

A New Secure Authentication Scheme Based Threshold ECDSA For Wireless Sensor Network

Hao Wang and Zhongfu wu
Computer Science Department
Chongqing University
Chongqing, CN

Xin Tan
Communication Department
CQUPT
Chongqing, CN

Abstract - *Security is an important issue for Wireless Sensor Network (WSN), especially for security-sensitive application. In this paper, we address secure authentication problems in WSN. First, we propose distributed authentication model for WSN, and secondly, advance the threshold ECDSA authentication scheme in this model, finally construct a new way to choose distributed CA servers of threshold authentication scheme. Our simulation results and analysis show that our scheme can reduce effectively bandwidth and computation resources while providing a robust, highly secure and highly available authentication service.*

Keywords: Authentication, Threshold ECDSA, WSN.

1 Introduction

Wireless sensor network (WSN) is used in dangerous and hazard remote environment, or in military tactical operations. Wireless sensor network relies on wireless communication to keep the network connected, and is without any infrastructure and centralized control. Also its resource is limited, less memory and low battery energy, and the topology of the WSN is dynamically changing and the nodes of the WSN are often mobile. Due to above characteristics, a major challenge in the design of WSN is to protect their vulnerability from security attacks. The security issues of WSN is list as follow: First, use of wireless link renders network to link attacks ranging from passive eavesdropping to active impersonation, message replay, and message distortion. Second, nodes roaming in a hostile environment (e.g., a battlefield), with relatively poor physical protection, have probability of being compromised. Third, trust relationships among nodes change because nodes frequently join and leave the network. Finally, the WSN may consist of hundreds or even thousands of nodes, and Security mechanisms should be scalable to handle such a large network. Therefore, the security issues must be thoroughly addressed to provide any successful applications.

The rest of paper is organized as follows. In Section 2, we describe the basic authentication schemes; In Section 3, we present our distributed authentication model for WSN; In Section 4, we advance threshold elliptic curve digital

signature algorithm (ECDSA) in this model; In Section 5, we propose a new method to choose distributed Certification Authorities (CA) servers; In Section 6, we present the simulation results; Finally, we conclude this paper in Section 7.

2 Relate Works

Popular network authentication architectures include Kerberos and the X.509 standard. Two entities authenticate each other via trusted certification authority (CA). While this model works well in wired networks, it fails in large WSN environment for several reasons: WSN provide no infrastructure support, and the cost of maintain such centralized servers may be prohibitively high. Second, the CA servers are exposed to single points of compromise and failures. Some variations of above model, such as hierarchical, can not address issues like service availability and robustness.

PGP follows a web-of-trust authentication model. In PGP, certificates are issued by the users themselves base on their acquaintances, and PGP uses digital signatures as its form of introduction. When any user signs for another user's key, he or she becomes an introducer of that key. However, this approach does not scale beyond a relatively small community of trusted individuals. It would be difficult for each node to maintain a long list of trusted friends.

Zhou and Hass propose a partially distributed certificate authority that makes use of a (t, n) threshold scheme to distribute the services of the certificate authority to a set of specialized server nodes, but they did not propose how to choose distributed certificate authority, and its signature mechanism is too complicated for WSN. Similar to the partially-distributed CA, the fully-distributed certificate authority proposed by Luo and Lu extends the idea of the partially-distributed approach by distributing the certificate service to every mode.

There are some other solutions to this problem. In this paper, we propose distributed authentication model for WSN, and advance threshold elliptic curve digital signature algorithm (ECDSA) in this model, which is faster, less

bandwidth and lower computation resources than other digital signature algorithm, and also design a new method to choose distributed CA serves in our authentication scheme.

3 Authentication Model for WSN

In the section, we build the authentication model for WSN. The model is based on followed assumption: no single node is be trusted in WSN, only a group of nodes can be trusted; the model consists of t CA server node which perform (n, t) threshold ECDSA operation, and normal node. The basic theory is that only t shares can perform certification verification jointly, whereas, it is infeasible for at most t shares to do so, even by collusion. The authentication model for WSN is shown in Fig.1. In the Fig.1, the system has n nodes with a public key and a secret key (K/k) . System public key K is open to all nodes,

