ACCELERATING SIGNATURE-BASED BROADCAST AUTHENTICATION IN DECENTRALIZED WIRELESS SENSOR NETWORKS

[1]NOURAH AL-ANGARI, [2]MZNAH AL-RODHAAN
[1,2]Computer Science Department
College of Computer & Information Sciences
King Saud University
Riyadh, Saudi Arabia
nmalangari@ksu.edu.sa, rodhaan@ksu.edu.sa
E-mail: [1]nmalangari@ksu.edu.sa, [2] rodhaan@ksu.edu.sa

ABSTRACT

Securing the broadcast in wireless sensor networks (WSNs) is crucial due to a wide range of possible attacks in addition to the limited resources in sensor node. Many public-key based solutions have been proposed to address the authentication problem in WSN. Unfortunately the slow in signature verification process cause many weaknesses related to the delay and power consumption. Additionally, the dependency on the base station in the certifying the public keys leads to one failure point and bottle neck problem. Also, the node near the base station will take additional role to reroute the packets to the base station which consume their limited power. In this paper we present a new idea that utilizes the signature-based broadcast authentication to accelerate the signature verification and minimize the base station dependency by allowing sensors sign on other sensor public key.

Keywords: *Public key, Elliptic Curves Cryptography; wireless sensor networks; symmetric; web of trust*

1. INTRODUCTION

Wireless senor networks (WSNs) is composed of a large set of resource-constrained sensor nodes. Where the nodes are deployed in an environment and interact with the physical world. This environment could be unattended or even hostile while the sensors may transmit sensitive data.

Securing the sensitive data is important to assure the confidentiality of the data and authenticate the sensor nodes that participate in the network; this importance is increasing in military and medical applications.

Securing broadcast in wireless networks is an essential issue in many applications (e.g. military). Some environments need to deploy the sensors in unattended or hostile environments. Because of the wireless communication in WSN it is easy to eavesdropping traffics, inject data, modify (alter) message content, or impersonate others. Therefore, there is an essential need to protect broadcast against such attacks.

Many solutions have proposed for the security problem one of them is the cryptography. The first category of solutions is symmetric cryptography, the main drawback of using this cryptography is that it requires secure establishment prior the deployment between the two parities. Additionally, it requires storing (n-1) keys in each node in case of n-network. The second category of solutions is asymmetric which provides higher degree of security than symmetric. However, it requires higher computational time and more power consumption where sensors have limited resource.

This paper aims to provide a secure environment for exchanging messages while conserving the limited resources of the sensors. In this paper we presents a method that utilizes the signature-based broadcast authentication to accelerate the signature verification and minimize the base station dependency by allowing sensors sign on other sensor public key.

This paper organized as follows: in section 2, we reviewed some of the papers that tried to address the security problem using public key cryptography method. In section 3, we provided a background about the elliptic curve cryptography and web of trust showing some work that successfully employed this idea in mobile ad hoc. In section 4, we presented our proposed idea. In section 5, we concluded our work.

2. PUBLIC-KEY

In WSN, broadcast authentication was addressed by µTESLA in [1] with the assumption of trustworthy users. This scheme adopts a one way hash function and uses the hash as keys in a Message Authentication Code algorithm. Generally, all versions of µTESLA schemes buffer messages received within one time interval which makes it vulnerable to certain attacks. Researchers thought about using simplified public key cryptography (PKC) and Curve Cryptography (ECC) to suit such resource limited environment [2, 3].

The authors in [4] presented the first implementation of elliptic curve cryptography in WSN, their results show the feasibility of using public key based in current generation of sensors.

TinyPK that presented in [2] allows the authentication and key agreement between two sensors in addition to the network and third parity. The first version of TinyPK exploited the RSA cryptosystem which is computationally expensive. Their system has a trusted parity Certification Authority (CA) with public and private keys, where its public is known to all network sensors [5, 6]. Also any sensor want to intact must have public and private keys and its public must be signed by the CA's private key. They showed the feasibility of public key even in lightweight sensors. In [7, 8] the authors present a lightweight implementation public key based that uses elliptic curves with 53 bit key. They used an offline pre-computed points table to optimize which save (28%) of RAM.

