International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 1 | ISSN : 2456-3307

Multi-User Broadcast Authentication Scheme in Wireless Sensor Networks

Yenumala Sankara Rao[*1], Khareedu Sairam[2]

[*1]Associate Professor, Department of Mca, St. Mary's Group of Institutions, Guntur, Andhra Pradesh, India
[2]PG Students, Department of Mca, St. Mary's Group of Institutions, Guntur, Andhra Pradesh, India

ABSTRACT

Multi-user broadcast authentication is an important security service in wireless sensor networks (WSNs), as it allows a large number of mobile users of the WSNs to join in and broadcast messages to WSNs dynamically and authentically. To reduce communication cost due to the transmission of public-key certificates, broadcast authentication schemes based on identity (ID)- based cryptography have been proposed, but the schemes suffer from expensive pairing computations. In this paper, to minimize computation and communication costs, we propose a new provably secure pairing-free ID-based signature schemes with message recovery, MR-IBS, and PMR-IBS. We then construct an IDbased multi-user broadcast authentication scheme, BASIS, based on MR-IBS and PMR-IBS for broadcast authentication between users and a sink. We evaluate the practical feasibility of BASIS on WSN hardware platforms, MICAz and Tmote Sky are used in real-life deployments in terms of computation/communication cost and energy consumption. Consequently, BASIS reduces the total energy consumption on Tmote Sky by up to 72% and 17% compared with Bloom filter-based authentication scheme based on a variant of ECDSA with message recovery and IMBAS based on a ID-based signature scheme with message appendix, respectively.

Keywords : Broadcast Authentication, Multiuser, Security

I. INTRODUCTION

WIRELESS sensor networks (WSNs) have enabled data gathering from a vast geographical region and present unprecedented opportunities for a wide range of tracking and monitoring applications from both the civilian and military domains [2]–[8]. In these applications, WSNs are expected to process, store, and provide the sensed data to the network users upon their demands [9]. As the most common communication paradigm, the network users are expected to issue the queries to the network to obtain the information of their interest. Furthermore, in wireless sensor and actuator networks [3], network users may need to issue their commands to the network (probably based on the information that they received from the network). In both cases, there could be a large number of users in the WSNs, which might be either mobile or static, and the users may use their mobile clients to query or command the sensor nodes from anywhere in the WSN. Obviously, broadcast/ multicast[1] operations are fundamental to the realization of these network functions. Hence, it is also highly important to ensure broadcast authentication for security purposes.

Broadcast authentication in WSNs was first addressed by μTESLA [10]. In μTESLA, users of WSNs are assumed to be one or a few fixed sinks, which are always assumed to be trustworthy. The scheme adopts a one-way hash function h() and uses the hash preimages as keys in a message authentication code (MAC) algorithm. Initially, the sensor nodes are preloaded with $K_0 = h^n(x)$, where x

is the secret held by the sink. Then, $K_1 = h^{n-1}(x)$ is used to generate MACs for all the broadcast messages sent within time interval I_1. During time interval I_2, the sink broadcasts K_1, and the sensor nodes verify $h(K_1) = K_0$. The authenticity of messages received during time interval I_1 is then verified using K_1. This delayed dis-closure technique is used for the entire hash chain and thus demands loosely synchronized clocks between the sink and sensor nodes. μTESLA was later enhanced in [11] to overcome the length limit of the hash chain. Most recently, μTESLA was also extended in [12] to support a multiuser scenario, but the scheme assumes that each sensor node only interacts with a very limited number of users.

It is generally held that μTESLA-like schemes have the following shortcomings, even in the single-user scenario: 1) All the receivers have to buffer all the messages received within one time interval. 2) They are subject to Wormhole attacks [13], where messages could be forged due to the propagation delay of the disclosed keys. However, here, we point out a much more serious vulnerability of μTESLA-like schemes when they are applied in multihop WSNs. Since the sensor nodes buffer all the messages received within one time interval, an adversary can hence arbitrarily flood the whole network. All the adversary has to do is to claim that the flooding messages belong to the current time interval, which should be buffered for authenti-cation until the next time interval. Since wireless transmission is very expensive in WSNs and WSNs are extremely energy constrained, the ability to arbitrarily flood the network could cause devastating Denial of Service (DoS) attacks. Moreover, these types of energy-depletion DoS attacks become more devastating in a multiuser scenario as the adversary can now

have more targets and, hence, more chances to generate bogus messages without being detected. Obviously, all these attacks are due to the delayed authentication of the broadcast messages. In [13], the TIK protocol is proposed to achieve immediate key disclosure and, hence, immediate message authentication based on precise time synchronization between the sink and receiving nodes. However, this technique is not applicable to WSNs, as pointed out by the authors. Therefore, multiuser broadcast authentication still remains a wide-open problem in WSNs.