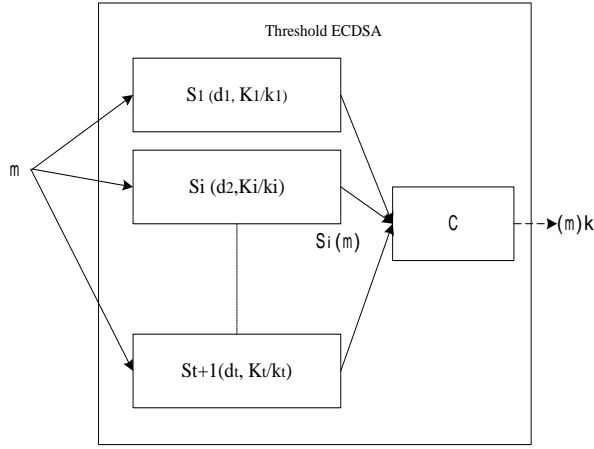


Fig.1 Authentication model for WSN

and system secret key k is divided into d_1, d_2, \dots, d_t by S_1, S_2, \dots, S_n node; Besides, each node has his own public/secret key $K_1/k_1, K_2/k_2, \dots, K_n/k_n$. For the service of authentication, each CA server generates a partial signature $S_i(m)$ and submits the partial signature to a combiner with t correct partial signatures, the combiner is able to compute the signature so that verify signature.

4 Threshold ECDSA Scheme

In the section, we design (t, n) Threshold ECDSA, and then, present three phases of Threshold ECDSA, that is initialization phase, Threshold signature phase, and signature recovery phase, and Threshold ECDSA system consists of distributed CAs, sender A of signature, and receiver B.. Threshold ECDSA consumes less computing resource with higher security, especially for constraint device, such as WSN.

(1) System initialization

System initialization steps is listed as follows:

Step 1: Select a secure elliptic curve in $F_q, E(F_q)$, and a base point G with order n (n is a large number);

Step 2: Assume CAs set is $S = \{p_1, p_2, \dots, p_n\}$, where $p_i (i=1, 2, \dots, n)$ is the identification of CA nodes, ID_i , and $ID_i \neq ID_j (i, j = 1, 2, \dots, n)$;

Step 3: random select secret key of S is $d_s \in \{1, 2, \dots, n-1\}$; and the public key is

$$K_s = d_s \bullet G \in E(F_q)$$

and then random generate $t-1$ polynomial:

$$f(x) = d_s + a_1x + a_2x^2 + \dots + a_{t-1}x^{t-1} \text{ mod } n$$

And then, compute secret key and public key of p_i

$$d_i = f(ID_i)$$

$$K_i = d_i \bullet G$$

Step 4: CA distribute d_i through secure channel to p_i , and then public the parameters: $E(F_q), G, n, K_s$, and ID ;

(2) Threshold signature phase

Assume there are t signers in the S set, p_1, p_2, \dots, p_t , who sign the Message M and send to P_B , the steps of signature is as follows:

Step 1: for the signers $p_i (i= 1, 2, \dots, t)$, random select $k_i \in \{1, 2, \dots, n-1\}$, and then compute

$$V_{i1} = k_i \bullet G$$

$$V_{i2} = k_i \bullet K_B$$

Finally, send them to signature collector;

Step 2: verify $V_{i1} \neq V_{i2}$, then compute

$$T_1 = \sum_{i=1}^t V_{i1}$$

$$T_2 = \sum_{i=1}^t V_{i2}$$

$$v = F_x(T_1 + T_2) \text{ mod } n$$

$$R = M \bullet v$$

Finally, broadcast R;

Step 3: the signers π_i ($i=1, 2, \dots, t$) compute

$$s_i = k_i + R \bullet d_i \bullet ID_i \bmod n ;$$

Finally, the signers send s_i to signature composer.

Step 4: after receive signature, the signature composer verify signature message s_i :

$$s_i \bullet G - K_i \bullet R \bullet ID_i = V_{i1},$$

if this equation is correct, then signature composer compute

$$S = \sum_{i=1}^t s_i ; \text{ and send signature message to } P_B.$$

(3) Signature message recovery

When P_B receives (R, S), firstly, compute

$$T_1 = S \bullet G - R \bullet K_B ;$$

$$T_2 = T_1 \bullet d_B ;$$

And then compute

$$v = F_x(T_1 + T_2) \bmod n ;$$

Finally we get recovery message

$$M = R \bullet v^{-1} ;$$

5 Selection of CA Servers

Authentication in a network usually requires participation of trusted entities, Since WSN has no centralized server for trust and key management. We firstly define a fully distributed trust management algorithm to choose distributed CA with higher trust value. We also can choose CA servers which trust value are more than some thresh value. The trust model is shown in Fig.2. In the Figure, the line with arrow from K_u to K_1 represent K_u trusts K_1 , the number above this line, T_{u1} means the value of trust., $T_{u1} \in [0,1]$, if $T_{u1}=1$, which means K_u trust K_1 completely, and if $T_{u1}=0$, which means K_u distrust K_1 .

