IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 1, Issue 5, Oct-Nov, 2013
ISSN: 2320 - 8791
www.ijreat.org

Secure Multicasting for Wireless Sensor Networks

Dona Maria Mani
M.Tech Student(CSE),
Viswajyothi College of Engineering and Technology, Muvattupuzha, Kerala, India.

Abstract **–** Wireless sensor networks (WSNs) consist of one or more base stations and a number of sensor nodes that get stimulated from external events. Message transmission is a critical security service in Wireless Sensor Networks and is vulnerable to different types of attacks. Symmetric key based schemes such as μTESLA [1] and multilevel μTESLA [2] were proposed to provide such services for WSNs; however, these schemes suffer from serious DoS attacks due to the delay in message authentication. Elliptic Curve Cryptography (ECC) [3][4] is widely deployed in wireless devices, where computing power, memory and battery life are limited, owing to its significant advantages over RSA. Public Key Cryptography (PKC) is widely used for multicast authentication. But intensive use of PKC for authentication is expensive to resource constrained sensor nodes. Hence, a novel PKC based authentication scheme using signature amortization is implemented for Wireless Sensor Networks (WSNs) that overcomes the vulnerabilities in symmetric based schemes and reduces the overhead for authentication significantly. The scheme uses a variant of ECC called as Elliptic Curve Digital Signature Algorithm (ECDSA) for this purpose. In addition the scheme also ensures the confidentiality and integrity of messages by Secure Socket Layer protocol (SSL) and encryption using blowfish algorithm followed by base64 encoding. Finally a comparison between the basic schemes of message authentication like HMAC and RSA, and the newly implemented scheme is conducted based on the amount of CPU time consumption.

Key Terms **- WSN, ECC, PKC, ECDSA, authentication, integrity, confidentiality, RSA, HMAC.**

I. INTRODUCTION

Wireless sensor networks (Fig 1.1) consists of spatially dispersed autonomous sensors that monitor physical or environmental conditions, such as temperature, sound, pressure, etc. that cooperatively pass their data through the network to a main location or a base station. Multicasting is one of the fundamental communication primitives in wireless sensor networks. Efficient and intelligent use of bandwidth is paramount, particularly with the advent of video, mobility, and cloud technologies. It is also critical considering the surge in related one-to-many or many-to-many communication-based applications. Multicasting helps to fulfill the requirement of such bandwidth-intensive applications with its inherent ability to replicate single stream when and where necessary. However messages multicasted may be intercepted by adversaries and can lead to destruction of the wireless sensor networks. Hence in order to defend wireless networks against attackers, ensuring secure multicasting is necessary and it serves as a critical security service in Wireless Sensor Networks.

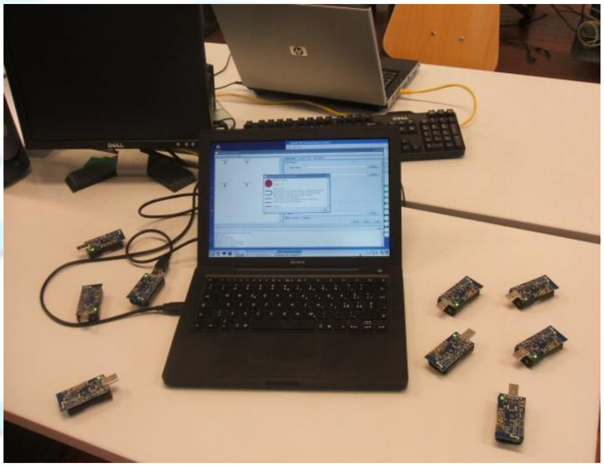


Fig.1.1. Wireless Sensor Network

Sensor nodes are vulnerable to node compromise attacks such as spoofing, altering, or replaying of routing information, selective forwarding, sinkhole attacks, sybil attacks, wormholes, HELLO flood attacks etc [5]. Eavesdropping cause a significant challenge to sensor networks since transmitted messages may be intercepted by an adversary, modified and later rebroadcast to sensor nodes. Moreover a third party can even insert false messages into the network. To counter this insecurity in sensor networks, many symmetric key based authentication schemes such as µTESLA and multilevel µTESLA were introduced.