When μTESLA was proposed, sensor nodes were assumed to be extremely resource constrained, particularly with respect to computation capability, bandwidth availability, and energy supply [10]. Therefore, public key cryptography (PKC) was thought to be too computationally expensive for WSNs, al-though it could provide much simpler solutions with much stronger security resilience. At the same time, the computation-ally efficient one-time signature schemes are also considered unsuitable for WSNs, as they usually involve intense commu-nications [10]. However, recent studies [14]–[16] showed that, contrary to widely held beliefs, PKC with even software imple-mentations is only very viable on sensor nodes. For example [14], elliptic curve cryptography (ECC) signature verification takes 1.61 s, with 160-bit keys on an ATmega128 8-MHz processor, which is the processor used in the current Crossbow motes platform [17]. Furthermore, the computational cost is expected to decrease faster than the cost to transmit and receive. For example, ultralow-power microcontrollers such as the 16-bit MSP430 from Texas Instruments Incorporated [18] can execute the same number of instructions at less than half the power required by the 8-bit ATmega128L. The benefits of transmitting shorter ECC keys and, hence, shorter messages/ signatures will, in turn, be more significant. Moreover, next-generation sensor nodes are expected to combine ultralow-power circuitry with so-called power scavengers such as Heliomote [19], which allow continuous energy supply to the nodes. At least 8–20 μW of power can be generated using microelectromechanical-systems-based power scavengers [20]. Other solar-based systems are even able to deliver power up to 100 mW for the MICA Motes [19], [21]. These results indicate that, with the

advance of fast-growing technology, PKC is no longer impractical for WSNs, although it is still expensive for current-generation sensor nodes, and its wide acceptance is expected in the near future [15].

Having this observation and knowing that symmetric-key-based solutions such as μTESLA are insufficient for broadcast authentication in WSNs, we resort to PKC for more effective solutions. In this paper, we address the multiuser broadcast authentication problem in WSNs by designing PKC-based solu-tions with minimized computational and communication costs.

II. RELATED WORK

Digital Signature

A digital signature algorithm is a cryptographic tool for gen-erating nonrepudiation evidence, authenticating the integrity and the origin of a signed message. In a digital signature algorithm, a signer keeps a private key secret and publishes the corresponding public key. The private key is used by the signer to generate digital signatures on messages, and the public key is used by anyone to verify signatures on messages. The digital signature algorithms mostly used are RSA [22] and DSA

1. ECDSA is referred to as the elliptic curve digital signature algorithm [24]. While RSA with 1024-bit keys (RSA-1024) provides the currently accepted security level, it is equivalent in security strength to ECC with 160-bit keys (ECC-160). Hence, for the same level of security strength, ECDSA uses a much shorter key size and, hence, has a short signature size (320 bit).

