Secure Data Aggregation Protocol Using AES in Wireless Sensor Network

Pooja Parmar and Bintu Kadhiwala

Abstract Wireless sensor networks (WSNs) in recent years have emerged widely and triggered many active research areas. Data aggregation is one of the important areas to achieve reduction in the number of messages exchanged to improve the average lifetime of WSN nodes. Though there have been many secure data aggregation protocols proposed, there is still need of improvement from a security point of view. The minimum security attributes include confidentiality, integrity, and authentication. In this paper, we propose a protocol that incorporates the minimum security attributes. Our protocol uses AES as the encryption standard to provide tremendous high security. We have used Eclipse, Avrora, and TinyOS for the analysis and simulations. After results were obtained, we concluded that our protocol produces a significant rise in consumption of energy while providing high security. To the best of our knowledge this is a unique attempt that integrates security features in a single protocol.

Keywords Secure data aggregation · Wireless sensor network · Security · Encryption

1 Introduction

Wireless sensor networks (WSNs) have achieved more attention in recent years. There are several applications present nowadays, which are such as military applications, medical applications, environment monitoring, home automation, and so on. Data transmission and reception operation consume most of the energy supply. Often sensed information contains redundant data. Data aggregation is one of the techniques that avoids redundant communication of the sensed data values towards the base station.

Pooja Parmar · Bintu Kadhiwala (✉)
Department of Computer Engineering, SCET College, Surat 395001, Gujarat, India
e-mail: poojaparmar0503@gmail.com

N.R. Shetty et al. (eds.), *Emerging Research in Computing, Information, Communication and Applications*, DOI 10.1007/978-981-10-0287-8_39

SEDAN [1] uses two hop verification mechanisms to ensure integrity and does not need the base station to verify the aggregated readings. The goals of any SDA protocol are confidentiality, integrity, and authentication. SEDAN [1] provides authenticity and integrity but sends raw data readings. Hence it is necessary to integrate confidentiality in the protocol. In this paper we present a highly secure protocol that will incorporate the goals of the SDA protocol. To the best of our knowledge, ours is a unique attempt to achieve confidentiality, integrity, and authentication together in a single protocol.

The paper is further structured as follows: Sect. 2 elaborates the related work and different protocols. In Sect. 3 we propose our protocol and in Sect. 4 we compare it against the existing protocol.

2 Related Work

Aggregation of data is a method of summarizing data coming from different sources using aggregating functions to reduce redundancy within the data transmitted. The protocols here are mainly emphasizing combining the data coming from sensor devices. In context with transmission, we opt to send only summarized data to the base station so that the sensors incur less energy. We observe that for designing a secure data aggregation protocol we need to trade off between security and energy efficiency [2, 3]. There have been various protocols proposed in the past. We have classified some of the protocols into tree-based and cluster-based protocols in Fig. 1.

2.1 *Cluster-Based Protocols*

A secure information aggregation (SIA) protocol was proposed by Przydatek et al. [4, 5]. The approach used was called aggregate-commit-prove. They focused mainly

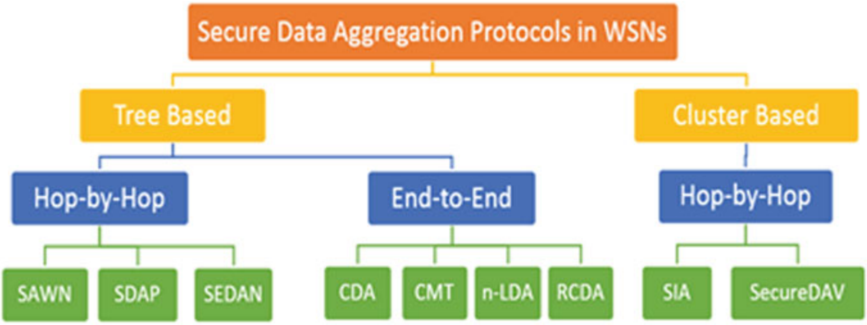


Fig. 1 Secure data aggregation protocols

on an attack called stealthy attack. The framework designed was to provide resistance against these attacks. The attack focused on was such that the attacker makes the user accept false aggregation results although its presence is not disclosed to the user. SIA offers authentication, integrity of data, and confidentiality of data.

