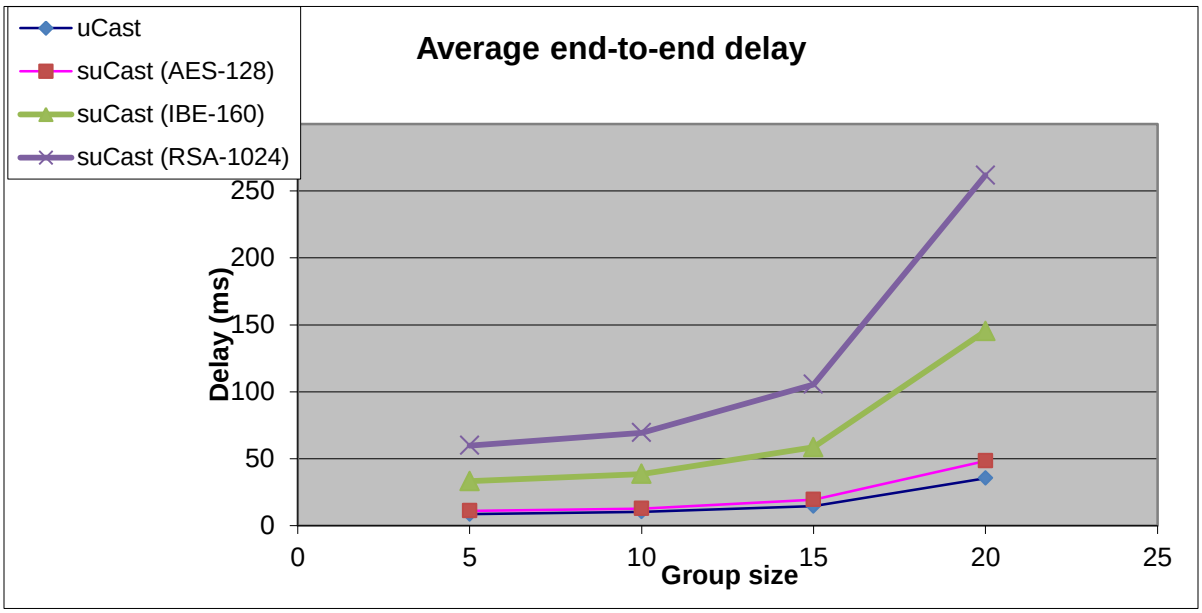

Figure 6.4. Average end-to-end delay

The second evaluation is to measure the packet delivery ratio of the two protocols. Figure 6.5 show that uCast and suCast with all techniques have the same ratio approximately. That result is expected because suCast with all techniques does not change the routing mechanism. The only change in routing is the packet size.

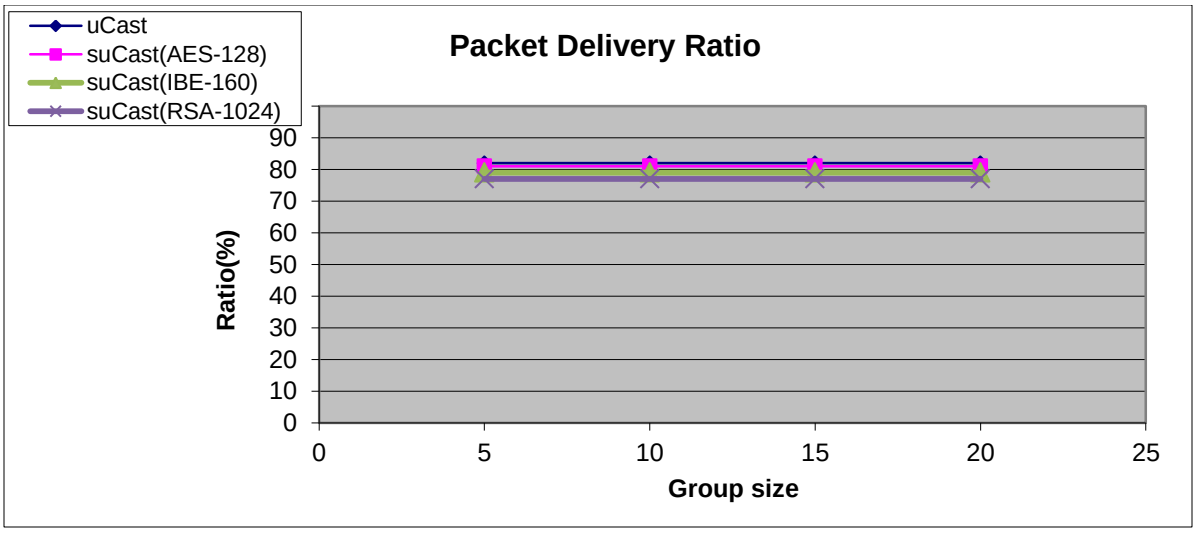

Figure 6.5 Packet delivery ratio

The last evaluation is to measure the average power consumed of the all techniques. Figure 6.6 shows that the power consumed of suCast using IBE-160 bit algorithm is the smallest average power consumed in asymmetric security algorithm larger than uCast. The reason is that the encryption and decryption in suCast-IBE-160 needs more processing time. Compared with suCast-AES-128 algorithm the symmetric algorithm consumes less average power but less secure compared with suCast-IBE-160.