The authors in [9] present a new algorithm SHESP that utilizes the public key cryptography in WSN. Their algorithm provided confidentiality, by adapting the symmetric RC2. Also, it provided node authentication through asymmetric Elliptic Curve algorithms and data integrity by using MAC-like checksum. They have three entities a Base Station they assumed it is secure against all kinds of possible attacks and so powerful, cluster-heads, and regular nodes. All keys are generated only once and used for all protocol transmissions.

The authors in [10] proposed a very efficient technique for accelerating the signature verification in WSNs. The idea depends on the cooperation among sensor nodes. They allow for some sensor nodes to release their intermediate computation results to their neighbors during the signature verification. So, many of the sensor nodes can accelerate their signature verification process significantly. This technique helps to prevent the nodes from re-executing the same signature verification process and determines whether the received packet should be forwarded to other nodes.

3. PRELIMINARIES

In this section a brief background about the elliptic curve cryptography, then Elliptic curve digital signature algorithm, finally the web of trust.

3. 1 Elliptic curve cryptography

Elliptic curve is an algebraic, where F_p denote a field of integers modulo p which is a prime number. The curve E over F_p satisfy the equation of the form [7]:

$$2 \qquad y^2 = x^3 + ax + b \qquad (1)$$

Where a, b $\in F_p$ satisfy $4a^3 + 27b^2 \neq 0$,

3. 2 Elliptic curve digital signature algorithm

The ECDSA is ElGamal variant scheme which can be described as follow:

1. G is a cyclic subgroup of F_q that is generated by the point P with prime order n and identity element O. and let H:{0,1}*→Z^*_n is a collision resistant function

2. Setup: Singer A will select a random integer d from [1,n-1] then calculate the public key Q = dP which will be published while d is kept as A secret.

3. Generate signature: A will use the private d to sign and it will generate (r, s) for message M in {0,1}*

    - Select a random integer k in [1, n -1], then compute R = kP and take the x-coordinate of R which is r
    - Compute s = k^{-1}(e + dr) mod n, where e = H(M).
    - If r, s in [1,n -1], return (r, s); else, go to Step (3.1).

4. Verifying the signature. After receiving the message M in {0,1}* and the signature (r, s)

from A, a verifier B verifies the signature using A’s public key Q.

- Check that r, s in [1,n-1]. If any fails, return “reject signature”.
- Compute R0 = $s^{-1}(eP + rQ)$ where e = H(M).
- Check the x-coordinate of R’ is equal to r. If succeeds, return ‘accept signature’; else, return “reject signature”.

We selected the Elliptic curve cryptography since it has a significant advantage over the RSA [7]. Based on the experiment conducted by the authors in [9] they found it reduces the required computation time in addition to the amount of data transmitted and stored.

Table 1: Energy cost of digital signature [10]

Algorithm	Sign	Verify
RSA-1024	304	11.9
ECDSA-160	22.82	45.09
RSA-2048	2302.7	53.7
ECDSA-224	61.54	121.98

As seen in Table 1, the real problem with ECC is the verification time that requires much more than sign and even more than the RSA.

3.3 Web of trust

Instead of relying on the CAs to issue the certificates, every entity can certifies the public key of other entities [10]. "Web of trust" module is a decentralized system of trusted introducers same as CA, in web of trust each entity can choose whom to trust and whom not.

It enables anyone to sign on anyone else public key. E.g. A can sign on B's key, so A is introducing B's key to anyone who trusts A.

This idea has been successfully employed in ad hoc networks. Capkun et al [11] proposed a fully organized trust model by using PGP like mechanism. There was no need for central authority in their scheme. The certificates are created by the nodes them self’s and the trust establishment is based on offline trust relationships. The authors in [12] modify the Capkun scheme by using a boot server that initialize the system by distributing the IDs and keys for each entity.

A new style of web of trust has been proposed by [13] where the nodes can create, store, and distribute their public key without any central authority. In network initialization, the nodes share “network private-key” and each node have one private share to use it for certificate signing.

4. PROPOSED WORK

In this paper we adopted the idea presented in [7], but instead of relying on the base station and to avoid having a third party that is responsible of giving the credentials for the users. Here, we suggested using the web of trust (PGP like) as a decentralized to manage the certificates. So, sensors can sign on other sensor public key.

When node A sign on B certificate and sent it to C where C trust A, C verify A’s signature if it correct then it may release one of the scalar multiplication to its neighbors so they also can trust B if they are trusting A.

Hence, there is no need for base station to play the CA role, and the routing to base station is reduced. The nodes do not need to route their information to the base station through other nodes. Thus, minimize the time and power consumption. In addition it removes the bottle neck problem since it reduces the dependency on the base station.