We also define two formula to computes the trust relationships. The first formula computes the trust relationship:

$$V_{K_u, K_i, K_v} = V_{K_u, K_i} \odot V_{K_i, K_v} = 1 - (1 - V_{K_i, K_v})^{V_{K_u, K_i}} \quad (1) \quad \text{This}$$

formula can be used to compute value from K_u to K_v , and K_i is the introducer. Another formula combines values of different relationships:

$$V_{K_v} = 1 - \prod_{i=1}^n (1 - V_{K_u, K_i, K_v}) \quad (2)$$

This formula is used for drawing a consistent conclusion when there are several derived trust relationship of the same trust class between two entities.

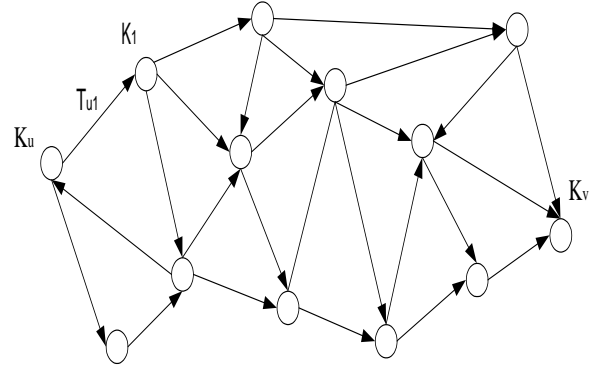


Fig 2 Trust Model for WSN

After we compute the trust value of nodes, then we choose highest value as distributed CA as as to provide secure CA server.

6 Simulation Results

We implemented our design in the network simulator NS-2, and we evaluate the performance of our system in the successful rate, failure rate, and unreachable rate on the requests of public key certification. We simulate a network that contains 200 nodes divided into 10 groups, and t is the percentage of trustable nodes, and m is percentage of malicious nodes. ECDSA conforms to X9.62-1998 standard.

Elliptic curve Domain parameter Setup is as follows:

The field F_p is generated by the prime:

$P=627710173538668076383578942320766641608390870$
 0390324961279

The curve is E:

$$y^2 = x^3 + ax^2 + b \text{ over } F_p, \text{ where}$$

SEED = 3045AE6F C8422F64 ED579528 D38120EA
E12196D5

$r = 3099D2BB$ BFCB2538 542DCD5F B078B6EF
5F3D6FE2 C745DE65

$a =$ FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
FFFFFFFF FFFFFFFF

b = 64210519 E59C80E7 0FA7E9AB 72243049
 FEB8DEEC C146B9B1
 Base point G (with point compression):
 03 188DA80EB03030F6 7CBF20EB
 43A18800F4FF0AFD 82FF1012
 G has prime order:
 n=627710173538668076383578942317605901376719477
 3182842284081
 h = 1
 Key Generation:
 d=651056770906015076056810763456358567190100156
 695615665659

$$K = d \bullet G = (x_K, y_K)$$

with point compression):
 02 62B12D60 690CDCF3 30BABAB6 E69763B4
 71F994DD 702D165A

6.1 Evaluation on rating to trustable nodes

We compare the relationship ratings to percentage of trustable nodes. Fig.2 shows that the successful rate rises with the increase in the percentage of trustable nodes. The increase of successful range is mainly due to more trustable nodes can be selected to be distributed CA servers. The Fig.3 also shows that the successful rate remains steady after the percentage of trustable nodes at reach , this implies that nodes are able to find enough distributed CA servers to request public key authentication after t reaches a certain value.

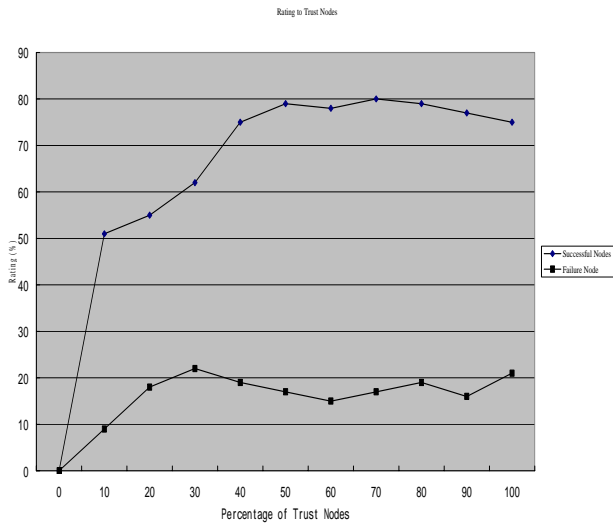


Fig.3 Rating to % of Trustable nodes with m=80%

6.2 Comparison to PGP method

We compare different ratings of authentication service we proposed with the Pretty Good Privacy (PGP) method. In Fig.4, we find that our authentication service has a much higher successful than of the PGP approach in facing malicious nodes.