Providing message authentication for wireless sensor networks using symmetric key cryptography based schemes encounters many challenges. The most important problem is stringent resource constraints on sensor nodes. As a result conventional Public Key Cryptography (PKC) based message authentication schemes seems to be too expensive for wireless sensor networks. For instance, it takes 14 seconds for an exponential operation of 1024-bit RSA on Mica1 motes. Besides resource constraints, sensor nodes are vulnerable to node compromise attacks. Moreover all these schemes suffered from serious DoS attacks due to the delay in message authentication. This renders application of conventional symmetric key cryptography based broadcast authentication schemes to WSNs impractical.

Public Key Cryptography (PKC) is widely used for broadcast authentication to remove the drawbacks of symmetric based schemes[6]. Overhead of PKC is significantly reduced using Elliptic Curve Cryptography. But intensive use of ECDSA for broadcast authentication is highly expensive and hence unaffordable for sensor nodes which are constrained by resources.

The proposed scheme is to implement a secure multicasting scheme in wireless sensor networks that ensures confidentiality, authentication and integrity of messages and reduces the overhead in message transmission significantly. The scheme overcomes the vulnerabilities in symmetric based schemes and reduces the overhead for message authentication significantly. A single signature is used for the authentication of entire multicast messages, which is generated by a variant of elliptic curve cryptography known as Elliptic Curve Digital Signature Algorithm [7]. This reduces the overhead of having separate signature for each message significantly. The proposed multicasting scheme implements a protocol known as Secure Socket Layer for ensuring confidentiality of message transmission. Moreover integrity of multicast messages are ensured using blowfish encryption followed by an efficient encoding technique. The scheme is designed to meet the following properties-

- Low overhead - The computation and communication overhead is to the same degree of the Keyed-Hash Message Authentication Code (HMAC).
- Strong authenticity-Confidence of a receiver in authenticating multicast messages is as strong as each extended block in EB is authenticated by an ECDSA signature.
- Immediate authentication-A receiver can authenticate multicast messages upon receiving them.
- No time synchronization –Time synchronization is not required.
- Resilience to node compromise attacks-It is impossible for an adversary to exploit a compromised receiver to launch a valid multicast authentication.
- High level security- Ensures authenticity, confidentiality and integrity of multicast messages.

Multicasting in wireless sensor networks encounters many significant challenges. Confidentiality, integrity and authentication of multicast messages need to be ensured for a secure multicast scheme in sensor networks. Confidentiality of messages are ensured using a networking protocol called SSL. Integrity of messages are ensured by a strong and efficient way of encryption by blowfish followed by base64 encoding. Authenticity of messages are ensured by Elliptic curve cryptographic variant called ECDSA which ensures low overhead using public key cryptography as well as strong and immediate authentication. In the paper a brief comparison is made between the existing schemes of message authentication and the implemented scheme. From the simulation results it is seen that the implemented scheme offers 3 level security and takes significantly less amount of time for authentication of multicast messages.

ECC is proven to be a methodology that can be widely deployed in sensor networks compared to RSA. ECC employs a relatively short encryption key, a value that must be fed into the encryption algorithm to decode an encrypted message. This short key is faster and requires less computing power than other first-generation encryption public key algorithms. For example, a 160-bit ECC encryption key provides the same security as a 1024 bit RSA encryption key and can be up to 15 times faster, depending on the platform on which it is implemented. The advantages of ECC over RSA are particularly important in wireless devices, where computing power, memory and battery life are limited.

A 3 level secure multicast scheme for message transmission is implemented. The scheme includes the following modules.

- Ensuring integrity of messages by encryption and encoding.
- Single ECDSA to ensure authenticity of entire multicast messages.
- SSL based sensor node authentication to ensure confidentiality of multicast messages.

II. ENSURING INTEGRITY OF MESSAGES BY ENCRYPTION AND ENCODING

Integrity of messages multicasted are ensured by encryption using Blowfish algorithm [8] followed by base64 encoding. On the otherside base64 decoding followed by Blowfish decryption is done.