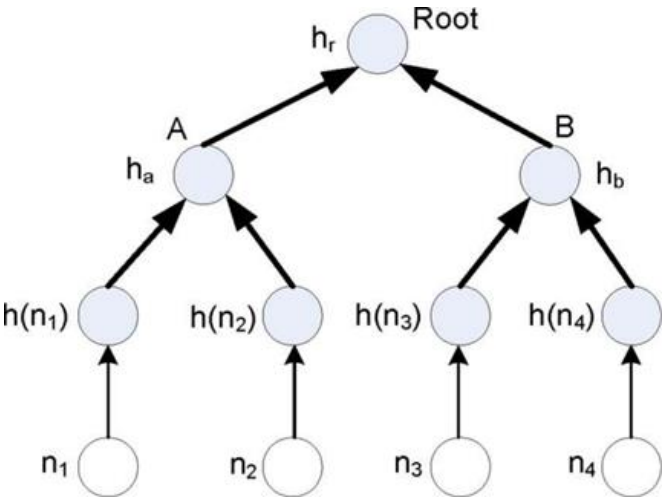


Figure 1. Example of the Merkle hash tree

B. Bloom Filter and Counting Bloom Filter

A Bloom filter is a simple space-efficient randomized data structure for representing a set to support membership queries A Bloom filter for representing a set $S = s_1, s_2, \ldots, s_n$ of n elements is described by a vector V of m bits, which are initially all set to 0. A Bloom filter uses k independent hash functions $h_1, \ldots, h_k$ with range $0, \ldots, m - 1$, which map each item in the universe to a random number uniform over $[0, \ldots, m - 1]$. For each element $s \in S$, bits $h_i(s)$ are set to 1 for $1 \le i \le k$. Note that a bit of V can be set to 1 multiple times. To check if an item x is in S, we check whether all bits $h_i(x)$ are set to 1. If not, x is not a member of S for certain, i.e., there is no false negative error. If yes, x is assumed to be in S. A Bloom filter may yield a false positive. It may suggest that an element x is in S, even though it is not. The probability of a false positive for an element that is not in the set can be calculated as follows: After all the elements of S are hashed into the Bloom filter, the probability that a specific bit is still 0 is $(1 - 1/m)^{kN} \approx e^{-kN/m}$. The probability of a false positive is then $f = (1 - (1 - 1/m)^{kN})^k \approx (1 - e^{-kN/m})^k$. We let $f = (1 - p)^k$. From now on, for convenience, we use the asymptotic approximations p and f to represent the probability that a bit in the Bloom filter is 0 and the probability of a false positive. Let $p = e^{-kN/m}$, respectively.

The counting Bloom filter is a variation of the Bloom filter, which allows member deletion. In the counting Bloom filter, each entry in the Bloom filter is not a single bit but a small counter that tracks the number of elements that have hashed to that location [26]. When an element is deleted, the correspond-ing counters are decremented. To avoid overflow, counters must be chosen to be large enough [26].

C. Merkle Hash Tree

A Merkle Tree is a construction introduced by Merkle in 1979 to build secure authentication schemes from hash func-tions [27]. It is a tree of hashes where the leaves in the tree are hashes of the authentic data values n_1, n_2, . . . , n_w. Nodes further up in the tree are the hashes of their respective children. For instance, assuming that w = 4 in Fig. 1, the values of the four leaf nodes are the hashes of the data values $h(n_i)$, i =1, 2, 3, and 4, respectively, under a one-way hash function h() (e.g., SHA-1 [28]). The value of an internal node A is $h_a = h(h(n_1) h(n_2))$, and the value of the root node is $h_r = h(h_a h_b)$. h_r is used to commit to the entire tree to authenticate any subset of the data values n_1, n_2, n_3, and n_4, in conjunction with a small amount of auxiliary authentication information AAI (i.e., $\log_2 N$ hash values, where N is the number of leaf nodes). For example, a receiver with authentic h_r requests for n_3 and requires the authentication of the received n_3. The source sends the AAI : h_a, $h(n_4)$ to the receiver. The receiver can then verify n_3 by first computing $h(n_3)$, $h_b = h(h(n_3) h(n_4))$, and $h_r = h(h_a h_b)$ and then checking if the calculated h_r is the same as the authentic root value h_r. Only if this check is positive does the user accept n_3. The Merkle hash tree can prevent an adversary from sending bogus data to deceive the client. In the earlier example, an adversary impersonating cannot send a bogus n_3 to the client without being detected. This is because he cannot find h_a and $h(n_4)$ such that $h(h_a h(h(n_3) h(n_4))) = h_r$, as h() is a one-way function.

Broadcast authentication (BA) schemes based on symmetric-key cryptography (SKC) were used.

A SKC-based BA scheme, μTESLA a lightweight version of TESLA achieves source authentication and message integrity by using a one-way hash chain, message authentication code (MAC), loose time synchronization between a sender and receivers, and delayed secret key disclosure.