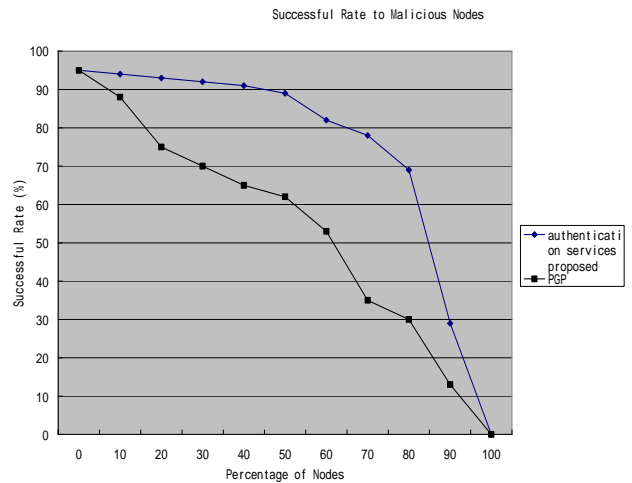


Fig.4 Comparison of Successful Rate to Percentage of Malicious Nodes

7 Conclusion

The security of this scheme is based on the difficulty of ECDLP; Any attackers less than $t-1$ could not reconstruct k from the compromised CA nodes; t nodes sign the message with their secret key d_i to avoid signature node get the secret key of t nodes; In our trust model , if we find some nodes is compromised or is malicious, we can reduce their trust value to 0; In this scheme, we choose CA nodes with highest trust value through our trust model to reduce risks of distributed CAs. In this scheme, we bind trust value, ID with secret key/public key together to prevent disguised nodes attacks.

WSN can be used in dangerous or military fields and its security is an important issue for WSN, especially for security-sensitive application. In this paper, we address secure authentication problems in WSN. First, we propose trust model for WSN, and describe how to choose distributed CAs with highest trust value in WSN, then based on threshold ECDSA, we advance a new authentication scheme, our theoretic security analysis and simulation show that our scheme can prevent dishonest nodes, malicious nodes or compromised nodes from attack to WSN, meanwhile, our scheme can reduce effectively requirements for bandwidth and computation resources while providing robust authentication service.

8 References

- [1] S. Olariu and Qinwen Xu, "Information Assurance In Wireless Sensor Networks", *IPDPS'05*, pp 236a, April 2005.
- [2] C. Pearce, V. Y. Ma, and P. Bertok, "A Secure Communication Protocol for Ad-Hoc Wireless Sensor Networks", *Intelligent Sensor, Sensor Network and Information, Processing Conference*, pp 79-84, Dec 2004;

- [3] L.Zhou and Z.j. Hass, "Securing Ad Hoc Networks", *IEEE Network Magazine*, vol. 13, issue 6, 1999
- [4] J.Kohl and B. Neuman, "The Kerberos network authentication service (version 5)," *RFC-1510*
- [5] A. Aresenault and S. Turner, "Internet X.509 public key infrastructure", *draft-ietf-pkis-roadmap-06.tex*, 2000
- [6] S.Garfinkel, "PGP: Pretty Good Privacy," *O'Reilly & Associates Inc.*, USA, 1995
- [7] A. Abdul-Rahman, "The PGP trust model", *EDI-Forum: the Journal of Electronics Commerce*, April 1997
- [8] E.C.H. Ngai and M.R. Lyu, "An authentication Service Against Dishonest User in Mobile Ad Hoc Networks", *IEEE Aerospace Conference Proceedings*, pp 1275-1285, 2004
- [9] Y.Desmedt, Y. Frankel, "Threshold cryptography ", *In Proc ofCrypto'89, Lecture Notes in Computer Science, LNCS 435, Springer Verlag, Brighton English*,11990. 307-315
- [10] Y. Dai, C. Yang, "(t, n) threshold signature scheme based on ECC", Vol 34, *Computer Application Research*, 2004.
- [11] C. Park, K. Kurosawa, "New ElGamal Type Threshold Digital Signature Scheme", *IEICE Trans*, 1996, E79-A(1):86-93
- [12] NS-2 (The Network Simulator). <http://www.isi.edu/nsnam/ns/>
- [13] "Public Key Cryptography for The Financial Service Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)", X9.62-1998, Sep 20, 1998.