Blowfish a symmetric key block cipher. It uses 64 bits of data blocks and a variable size key maximum up to 448 bits. It is a version of Feistel Network having 16 times of iteration of a simple encryption function. The main features of Blowfish algorithm is that it includes key dependent S-boxes and has a complex key schedule which makes the algorithm stronger.

The data block of 64 bits are first divided into two halves of 32 bits each as shown in fig 2.1.This algorithm uses two sub key arrays 18-entry P-array and 256-entry S-boxes. The S-boxes maps the 8 bit input into 32 bits output. One entry of P-array is compulsory for each of 16 rounds. The remaining 2 entries of P-array are used after the final round to separately XOR the outputs of each of the halves of the data block.

In the function F, four S-boxes are used and two types of bit operations: XOR and addition of modulo 2^{32} are used. The function divides the input of 32 bits into four S-boxes of 8 bits each. The outputs of first and second S-boxes are first added to modulo 2^{32} and the output of the addition is XOR-ed with the output of third S-box output. The result of XOR operation and the output of fourth S-box is finally added to modulo 2^{32} to get the final output from the function F. The key schedule of Blowfish algorithm starts by initializing the P-array and S-boxes with values derived from the hexadecimal value of pi. The secret key is then byte wise XOR-ed with all the P-entries in order. Because the P-array is 576 bits long (18 P-entries * 32 bits) and the bytes are XOR-ed with all these bits, many implementations may support 576 bit key size. Decryption is exactly the same as encryption technique except the P_1, P_2.... P_{18} are used in reverse order.

Base64 is a group of similar binary-to-text encoding schemes that represent binary data in ASCII string format by translating it into a radix-64 representation. This is done to ensure that the data remain the same without modification during transport. The term Base64 originates from a specific MIME content transfer encoding. These are commonly used when there is a need to encode binary data that needs to be stored and transferred over media that are designed to deal with textual data. It has a number of applications including email via MIME, and storing complex data in XML.

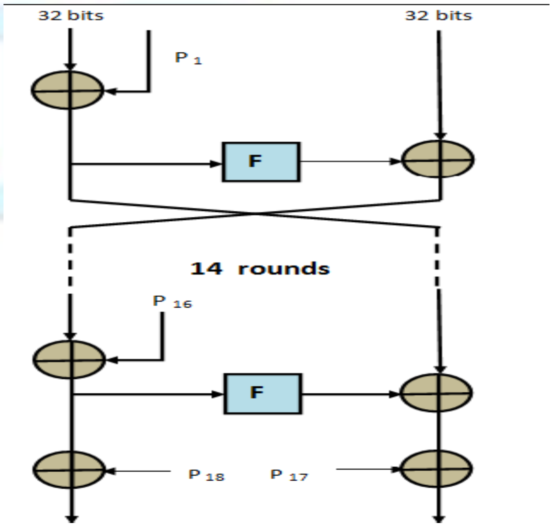


Fig.2.1.Blowfish algorithm

III. SINGLE ECDSA TO ENSURE AUTHENTICITY OF ENTIRE MULTICAST MESSAGES

3.1 SYSTEM DESCRIPTION

Assume a single base station and a number of sensor nodes that are vulnerable to adversary attacks. A single sender multicasts a message to many receivers. Sender may be a base station or a sensor node. The following assumptions are made

bs- base station with key pair (PR_{bs},PU_{bs}), PU_{bs} is stored by all sensor nodes

s- Sender with keypair (PR_s,PU_s)

R_s-Receivers [ri] i=1..c assuming c receivers

Multicast messages from s to receivers in R_s is denoted by $M=[m_i]$ i=1…n assuming n messages.

These messages are organized as extended blocks which is the unit for authentication.

Extended blocks represented as $[EB_i]$ i=0....k assuming a total of k+1 extended blocks.

EB0 – contains authenticator which may be a signature or a MAC used to authenticate an extended block

EB_1…..EB_k contains b multicast messages in M and a specified authenticator.