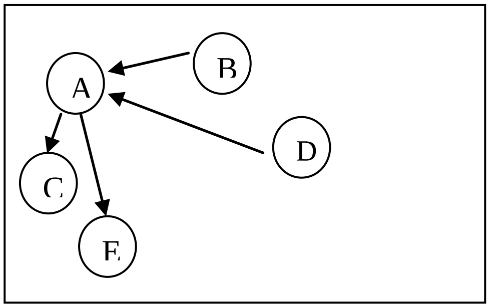

Figure 1: Network example

In Figure 1, B and D trust A also, A trusts C and E. when B need C’s public key A will sign on C’s public key and broadcast it. Once B receive the signature B will verify A’s signature If it pass then it will release one scalar product. D can use it if it trusts A so now B and D can trust C (transitive).

5. CONCLUSION

Many researches showed the importance and the feasibility of using public-key based cryptography in WSN. However, it has the drawback of consuming the limited resources.

In this paper we presented an approach that uses public-key cryptographic scheme for authentication with an acceleration method for the signature verification in order to gain a high security level while preserving the limited resources in WSN. Also, by utilizing the idea of web of trust the dependency on the base-station has been reduced and the authentication process done in a decentralized model. As a future work, we would like to implement the system in a real-time application to evaluate and improve the idea more.

REFRENCES:

[1] D.J. Malan, M. Welsh, and M.D. Smith, "A public-key infrastructure for key distribution in TinyOS based on elliptic curve cryptography," Proceedings of IEEE SECON 2004, pp. 71-80, Oct. 2004.

[2] R. Watro, D. Kong, S. Cuti, C. Gardiner, C. Lynn1 and P. Kruus, "TinyPK: Securing Sensor Networks with Public Key Technology," Proceedings of ACM SASN'04, Washington, DC, USA, pp. 59-64, 2004.

[3] E.O. Blaß, and M. Zitterbart, "Towards Acceptable Public-Key Encryption in Sensor Networks," Proceedings of ACM 2nd International Workshop on Ubiquitous Computing, Miami, USA, May 2005, pp. 88-93.

[4] M. Watfa, M. El-Ghali, and H. Halabi, "A Scalable Security Protocol for Wireless Sensor Networks", The International Conference on security and management, SAM, 8p, 2008.

[5] A. Perrig, R. Szewczyk, J.D. Tygar, V. Wen, D.E. Culler, SPINS, *security protocols for sensor networks*, ACM Wireless Networks 8 (5) 2002, pp.521–534.

[6] X. Fan , G. Gong, "Accelerating signature-based broadcast authentication for wireless sensor networks," Ad Hoc Networks, 10(4) 2012, 10(4), pp. 723–736.

[7] D. Hankerson, A. Menezes, S. Vanstone, *Guide to Elliptic Curve Cryptography*, Springer Professional Computing Series, Springer, 2004.

[8] Lejla Batina, Nele Mentens, Kazuo Sakiyama, Bart Preneel, and Ingrid Verbauwhede. *Low-Cost elliptic curve cryptography for wireless sensor networks.* In Proceedings of the Third European conference on Security and Privacy in Ad-Hoc and Sensor Networks (ESAS'06), Levente Buttyán, Virgil D. Gligor, and Dirk. Springer-Verlag, Berlin, Heidelberg, 2006, pp.6-17.

[9] A.S. Wander, N. Gura, H. Eberle, V. Gupta, and S.C. Shantz, "Energy Analysis of Public-Key Cryptography for Wireless Sensor Networks," Proceedings of PerCom'05, 2005, pp. 324-328.

[10] A Abdulrahman , "The PGP trust model", The Journal of Electronic Commerce, 1997, 10(3), pp. 27–31.

[11] S Capkun S, L Buttyan, J Hubaux , "Self-organized public key management for mobile ad hoc networks. " IEEE Transactions on Mobile Computing, 2003, 29(1), pp. 52–64.

[12] K Ren , T Li , Z Wan , F Bao , R Deng, K Kim, "Highly reliable trust establishment scheme in ad hoc networks", Computer Networks, 2004, 45(6), pp. 687–99.

[13] M Omar, Y Challal, A Bouabdallah, "Reliable and fully distributed trust model for mobile ad hoc networks", In: Computers and Security, 2009.