AN EVALUATION OF ENERGY EFFICIENT SOURCE AUTHENTICATION METHODS FOR FALSE DATA FILTERING IN WSN

Syama M and Deepti C

Department of Information Science Engineering,
The Oxford College of Engineering, Bangalore, India.
syama.mk@gmail.com
deeptic82@gmail.com

ABSTRACT

The false data injection attack is a major security threat in Wireless Sensor Network (WSN) since is degrades the network capability. The bandwidth efficient cooperative authentication (BECAN) scheme is used for filtering the false data injection attack. It is used to save energy of sensor nodes in WSN by early detection and filtering of maximum possible injected false data. Source authentication is a critical security requirement in wireless sensor networks to identify attacker nodes that injects false data. Solutions based on Elliptic Curve Cryptography (ECC) have been used for source authentication, but they suffer from severe energy depletion. This results in high computational and communication overheads. Bloom filter based Symmetric-key source authentication scheme exhibits low authentication overhead .This avoids the inherent problems associated with public key cryptography based schemes. The current work demonstrates the efficiency of bloom filter based source authentication using BECAN scheme by comparing ECC and Bloom filter based methods in terms of energy consumption.

KEYWORDS

False data, Elliptical Curve Cryptography, Bloom filter, WSN.

1. INTRODUCTION

Wireless sensor networks are used in a wide range of applications such as battlefield surveillance, traffic control, management of commercial inventory, home monitoring and habitat monitoring [1-5]. Wireless sensor networks are usually deployed at environments where human intervention is difficult. In WSNs, the base station sends commands to the sensor nodes within its coverage area. These nodes report detected events to the base station.

An adversary can easily compromise nodes or inject false data into the wireless communication. These attacks are difficult to detect since the networks are broadcast in nature. An adversary can eavesdrop, intercept, inject, and eventually transmit data. Thus, it is important to ensure that the information transmitted within the wireless sensor networks is sent by an authenticated source. For a false data injection attack, an adversary first compromises several sensor nodes through access of all keying materials stored in the nodes. It then takes control of these compromised nodes to inject false information. This information is then sent to the sink to cause upper-level error decisions. This also leads to high energy wastage in en-route nodes. Sometimes an adversary could also construct a wildfire event or send wrong location information of the wildfire to the sink. Expensive resources wastage will happen by taking rescue actions to a non-existing

DOI : 10.5121/ijsptm.2013.2203

wildfire location. Therefore, it is critical to filter the false data as early as possible in wireless sensor networks. At the same time, if the entire false data flood into the sink simultaneously, the sink will face high verification burdens. This results in rapid paralyzation of the whole network. Therefore, filtering false data should be a high priority task. Few false data filtering techniques [5-10] have been developed for handling this issue. The comparison of these filtering mechanisms illustrates that once a node is compromised, it is difficult to identify the compromised node through symmetric key technique for source authentication. Hence this leads to the degradation of the performance of the filtering mechanisms.

The primary need for avoiding false data injection by an adversary is to authenticate the source. Source authentication defines that a receiver validates that the received data is sent by a legitimate source. There are two approaches which are used for source authentication. The first approach uses symmetric cryptography and the second uses asymmetric cryptography. BECAN scheme based on asymmetric key cryptography is an efficient filtering method for false data injection attack [13]. This paper deals with efficient source authentication using asymmetric cryptography implemented using bloom filter mechanism.

2. RELATED WORK

Recent studies in the area of false data filtering in wireless sensor networks are presented below. Location-Based Resilient Secrecy (LBRS) proposed by Yang et al. [6] adopts location key binding mechanism to reduce the damages caused by node compromise. This mechanism controls the false data generation in wireless sensor networks. Ren et al [7]. Proposes more efficient location-aware end-to-end data security design (LEDS). LEDS includes efficient en-route false data filtering capability and assurance on data availability, but it also requires location-aware key management to achieve en-routing filtering since it is based on symmetric key method. Hence each node should share at least one authentication key with a node in its upstream/downstream cell. Zhang et al [8] presents a public key based solution. It binds private keys of individual nodes to both their IDs and geographic locations. This results in 20 bytes of authentication overheads to achieve en-routing filtering. In bandwidth-efficient cooperative authentication (BECAN) scheme [9] the cooperative bit-compressed authentication technique is used. BECAN scheme is energy efficient by early detecting and filtering maximum of injected false data with minor overheads at the en-route nodes. In addition, only a very small fraction of injected false data needs to be checked by the sink, which thus largely reduces the burden of the sink. But still the energy is wasted since it is using Elliptical curve cryptography (ECC) [10] for source authentication.