Total number of multicast messages=n=b*k where k is an integer

EB_k contains (n mod b) messages and k= ┌ n/b ┐

Encryption of m=E(K,m),where K is the key

Decryption of m=D(K,m)

Authenticated Relation

AR on EB consists of ordered pairs <EB_i, EB_j> ,authenticator in EB_i is used to authenticate EB_j.

Collision resistant hash H

It is computationally infeasible to find a pair of inputs(x,y) such that x ≠ y and H(x)=H(y).

Certificate

Contains public key and identity of public keys owner, both signed by a trusted third party called certification authority.

3.2 MODULES USED

The authentication scheme uses the concept of signature amortization that exploits only one ECDSA signature to authenticate all multicast messages. The only one signature is used to authenticate the authenticator in EB_0.The authenticator in EB0 is used to authenticate EB_1 that contains b multicast messages in M and one authenticator. The authenticator in EB_1, in its turn, is used to authenticate EB_2 that contains b multicast messages in M and one authenticator. The process continues until EB_k. As a result, all multicast messages can be authenticated with only one signature while the overhead of the signature is amortized over them.

The scheme employs the following modules.

1. Signature amortization
2. Generation of ECDSA signature
3. Public key distribution protocol

3.2.1 Signature Amortization

The signature amortization part is presented by three steps: generating extended blocks step multicasting extended blocks step and verifying extended blocks step.

3.2.1.1 Generation Of Extended Blocks

The following steps help in the generation of extended blocks

1. Partition n multicast messages in M into k blocks $B_1…B_k$

$$B= \begin{pmatrix} B1 \\ \vdots \\ Bk \end{pmatrix} = \begin{pmatrix} m1 & \cdots & mb \\ \vdots & & \vdots \\ m(k-1)b+1 & \cdots & mkb \end{pmatrix}$$

2. Initialize d_{k+1} with a random string of characters

3. Initialize i=k

4. Perform the following steps

4.1 Concatenate messages in B_i to generate $CON(B_i)$

4.2 Pad $CON(B_i)$ with digest d_{i+1} to generate Pad $(CON(B_i)\ )$

4.3 Compute the digest of result in step 4.2 using a collision resistant hash function H

4.4 Let EB_i=[B_i d_{i+1}]

4.5 Decrement i

5. Repeat step 4 till i greater than or equal to 1

6.Sign the digest with senders private key PR_s to generate $EB0=d1 \| E(PR_s,d1)$

7.Let EB= $[EB]_{i=0..k}$

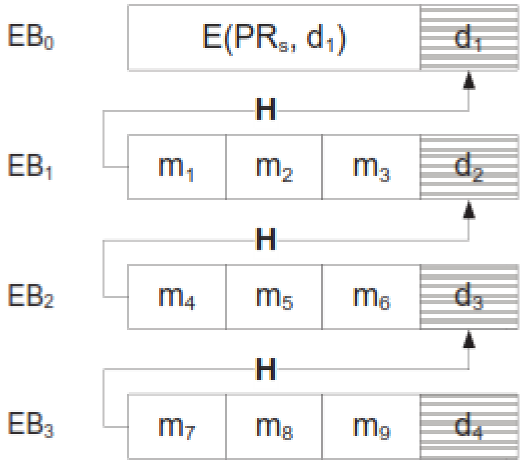


Fig 3.1 An example for generating extended blocks for 9 multicast messages

3.2.1.2 Multicasting Extended Blocks

All ordered pairs <EB$_{i-1}$, EBi> 1≤i≤k, belong to AR on EB. EBi's authentication must depend on EB$_{i-1}$.This is fulfilled by sequential and reliable multicasts.