Ren et al. presented a PKC-based BA scheme, Bloom filter based Authentication Scheme (BAS) built upon a variant of ECDSA with partial message recovery and the Bloom filter, and Hybrid Authentication Scheme (HAS) using Merkle hash tree.

Ren et al. proposed a BA scheme with scalability based on ID-based signature (IBS) scheme from pairings.

Cao et al. proposed a BA scheme based on a paring-free IBS scheme for users' broadcast authentication and a signature scheme with partial message recovery for a sink's broadcast authentication.

μTESLA-like schemes suffer from several active attacks due to the propagation delay of the disclosed keys and the delayed authentication of broadcast messages.

HAS does not provide user scalability, as the Merkle hash tree requires the fixed number of users.

The use of public-key certificates consumes substantial bandwidth and power due to the transmission and verification of the certificates.

IBS scheme is inefficient, as it requires expensive bilinear pairing operations.

III. Proposed System

This paper proposes a new pairing-free IBS schemes with message recovery, MR-IBS and PMR-IBS, which reduce communication overhead and energy consumption of a sensor node. They are the first pairing-free IBS schemes with message recovery

which are provably secure in the random oracle model under the intractability of the ECDLP (Elliptic Curve Discrete Logarithm Problem).

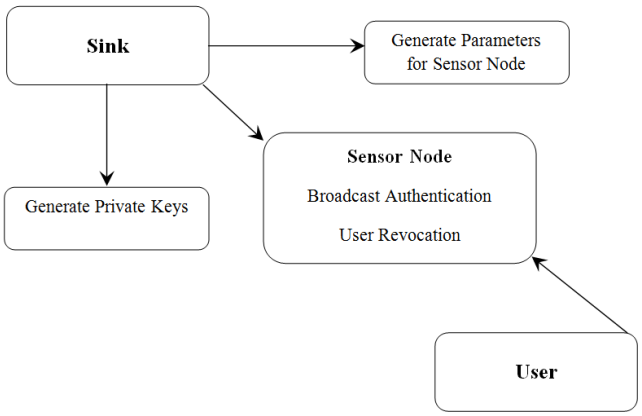
This paper constructed a provable secure ID-based multi-user BA scheme, BASIS, based on MR-IBS and PMR-IBS for broadcast authentication between users and a sink.

It proposes an ID-based multi-user BA scheme, BASIS, based on MR-IBS and PMR-IBS. BASIS consists of four phases: System initialization, Private Key Extraction, Broadcast Authentication and User Revocation.

Advantages:

It reduces communication overhead and energy consumption. It minimizes computation and communication costs.

System Architecture



1. System initialization

In this module the Sink generates system parameters using Setup algorithms of PMR-IBS and MR-IBS

These parameters are preloaded into each sensor node.

2. Private Key Extraction

In this module the sink generates its own private key and users private keys.

For a given user identity ID_i, the sink generates a private key $SKi = (Ri, vi)$ corresponding to $I\ Di$ by performing the Extract algorithm of MR-IBS.

The sink generates its own private key $SKS = (RS, vS)$ for signing, where RS is independent of messages being signed. This invariant value RS can be preloaded into each sensor in the WSN to reduce communication overhead from the sink, but the sink keeps vS secretly.

3. Broadcast Authentication

User Broadcast Authentication

Suppose that a user Ui with an identity I Di wants to broadcast a message mi.

The user Ui with a private key (Ri , si) chooses a current timestamp tti and generates a signature by performing the Sign algorithm of MR-IBS or PMR-IBS. Then the user sends signature to sensor nodes.

The sensor node recovers the message and verifies the signature using Verify algorithm of MR-IBS or PMR-IBS. If it holds, it propagates the message to the next hop. Otherwise, it drops the message.

Sink's Broadcast Authentication

A sink generates the signature using Sign algorithm of MR-IBS or PMR-IBS according to the message length and sends to sensor nodes.

Sensor node verifies using Verify algorithm of MR-IBS or PMR-IBS. If it holds, it propagates the message to the next hop, otherwise, drops it.