ECC based schemes suffer from high energy consumption as well as significant communication and computation costs. A Bloom filter is a space-efficient randomized data structure which represents a set in order to support membership queries. Although Bloom filters allow false positives, for many applications the space savings outweigh this drawback when the probability of an error is sufficiently low.

3. System Design

Bloom filter [11], is a space efficient probabilistic data structure that briefly represents a set that supports queries about membership. It has been used in areas such as web cache sharing [12] and distributed storage system [13] due to its space advantages and excellent distributed properties.

3.1. Properties of Bloom Filter

There is no false negative in the Bloom filter membership verification. An element which tests negative within a Bloom filter is definitely not a legitimate member of the set. On the other hand, Bloom filter may yield false positive, a member outside the set can pass membership verification by the Bloom filter. The probability of a false positive for an element (i.e., false positive probability) can be calculated in a straightforward manner.

3.2. Implementation of Bloom Filter

Bloom filter is implemented as a bit-array consists of ' m ' bits associated with ' h ' different hash functions. Each of the hash function maps an element to one of the m array positions in a uniformly random manner. In an initial Bloom filter all elements are set to 0 (represents an empty set). To insert an element ' a ' into a Bloom filter ' BF ', h array positions are calculated by applying hash functions on e and the bits at those positions in BF are set to 1. When it is required to check the membership of an element ' b ' within the Bloom filter BF, hash functions are applied to the element 'b' which outputs h array positions. If any of this position is having a 0 bit, then element b does not belong to the BF, otherwise the element is a member of the set.

4. Bloom Filter Based Source Authentication Scheme

Bloom filter scheme is based on multi-level key chains in order to enhance scalability in terms of receivers. Two-level key chains are used here. The two-level key chains consist of a high-level key chain and multiple low-level key chains. The low-level key chains are used for authenticating broadcast messages, while the high-level key chain is used to authenticate commitments (or first key) of the low-level key chains. The low-level key chains have short enough intervals so that the delay between the reception and the verification of the messages is tolerable.

In order to use a low level key chain $<ki,0>$ during the time interval Ti, sensor nodes must authenticate the commitment $Ki,0$ before Ti. To achieve this, the sender broadcasts a commitment distribution message (CDMi) during each time interval Ti.

$$CDMi = i \mid ki+2 \mid \mathrm{MAC}\ (k'i|i|ki+2) \mid ki\text{-}1$$

where the ‘|’ symbol denotes message concatenation, and $k'i$ is derived from key ki with a pseudo random function. Commitment Distribution Message (CDM) packets are essential for key authentication. The key ki is generated in time interval Ti.

The sender broadcasts a data packet generated in time interval Ti

$$Pi,j= level\_number \mid index \mid Mi,j \mid \mathrm{MAC}(ki,Mi,j) \mid ki\text{-}d$$

Where *level_number* represents the level of the hash chain, *ki-d* represents the disclosure key in time interval *Ti* that was generated in *Ti-d, d* represents the key disclosure delay.

Bloom filter is used in order to test membership queries. To represent a set *E*= *e1*, *e2*, ..., *en* of *n* elements, a bloom vector *V* of *m* bits can be used .The *m* bits are initially all set to 0. Moreover, this structure needs *k* independent hash functions *h1*... *hk*. These *k* hash functions range between 0 and *m*-1, and each element is mapped to [0, ..., *m*-1]. For each element *e* in *E*, the bits *hi(e)* are set to 1 for $1 \leq I \leq k$.In order to verify if an item *x* is in *E*, tests have to be performed to check whether all bits *hi(x)* are set to 1. If yes, *x* is assumed to be a member of *E*. A Bloom Filter may at times suggest that an element *x* is in *E* even though it is not (false positive). The probability of a false positive *f* is then, $f = (1-e^{-kn/m})^{k}$.

4.1. Network Model

A large spatially distributed WSN consists of one Base Station and a large number of sensor nodes. The sensor nodes have resource-constraints with respect to memory space, computation capability, bandwidth, and power supply.

The Base Station is assumed to be more powerful than sensor nodes in terms of computation and communication capabilities. The Base Station broadcasts queries/commands through sensor nodes. It expects replies that reflect the latest information/measurements. Here it is assumed that the Base Station is always trustworthy but the sensor nodes are subject to compromise.

4.2. Adversary Model

This model assumes that the adversary is able to compromise a limited number of sensor nodes.