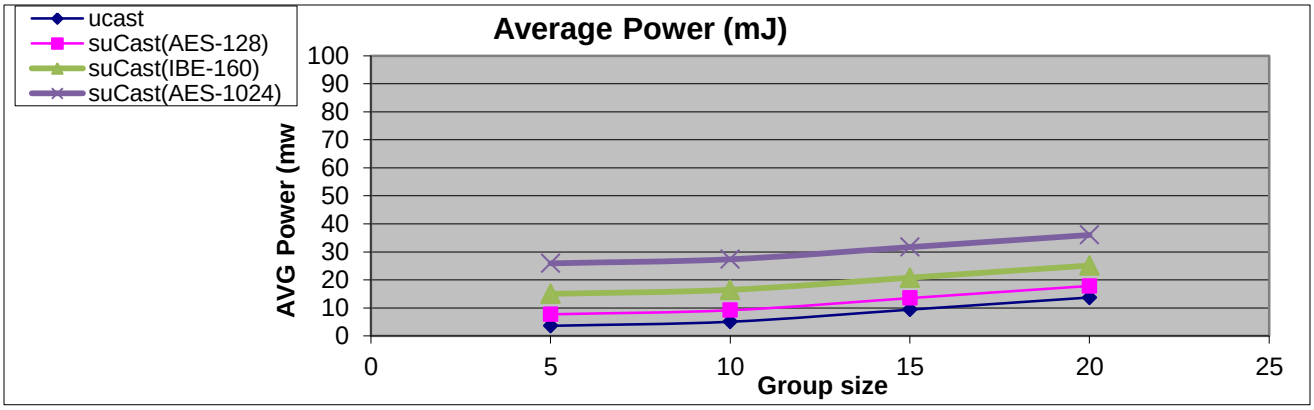

Figure 6.6. Average Power consumed

Chapter 7 – Conclusion and Future Work

Despite of a lot of techniques used to secure connection based multicast protocols and schemes, these techniques fails in providing security in connectionless multicast protocols because they depend on groups creation and management.

This thesis presents a secure and efficient connectionless multicast scheme in WSN using Identity based encryption (IBE) which was proved to be a very effective solution to the problem of providing security to connectionless multicast protocols. The mechanism provides novel techniques of key management for any connectionless multicast group taking in consideration that there is neither group creation nor discovery.

We test our scheme in uCast protocol as example of connectionless multicast protocols. The encryption algorithm used with our novel key management was Identity based encryption (IBE), the comparison shows that IBE was the best compared with PKI and any symmetric algorithm as efficiency and security analysis.

The results show that our adaptive secure connectionless multicast suCast-IBE-160 bit has good behavior taking in consideration both limitation of WSN and connectionless mechanism. Our approach is designed to be more efficient and secure against attacks.

Our techniques can be improved by providing custom encryption algorithm that satisfies both integrity and non-repudiation taking in account the limitations of wireless sensor networks as power, computation capability and storage resources.

References

[1] Qing Cao; Tian He; Abdelzaher," uCast: Unified Connectionless Multicast for Energy Efficient Content Distribution in Sensor Networks"; Issue 2, Feb. 2007 IEEE JNL.

[2] Di Pietro, R.; Mancini, L.V.; Yee Wei Law; Etalle, S.; Havinga, P.; "LKHW: a directed diffusion-based secure multicast scheme for wireless sensor networks"; on 6-9 Oct. 2003 IEEE CNF.

[3] Hung-Min Sun; Chien-Ming Chen; Feng-Ying Chu; "An Efficient and Scalable Key Management Protocol for Secure Group Communications in Wireless Sensor Networks" ; Computers and Communications, 2007. ISCC 2007.

[4] Haowen Chan, Adrian Perrig, and Dawn Song; "KEY DISTRIBUTION TECHNIQUES FOR SENSOR NETWORKS"; Carnegie Mellon University, {haowenchan, perrig, dawnsong}@cmu.edu ; chapter 1, pages 1–27, Nov. 2002.

[5] Wireless Network Security;YANG XIAO, XUEMIN SHEN,and DING-ZHU DU;2007 Springer Science.