IV. CONCLUSION

In this paper, we have studied the problem of multiuser broadcast authentication in WSNs. We have pointed out that symmetric-key-based solutions such as μTESLA are insuffi-cient for this problem by identifying a serious security vulnera- bility that is inherent to these schemes: The delayed authentica-tion of the messages can easily lead to severe energy-depletion DoS attacks. We have then come up with several effective PKC-based schemes to address the problem. Both the computational and communication costs of the schemes have been minimized through a novel integration of several cryptographic techniques. A quantitative energy consumption analysis, as well as security strength analysis, has been given in detail, demonstrating the effectiveness and efficiency of the proposed schemes.

V. REFERENCES

[1]. K. Ren, W. Lou, and Y. Zhang, "Multi-user broadcast authentication in wireless sensor networks," in Proc. SECON, San Diego, CA, Jun. 2007, 223-232.

[2]. I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," IEEE Commun. Mag., vol. 40, no. 8, pp. 102-114, Aug. 2002.

[3]. I. Akyildiz and I. Kasimoglu, "Wireless sensor and actor networks: Re-search challenges," Ad Hoc Netw., vol. 2, no. 4, pp. 351-367, Oct. 2004.

[4]. K. Ren and W. Lou, Communication Security in Wireless Sensor Net-works. Saarbrücken, Germany: VDM Verlag, 2008.

[5]. K. Ren, W. Lou, and Y. Zhang, "LEDS: Providing location-aware end-to-end data security in wireless sensor networks," IEEE Trans. Mobile Comput., vol. 7, no. 5, pp. 585-598, May 2008.

[6]. Q. Wang, K. Ren, W. Lou, and Y. Zhang, "Dependable and secure sensor data storage with dynamic integrity assurance," in Proc. IEEE INFOCOM, 2009, to be published.

[7]. S. Yu, K. Ren, and W. Lou, "FDAC: Toward fine-grained distributed data access control in wireless sensor networks," in Proc. IEEE INFOCOM, 2009, to be published.

[8]. R. Zhang, Y. Zhang, and K. Ren, "DP2AC: Distributed privacy-preserving access control in sensor networks," in Proc. IEEE INFOCOM, 2009, to be published.

[9]. C. Lu, G. Xing, O. Chipara, C. Fok, and S. Bhattacharya, "A spatiotem-poral query service for mobile users in sensor networks," in Proc. ICDCS, Washington, DC, Jun. 2005, pp. 381-390.

[10]. A. Perrig, R. Szewczyk, V. Wen, D. Culler, and D. Tygar, "SPINS: Se-curity protocols for sensor networks," in Proc. MobiCom, Rome, Italy, Jul. 2001, pp. 189-199.

[11]. D. Liu and P. Ning, "Multi-level mTESLA: Broadcast authentication for distributed sensor networks," ACM Trans. Embed. Comput. Syst., vol. 3, no. 4, pp. 800-836, Nov. 2004.

[12]. D. Liu, P. Ning, S. Zhu, and S. Jajodia, "Practical broadcast authenti-cation in sensor networks," in Proc. MobiQuitous, San Diego, CA, Jul. 2005, pp. 118-132.

[13]. Y. Hu, A. Perrig, and D. Johnson, "Packet leashes: A defense against wormhole attacks in wireless ad hoc networks," in Proc. INFOCOM, San Francisco, CA, Apr. 2003, pp. 1976-1986.

[14]. A. Wander, N. Gura, H. Eberle, V. Gupta, and S. Shantz, "Energy analysis of public-key cryptography on small wireless devices," in Proc. IEEE PerCom, Kauai, HI, Mar. 2005, pp. 324-328.

[15]. W. Du, R. Wang, and P. Ning, "An efficient scheme for authenticating public keys in sensor networks," in Proc. MobiHoc, Urbana-Champaign, IL, May 2005, pp. 58-67.

[16]. K. Ren, K. Zeng, W. Lou, and P. Moran, "On broadcast authentication in wireless sensor networks," IEEE Trans. Wireless Commun., vol. 6, no. 11, 4136-4144, Nov. 2007.

[17]. Wireless Sensor Networks, Crossbow Technol. Inc., San Jose, CA, 2004. [Online]. Available: http://www.xbow.com/

[18]. MSP430 Family of Ultra-Lowpower 16-bit RISC Processors, Texas In-struments Inc., Dallas, TX. [Online]. Available: http://www.ti.com