4.3. Communication model

The sender sends CDM packets in order to distribute and authenticate the commitments of lower level chains. The jth data packet generated in *Ti* is constructed as follows:
Pi,j= level_number |*index*|*Mi,j* |MAC (*ki,Mi,j*)|MAC(*ki-1,Mi,j*)| MAC(*ki-2,Mi,j*)| …..|MAC (*ki-d-1,Mi,j*)| *ki-d*
In fact, in time interval *Ti*, the sender generates *d* message authentication codes for each data packet, and constructs the set *E.*

E = <MAC(*ki,Mi,j*), MAC(*ki-1,Mi,j*),..., MAC(*ki-d-1,Mi,j*) >

- the elements of *E* (each with |MAC| bytes) are mapped to an *m*-bit vector *V* with *V*= $v_0v_1...v_m$. Therefore *m* is made less tan *d*·|MAC| to reduce the filter size and *m is kept reater tan k·d* to have a small probability of a false positive. *k* represents the number of hash functions used in the bloom filter. The *k* hash functions are known by every node as well as the base station. v_i=1 if there exists a hash function $h_l(\mathrm{MAC}_j)=i$. Figure 1 illustrates how *d* MACs of each data packet are mapped into a bloom filter vector using *k* hash functions.

- Choice of the parameters of the Bloom Filter : Given the number of MACs generated for a data packet 'd' and concatenated in one packet, alon wit the storage space of *m* bits (bloom filter size) for a single Bloom filter, the minimum probability of a false positive *f* that can be achieved is

$$f = (0.6185)^{m/d}$$

Table 1 illustrates the probability of a false positive by applying various values for Bloom filter size and key disclosure delay. Fig. 2 represents the corresponding graphical representation. This probability decreases as the fraction (m/d) increases.

For example m = 16 bits, k = 3 and d = 3, then the minimum probability of a false positive is f = 0,077.

To send a data packet *Pi,j* during a time interval, the sender generates *d* MACs which are then mapped to a bloom filter vector *vi,j*. The data packet sent is then constructed as:

Pi,j = level_number | *index*| *Mi,j* | *vi,j*| *ki-d*

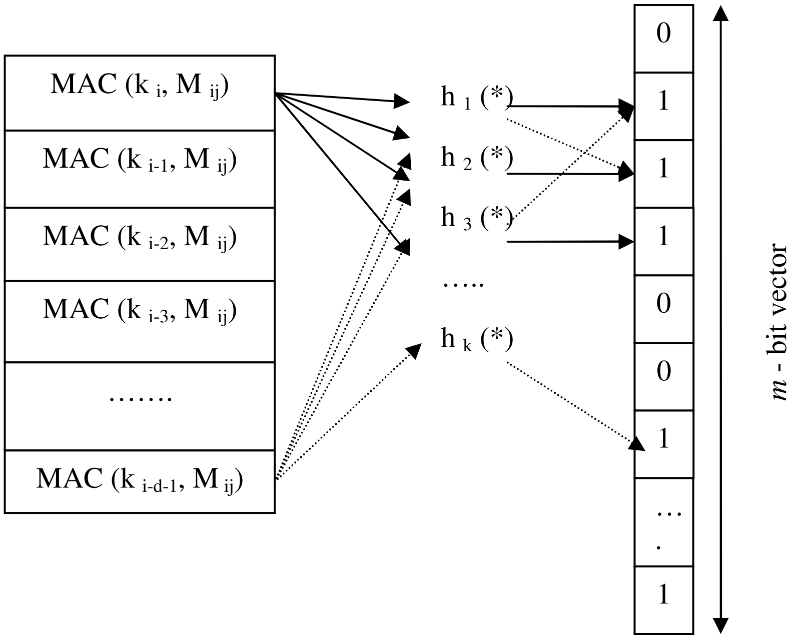
MAC (k_i, M_{ij})
MAC (k_{i-1}, M_{ij})
MAC (k_{i-2}, M_{ij})
MAC (k_{i-3}, M_{ij})
…….
MAC (k_{i-d-1}, M_{ij})
h_1 (*)
h_2 (*)
h_3 (*)
…..
h_k (*)
0
1
1
1
0
0
1
….
1
m - bit vector

Figure 1. Mapping d MACs into a Bloom Filter Vector

On receiving the data packet, the receiver tries to do the following operations:

- Verify if the number of one bits is less than or equal to $d \cdot k$ bits in the vector. If it is not, the packet is dropped else, the receiver computes the message authentication code MAC' with the correspondent key.
- Verify if the computed message authentication code is in the bloom filter *vi,j*. In fact, for each hash function *hi* (with 1≤ *i*≤ *k*) used in the bloom filter, it verifies if *hi* (MAC') is

between 0 and *m*-1. If all the corresponding bits in the vector are set to one, then the packet is assumed to be partially authenticated and it is degraded to the lower levels of the buffer until all the correspondent MACs are verified. Else, the verification fails and the packet is dropped.