[6] P. Trakadas, T. Zahariadis, H.C. Leligou, S. Voliotis, K. Papadopoulos;"Analyzing Energy and Time Overhead of Security Mechanisms in Wireless Sensor Networks"

[7] Sultana, N.; Eui-Nam Huh; "Secure Group Communication in Mobile Wireless Sensor Networks "; Advanced Communication Technology, 2008. ICACT 2008.

[8] Jyh-How Huang; Buckingham, J.; Han, R.; "A Level Key Infrastructure for Secure and Efficient Group Communication in Wireless Sensor Network ";Security and Privacy for Emerging Areas in Communications Networks, 2005 IEEE.

[9] A. S. Wander, N. Gura, H. Eberle, V. Gupta, and S. C. Shantz, "Energy analysis of public-key cryptography for wireless sensor networks," in Third IEEE International Conference on Pervasive Coinputing and Communications (PERCOM'05). IEEE Computer Society Press, 2005, pp. 324-328.

[10] H. Yang, H. Luo, F. Ye, S. Lu, and L. Zhang, "Security in mobile ad hoc networks: challenges and solutions," IEEE Wireless Communications,vol. 11, no. 1, pp. 38-47, Feb. 2004.

[11] Java in Simulation Time / Scalable Wireless Ad hoc Network Simulator. http://jist.ece.cornell.edu/.

[12] Robert Szewczyk, Joseph Polastre, Alan Mainwaring, and David Culler. Lessons from a sensor network expedition. In First European Workshop on Wireless Sensor Networks (EWSN '04), January 2004.

[13] W. Du, J. Deng, Y. Han, and P. Varshney. A pairwise key pre-distribution scheme for wireless sensor networks. In 10th ACM Conference on Computer and communications Security (CCS'03), Washington D.C, USA, October 2003

[14] De-Nian Yang; Wanjiun Liao; "Protocol design for scalable and adaptive multicast for group communications"; Network Protocols, 2008. ICNP 2008. IEEE International Conference.

[15] Ghosh, S.K.; Patro, R.K.; Raina, M.; Thejaswi, C.; Ganapathy, V.; "Secure group communication in wireless sensor networks "; Wireless Pervasive Computing, 2006 IEEE

[16] C. Ozturk, Y. Zhang, and W. Trappe. Source-location privacy in energy-constrained sensor network routing. In 2004 ACMWorkshop on Security of Ad Hoc and Sensor Networks, October 2004.

[17] Yih-Chun Hu, Adrian Perrig, and David B. Johnson. Packet leashes: A defense against wormhole attacks in wireless networks. In Proceedings of IEEE Infocom 2003, April 2003.

[18] D. Liu and P. Ning, "Location-based pairwise key establishments for static sensor networks," in 2003 ACM Workshop on Security in Ad Hoc and Sensor Networks (SASN '03), pp. 72-82, 2003.

[19] A. Shamir, "Identity-based cryptography and signature schemes," Advances in Cryptology, CRYPTO'84, Lecture Notes in Computer Science, vol. 196, pp. 47-53,

[20] U. Feige, A. Fiat, and A. Shamir, "Zero-knowledge proofs of identity," J. Cryptology, vol. 1, pp. 77-94, 1988.

[21] A. Fiat and A. Shamir, "How to prove yourself: practical solutions to identification and signature problems," In Proceedings of CRYPTO'86, pp. 186-194, 1986.

[22] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in Advances in Cryptology, CRYPTO 2001, Lecture Notes in Computer Science, vol. 2139, pp. 213-229,2001.

[23] X. Boyen, “Multipurpose Identity-based signcryption, a Swiss army knife for identity-based cryptography,”; in Proceedings of the 23rd Interna. Conf. On Advances in Cryptology, Lecture Notes in Computer Science, vol. 2729, pp. 383-399, 2003.

[24] L. Chen and C. Kudla, “Identity-based authenticated key agreement protocols from pairings”; Cryptology ePrint Archive, Report 2002/184, http://eprint.iacr.org/2002/184, 2002.

[25] B. Lynn, “Authenticated identity-based encryption,” Cryptology ePrint Archive, Report 2002/072, http://eprint.iacr.org/2002/072, 2002.

[26] B. R. Waters, “Efficient Identity-Based Encryption Without Random Oracles,”; Cryptology ePrint Archive, Report 2004/180, http://eprint.iacr.org/2004/180, 2004.

[27] K. Lauter, “The advantages of elliptic curve cryptography for wireless security,”; IEEE Wireless Communications, vol. 11, no. 1, pp. 62-67, Feb 2004.

[28] Szczechowiak, P.; Collier, M.,"Practical Identity-Based Key Agreement For Secure Communication in Sensor Networks", Computer Communications and Networks, 2009. ICCCN 2009. Proceedings of 18th Internatonal Conference on

[29] S. Zhu, S. Setia, S. Xu, and S. Jajodia, “GKMPAN: An efficient group rekeying scheme for secure multicast in ad-hoc networks,” in Proc. ACM Mobiquitous, Boston, MA, Aug. 2004, pp. 42–51.