In sequential multicast extended blocks are multicast according to AR on EB. Sender s multicasts EB$_{i-1}$ before EBi. Messages in each extended block are multicast according to their indexes, i.e., sending sequence for EBi is $m_{(i-1)b+1}$,$m_{(i-1)b+2}$ …..m $_{ib}$. Digest d_{i+1} in EBi should be sent together with a multicast message in EBi since the size of a digest is relatively small. On receiving a multicast message mj, a receiver in Rs checks whether m_j belongs to current extended block, say, EBi, whose digest, di, has been received and authenticated with EB$_{i-1}$.If m_j belongs to EBi, the receiver tries authenticating EB and broadcasts m_j to its neighbors after a short back off. This means all receivers exchange messages of EBi with each other. During the exchange, a receiver may receive multiple copies of m_j from different neighbors. Thus, the receiver would obtain m with high probability though each transmission is not reliable. In short, the unreliable transmission is counteracted by making full use of multicast nature. If mj belongs the block EB$_{i+1}$ which follows EBi and EBi has not been authenticated yet, that indicates all messages of EBi has been multicast. The receiver would not get missing messages later. Hence, the receiver buffers mj and multicasts an acknowledgement to ask for the missing messages belonging to EBi. After authentication of EBi, mj will be multicast.

The reliable multicast is performed by acknowledgements and replies. To reduce communication overhead, one acknowledgement is used to specify all missing messages of one extended block but the size of an acknowledgement be several bits larger than that to one missing message. The acknowledgement contains two fields. The first field specifies the identity of an extended block. The second field is a bit-vector indicating all missing messages in one extended block. The bit-vector is a mapping to all messages in one extended block. Thus, the size of the bit-vector equals the number of messages in the extended block. The receiver's neighbors are responsible for remulticasting the lost messages specified in an acknowledgement.

Since the neighbors multicast mj belonging to EB$_{i+1}$ to the receiver, they possess all messages belonging to EBi. Hence, it is unnecessary to ask sender s that may be multi-hops far away for mj. To avoid collision, each neighbor selects a random time to delay its reply. Sequential multicast and reliable multicast hence guarantee the successful authentication of extended blocks with low overhead.

3.2.1.3 Verifying Extended Blocks

According to multicasting extended blocks step, EB0 reaches receivers in R first. di in EB0 is authenticated by the signature, that is, if D(PUs,E(PRs,d1)) = d_1, d_1 is authentic. Extended blocks in EB* are authenticated in an efficient way, just using a collision resistant hash. Digest di, 1< = i <= k, in EB$_{i-1}$ that reaches receivers in R in advance is used to authenticate EBi, that is, if H(m$_{(i-1)b+1}$||…||m$_{ib}$||di+1) = di,EBi is authentic.

3.2.2 Generation Of ECDSA Signature

The ECDSA signature is on the basis of ECC, which offers equivalent security with substantially smaller key size compared to RSA(e.g., a 160-bit key of ECC offers the same level of security as a 1024-bit key of RSA).Thus, ECC has the advantages in computation, bandwidth and memory savings. Because of the advantages, d1in EB0 is signed with an ECDSA signature.

Sender s and receivers in Rs establish elliptic curve domain parameters T =(p, a, b, G, q, h) in advance. Storing T in sensor nodes before deployment is an option. p is a prime that specifies the finite field Fp. a and b are coefficients of the elliptic curve $y^2= x^3+ax+b \pmod p$ where $4a^3+27b^2 \neq 0 \pmod p$. G refers to the base point on the elliptic curve.q is a prime indicating the order of G. h is the cofactor h =#Ep(a, b)/q where #E (a, b) stands for the number of points on the elliptic curve.

To sign digest d1, sender s creates the key pair (PRs,PUs) that satisfies PUs= PRsG where PRs is an integer in Fp and PUs is a point on the elliptic curve. Then, it selects an ephemeral key pair (u, U) that satisfies U = uG where u is an integer in Fp and U is a point on the elliptic curve. It computes $r = x_u$(mod q) where x_u is the x coordinate of point U, and d_e= H(d1) where H is a collision resistant hash. It sets e = d_e if $\ulcorner \log_2 q \urcorner \geq L_{de}$ where L_{de} refers to the length of d_e. Otherwise, let e equal the leftmost $\ulcorner \log_2 q \urcorner$ bits of d_e. At last, it computes $w = u^{-1(}e+rPRs)(modq)$. r and w are the ECDSA signature. The complete expression of EB0 is d1||r||w.