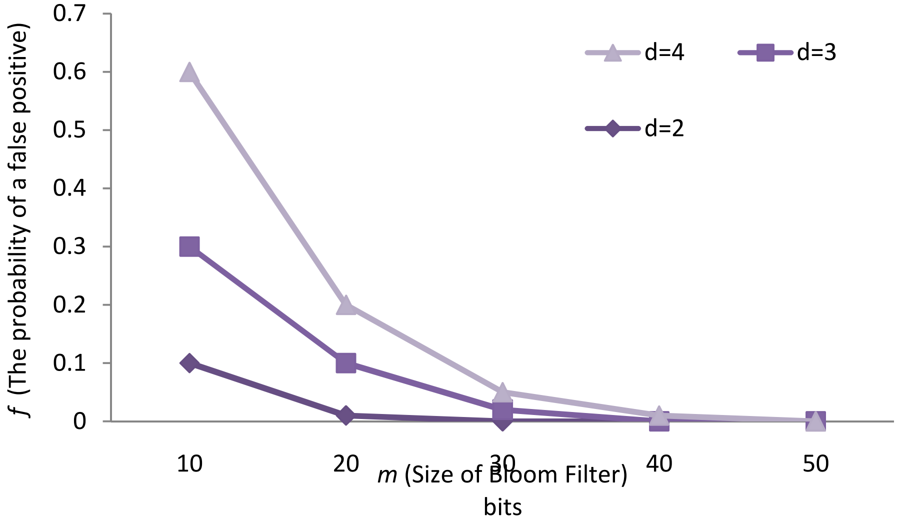

Figure 2. The probability of a false positive with respect of m (bits).

5. Performance Evaluation

A quantitative analysis of the energy consumption due to communicational and computational overhead is carried out here by focusing only on data packets.

5.1. End-to-End Energy

Size of Bloom filter	d = 2	d = 3	d = 4
10	0.1	0.2	0.3
20	0.01	0.09	0.1
30	0	0.02	0.03
40	0	0	0.01
50	0	0	0

Table 1. The false positive probability for different values of m and d.

Total energy consumed for all the protocols is directly proportional to the number of transmissions, which is the sum of the number of data packets sent and the number of control packets sent per node.

Table 2 illustrates the experimental result in terms of energy utilization for the source authentication by ECC based scheme and Bloom filter based scheme on different values of network size. Fig 3 provides the graphical representation of the values obtained in the experiment. It's clearly visible that energy efficiency is very much increased by the usage of bloom filter.

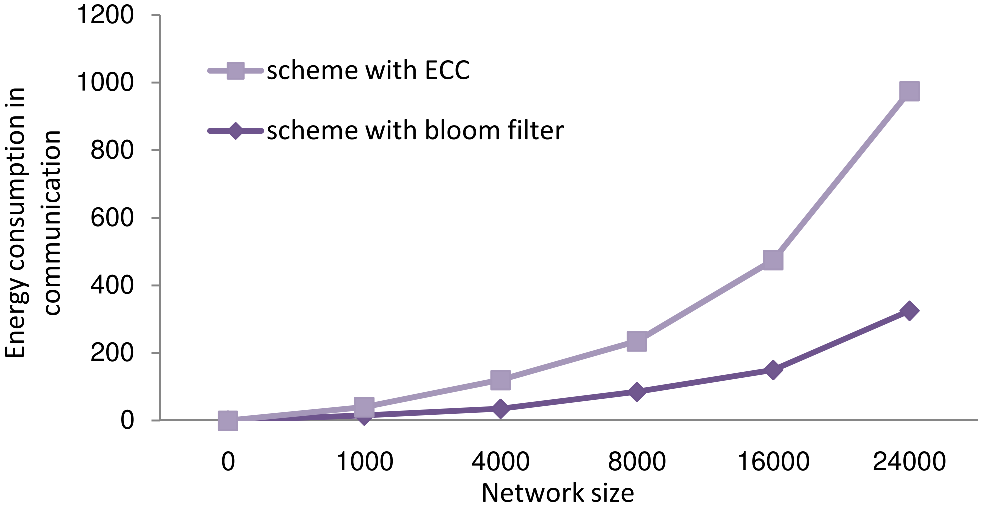

Figure 3. Graph showing energy utilization for ECC and bloom filter based source authentication schemes

6. CONCLUSION

The focus of this paper is the problem of source authentication in false data injection filtering scheme for WSNs. The asymmetric-key based solutions are energy consuming due to the communication and computation cost of the messages which can lead to severe energy-depletion

Network size	Scheme with ECC	Scheme with bloom filter
0	0	0
1000	25	15
4000	85	35
8000	150	85
16000	325	150
24000	650	325

Table 2. Experimental result in terms of energy utilization by ECC based scheme and Bloom filter based scheme on different values of network size.