[30] Gaddour, O.; Koubaa, A.; Cheikhrouhou, O.; Abid, M.; “Z-Cast: A Multicast Routing Mechanism in ZigBee Cluster-Tree Wireless Sensor Networks “; Distributed Computing Systems Workshops (ICDCSW), 2010 IEEE 30th International Conference.

[31] Wang Xiangli; Li Layuan; Wang Wenbo; “An Energy-Efficiency Multicast Routing Algorithm in Wireless Sensor Networks”; Computing, Communication, Control, and Management, 2008. CCCM '08.

[32] Xiaoping Li; Tao Jin; Feng Ling; Shuaizong Wang; Xiaoxing Lv; “The improve of multicast based on of collaboration coding in WSNs”; Digital Content, Multimedia Technology and its Applications (IDC), 2010 6th International Conference.

[33] Shaoliang Peng; Shanshan Li; Lei Chen; Nong Xiao; Yuxing Peng; “SenCast: Scalable multicast in wireless sensor networks “;Parallel and Distributed Processing, 2008. IPDPS 2008.

[34] Long Cheng; Das, S.K.; Jiannong Cao; Canfeng Chen; Jian Ma; “Distributed Minimum Transmission Multicast Routing Protocol for Wireless Sensor Networks “;Parallel Processing (ICPP), 2010 39th International Conference.

[35] D.Wallner, E. Harder, and R. Agee. “Key management for multicast”; Issues and architectures. RFC 2627, IETF, June 1999.

[36] Rong Fan; Jian Chen; Jian-Qing Fu; Ling-Di Ping; “A Steiner-Based Secure Multicast Routing Protocol for Wireless Sensor Network” ; Digital Object Identifier: 10.1109/ICFN.2010.50.

[37] E. M. Royer and C. E. Perkins, “Multicast operation of the ad-hoc ondemand distance vector routing protocol,” in Proc. of MobiCom, August 1999.

[38] J. G. Jetcheva and D. B. Johnson. Adaptive demand-driven multicast routing in multi-hop wireless ad hoc networks. In ACM MobiHoc, October 2001.

[39] S.-J. Lee, M. Gerla, and C.-C. Chiang. On-demand multicast routing protocol. In IEEE WCNC, September 1999.

[40] J. Xie, R. R. Talpade, A. Mcauley, and M. Liu. AMRoute: ad hoc multicast routing protocol. Mob. Netw. Appl., 7(6):429–439, 2002.

[41] C. Wu and Y. Tay, “AMRIS: A Multicast Protocol for Ad hoc Wireless Networks,” in Proc. of MILCOM, November 1999.

[42] C. Gui and P. Mohapatra, “Efficient overlay multicast for mobile ad hoc networks,” in Proc. of IEEE WCNC, March 2003.

[43] Kolberg, M.; Buford, J.; “An XCAST Multicast Implementation for the OverSim Simulator “; Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE.

[44] Alzyoud, F.Y.; Mohamad, I.J.; Tat-Chee Wan; “The effect of traffic load on using XCAST based routing protocol in wireless ad hoc networks “;Information, Communications and Signal Processing, 2009.

[45] Alzyoud, F.Y.; Tat-Chee Wan; Mohamad, I.J.; “The effect of using XCAST based routing protocol in wireless ad hoc networks”; TENCON 2009 - 2009 IEEE Region 10 Conference.

[46] Siregar, L.; Aji, R.F.; Hasibuan, Z.A.; Budiarto, R.; “Quality of Service for IPTV Using Xcast in Ipv6 Network “; Digital Object Identifier: 10.1109/NETAPPS.2010.25.

[47] S. Basagni, I. Chlamtac, and V. Syrotiuk, “Location aware, dependable multicast for mobile ad hoc networks,”; Computer Networks, vol. 36, pp. 659–670, August 2001.

[48] M. Mauve, H. Fuessler, J. Widmer, and T. Lang, “Position based multicast routing for mobile ad-hoc networks,”; University of Mannheim, Tech. Rep. CS TR-03-004, 2003.

[49] K. Chen and K. Nahrstedt, “Effective Location-Guided Tree Construction Algorithms for Small Group Multicast in MANET,” in Proc. of IEEE INFOCOM, June 2002.

[50] J. Sanchez, P. Ruiz, X. Liu, and I. Stojmenovic, “GMR: Geographic Multicast Routing for Wireless Sensor Networks,” in Proc. of IEEE SECON, 2006.

[51] S. Song, D. Kim, B-Y Choi, ”AGSMR: Adaptive Geo-Source Multicast Routing for Wireless Sensor Networks,” In Proc. of the International Conference on Wireless Algorithms, Systems and Applications (WASA), Aug 2009.