3.2.3 Public Key Distribution Protocol

To bootstrap the multicast authentication, receivers in Rs should get the public key PUs from sender s. PUs's distribution should be in an authenticated manner because adversaries may inject forged PUs. PUs's distribution should introduce low overhead to resource constrained sensor nodes .With these in mind, it is proposed to distribute PUs through a certificate.

The proposed public key distribution protocol is implemented by three steps as follows, where base station bs acts as CA because it cannot be compromised by adversaries. Sensor nodes are preloaded with base station bs's public key PUs.

1) Sender s sends a request REs to base station bs and asks for generation of a certificate. The request REs contains sender's identity IDs and public key PUs, i.e.,REs=IDs‖PUs

2) Base station bs replies with a certificate Cs,which contains the certificate's identity IDc and request REs signed by base station bs with its private key PRbs.The whole expression of the certificate is Cs=IDc‖IDs‖PUs‖E(PRbs,IDc‖IDs‖PUs)

3) Sender s multicasts Cs to all receivers in Rs.

Public key PUs included in Cs is distributed through multicast (step 3). Certificate Cs remains valid over a long period unless it is explicitly revoked by base station bs. It means that private key PRs corresponding to PUs in Cs used to sign many chains of extended blocks. Moreover, if the intended receivers are changed without revocation of Cs, just remulticasting Cs is required. Receivers do not need to contact CA throughout the protocol.

The public key distribution protocol may undergo attacks that adversaries mislead traffic or jam the communication to the CA. If there are multiple CAs, i.e., more than one base station, the effect of the attacks will be relieved to an extent.

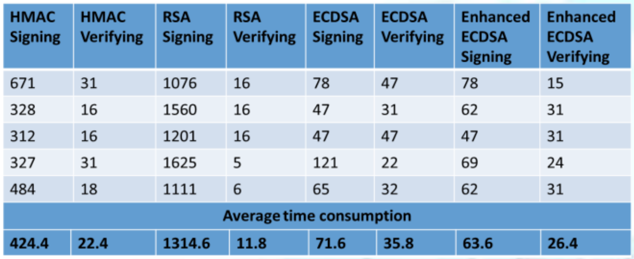
IV.SSL BASED SENSOR NODE AUTHENTICATION TO ENSURE CONFIDENTIALITY OF MULTICAST MESSAGES

Confidentiality of messages ensure that only genuine receivers in Rs gets PUs from sender. This is achieved using SSL[9].Secure Socket Layer (SSL) is a cryptographic protocol that provide communication security over the Internet. Most electronic commerce (e-commerce) applications in use today employ the Secure Sockets Layer (SSL) or Transport Layer Security (TLS) protocol to authenticate the server and to cryptographically protect the communication channel between the client and the server. SSL/TLS provides support for user authentication based on public key certificates. There are many options here, including personal identification numbers (PINs), passwords, passphrases, as well as "strong" authentication mechanisms, such as one-time password (OTP) and challenge-response (C/R) systems. A KeyStore stores the certificate. Sensor nodes are loaded using separate virtual machines and those nodes having access to KeyStore are only genuine.

V.RESULTS

A comparison is made between RSA, HMAC, ECDSA and enhanced ECDSA .It is found that enhanced ECDSA provides a 3 layer security over the normal ECDSA scheme of authentication and also significantly reducing the CPU time. Blowfish encryption takes significantly less amount of time compared to when the experiment was conducted on AES.

Table 5.1 shows a comparison chart based on the signing and verification times for RSA, HMAC, ECDSA and Enhanced ECDSA. Fig 5.1 shows the graph based on CPU consumption time for multicast authentication.