DoS attacks also. Here an effective symmetric key based scheme using bloom filter is used to address the problem. Communication cost is minimized through a novel integration of a bloom filter and symmetric key cryptography. A quantitative energy consumption analysis demonstrates the efficiency of the bloom filter based scheme.

REFERENCES

[1] Lokesh Sharma, Jaspreet Singh, Swati Agnihotri, "Connectivity and Coverage Preserving Schemes for Surveillance Applications in WSN " , International Journal of Computer Applications (0975 – 8887).

[2] Khalil M. Yousef, Jamal N. Al-Karaki and Ali M. Shatnawi, " Intelligent traffic light flow control system using wireless sensors networks", Journal of Information Science and Engineering 26, 753-768 (2010).

[3] N. Bulusu, J. Heidemann, and D. Estrin, "GPSless low-cost outdoor localization for very small devices", IEEE Wireless Commun., vol. 7, pp. 28-34, Oct. 2000.

[4] Zatout, Y, Campo. E,_Llibre, J.-F, " WSN-HM: Energy efficient Wireless Sensor Network for home monitoring", Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2009 5th International Conference on 7-10 Dec. 2009.

[5] R. Szewczky, A. Mainwaring, J. Anderson, and D. Culler, "An Analysis of a Large Scale Habit Monitoring Application," Proc. Second ACM Int'l Conf. Embedded Networked Sensor Systems (Sensys '04), 2004.

[6] H. Yang, F. Ye, Y. Yuan, S. Lu, and W. Arbaugh, "Toward Resilient Security in Wireless Sensor Networks," Proc. Sixth ACM Int'l Symp. Mobile Ad Hoc Networking and Computing (MobiHoc '05), pp. 34-45, 2005.

[7] K. Ren, W. Lou, and Y. Zhang, "LEDS: Providing Location-Aware End-to-End Data Security in Wireless Sensor Networks," Proc. IEEE INFOCOM '06, Apr. 2006.

[8] Y. Zhang, W. Liu, W. Lou, and Y. Fang, "Location-Based Compromise-Tolerant Security Mechanisms for Wireless Sensor Networks," IEEE J. Selected Areas in Comm., vol. 24, no. 2, pp. 247- 260, Feb. 2006.

[9] Rongxing Lu, Xiaodong Lin," BECAN: A Bandwidth-Efficient Cooperative Authentication Scheme for Filtering Injected False Data in Wireless Sensor Networks", IEEE transactions on parallel and distributed systems, vol. 23, no. 1, january 2012.

[10] Moncef Amara and Amar Siad, "Elliptic Curve Cryptographyand its Applications",2011 7th International Workshop on Systems, Signal Processing and their Applications (WOSSPA).

[11] B. H. Bloom, \Space/time trade-o_s in hash coding with allowable errors," Communications of the ACM, vol. 13, no. 7, pp. 422{426, 1970.

[12] L. Fan, P. Cao, J. Almeida, and A. Z. Broder, \Summary cache: a scalable wide-area web cache sharing protocol," IEEE/ACM Transactions on Networking, vol. 8, no. 3, pp. 281{293, 2000.

[13] F. Chang, J. Dean, S. Ghemawat, W. C. Hsieh, D. A. Wallach, M. Burrows, T. Chandra, A. Fikes, and R. E. Gruber, \Bigtable: a distributed storage system for structured data," in Proceedings of the 7th USENIX Symposium on Operating Systems Design and Implementation, vol. 7. Seattle, WA: USENIX Association, 2006.

Authors

Ms Syama M received her Bachelor of Engineering in Computer Science and Engineering from CUSAT University in 2011. She is pursuing her M. Tech in Computer Networking from Visvesyaraya Technological University. Her area of interest is security in wireless networks and communication in WSN.

Mrs. Deepti C received her Bachelor of Engineering in Electronics and Communication in 2004. She received her M. Tech in Computer Network Engineering with distinction from Visvesvaraya Tec hnological University in 2009.She is a PhD student in Electronics and Communication Engineering at Christ University, Bangalore. Currently she also holds a faculty position as Assistant Professor, Department of ISE, The Oxford College of Engineering. Her main research interests are signal processing, wireless sensor networks and wireless network security.