HMAC Signing	HMAC Verifying	RSA Signing	RSA Verifying	ECDSA Signing	ECDSA Verifying	Enhanced ECDSA Signing	Enhanced ECDSA Verifying
671	31	1076	16	78	47	78	15
328	16	1560	16	47	31	62	31
312	16	1201	16	47	47	47	31
327	31	1625	5	121	22	69	24
484	18	1111	6	65	32	62	31
Average time consumption							
424.4	**22.4**	**1314.6**	**11.8**	**71.6**	**35.8**	**63.6**	**26.4**

Table 5.1 Signing and Verification times of various schemes

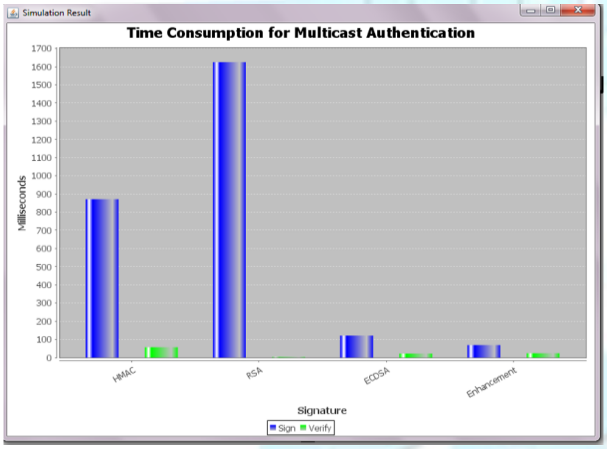


Fig 5.1 Time Consumption for Multicast Authentication

VI.CONCLUSION

A novel method was implemented for message multicasting that counters the defects of symmetric key based schemes. The scheme offers a 3 level secure way of multicasting since confidentiality; integrity and authentication of multicast messages are ensured. The scheme does not require time synchronization, has an efficient public key distribution protocol and achieves immediate authentication that a receiver authenticates a message immediately upon receiving it.

The scheme implements the concept of signature amortization using a single ECDSA signature for the authentication of all multicast messages. The overhead of signature is amortized over all multicast messages. The scheme retains the same security as the conventional PKC based schemes. Using Elliptic Curve Cryptography in the scheme helps to reduce the overhead of using public key cryptography in wireless sensor networks. Signature amortization makes the scheme affordable to the current generation of sensor nodes which were otherwise not affordable owing to the intensive use of Elliptic Curve Digital Signature Algorithm. Moreover the scheme has low computation time making it highly affordable.

VII. REFERENCES

[1] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler, “SPINS: security protocols for sensor networks”, in Wireless Networks, vol. 8, no. 5,pp.521–534, 2002.

[2] D. Liu and P. Ning, "Multilevel μTESLA: broadcast authentication for distributed sensor networks”, in ACM Trans. Embeded Computing Syst.,vol. 3, no. 4, pp. 800–836, 2004.

[3] I. Blake, G. Seroussi, and N, “Smart Elliptic Curves in Cryptography”,Cambridge, 1999.

[4] Lejla Batina, Nele Mentens, Kazuo Sakiyama, Bart Preneel, and Ingrid Verbauwhede , “Low-Cost Elliptic Curve Cryptography for Wireless Sensor Networks”, Katholieke Universiteit Leuven, ESAT/COSIC, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium.

[5] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures”,in Ad Hoc Networks, vol. 1, no. 2-3, pp.293–315, 2003.

[6] F Amin, A H Jahangir, and H Rasifard,“Analysis of Public-Key Cryptography for Wireless Sensor Networks Security”,*WorldAcademy of Science, Engineering and Technology*,31(July):530–535, 2008.

[7] Yongsheng Liu, Student Member, IEEE, JieLi, Senior Member, IEEE, and Mohsen Guizani, Fellow, "PKC Based Broadcast Authentication using Signature Amortization for WSNs", in IEEE Transactions,2012.

[8] Gulshan Kumar, Mritunjay Rai and Gang-soo Lee, “Implementation of Cipher Block Chaining in Wireless Sensor Networks for Security Enhancement”,

in International Journal of Security and Its Applications Vol. 6, No. 1, January, 2012.

[9] Rolf Oppliger, Ralf Hauser, David Basin, "SSL/TLS Session-Aware User Authentication:A Lightweight Alternative to Client-Side Certificates", eSECURITY Technologies, Beethovenstrasse 10, CH-3073 Gümligen.