SecureDAV was introduced by Mahimkar and Rappaport [6, 7]. It is a secret sharing-based aggregation protocol. It uses an elliptic curve cryptosystem. SecureDAV provides integrity of data, confidentiality of data, and authentication. This protocol supported only one aggregation function: average. Also data validation incurs the major cost of communication.

2.2 Tree-Based Protocols

A protocol named secure aggregation for wireless networks (SAWN) was coined by Hu et al. [8]. They used two concepts of delayed aggregation and delayed authentication. But it revealed keys to verify the integrity of data sent by grandchildren and aggregation of each child. But in this protocol the base station was referred to for the verification phase which ultimately caused a significant delay [1]. Also the sink rejected the aggregated data of the respective branch when it violated the data integrity. This protocol focused on authentication and integrity.

Concealed data aggregation (CDA) was a novel work mainly emphasizing end-to-end encryption in WSNs [9]. Here the protocol worked on cipher text. All the aggregations were done on cipher texts. This protocol assumed that keys are not disclosed to the aggregator and are known to only data source nodes. This protocol focuses only on confidentiality.

Castelluccia introduced “efficient aggregation of encrypted data in wireless sensor networks,” also known as CMT [10]. There was a secret key which was shared between the base station and each sensor node. But this protocol incurred an overhead when the reliability of the network was concerned. Also with respect to scalability it incurred an overhead when the network was huge. This protocol was concerned only with data confidentiality.

Chen et al. introduced “RCDA: Recoverable Concealed Data Aggregation for Data Integrity in Wireless Sensor Networks” [11]. It used homomorphic encryption and homomorphic signature. They used digital signatures. The data here were authentic and integrity was maintained.

“SDAP: A Secure Hop-By-Hop Data Aggregation Protocol for Sensor Networks” was the idea of Yang et al. [12]. They used a technique where they could split the trees in the network where the aggregation was simplified for each node. The disadvantage of the protocol was that the base station had to verify the aggregation. SDAP provided confidentiality of data, source authentication, and integrity of data.

Rodhe et al. proposed “n-LDA: n-Layers Data Aggregation in Sensor Networks” [13]. As the name of the protocol suggests, here the nodes were in the form of different layers and likewise encryption was done. The protocol only focused upon confidentiality [13].

In Ref. [1] Bagaa et al. proposed a new protocol named SEDAN. They used two types of keys inspired by Ref. [8]. This base station could accept an aggregate value immediately without any explicit verification phase. First all one-hop neighbors and two-hop neighbors shared a pairwise key. When a node sent data in response to a query it generated two MACs for authentication: one was for the parent (i.e., one-hop neighbor) and the second was for the grandparent (i.e., two-hop neighbor). When the parent received data it checked the integrity with the MAC designated for it, aggregated all child data, and generated the same signature for the parent and grandparent. It also forwarded child data to its parent. The grandparent of the child upon receiving data checked the integrity of the data with the MAC designated for it and checked the aggregation result correction by comparing the MAC with the generated MAC for the aggregation result by its immediate child node. This protocol provided in-network integrity control and authentication. It prevented bogus data from infecting the global aggregation. An impersonation attack was not possible here. The energy consumption was less than most of the aggregation protocols. In Table 1 we summarize our observations based on the security requirements for SDA protocols.

3 Proposed Protocol

To overcome the constraints in Ref. [1], we propose a new protocol. Our approach uses hop-by-hop confidentiality. The reason for using a hop-by-hop confidentiality approach is that end-to-end confidentiality requires a homomorphic encryption function that restricts aggregation functions often to sum and average. We assure integrity of data by providing aggregation steps computed by two different nodes and comparison of those results. In our framework authentication is achieved by sharing a secret key between two nodes that is derived from the transitory master key.

The proposed protocol has various features which are:

- It provides data confidentiality, data authentication, and data integrity.
- It is a zero configuration protocol; that is, to run it we just need to load the protocol in mote and deploy it in the field and the nodes themselves cooperate to get the aggregation process in working state.
- The base station is able to commit the result without any extra verification phase.

Our protocol is broadly divided into four phases: bootstrapping, aggregation tree construction, key establishment, and aggregation. Of these four phases, the first three phases of bootstrapping, aggregation tree construction, and key establishment

Table 1 Comparison between different aggregation protocols

Parameters	CDA [9]	CMT [10]	n-LDA [13]	RCDA [11]	SIA [4]	SecureDAV [6]	SAWN [8]	SDAP [12]	SEDAN [1]
Data confidentiality	Yes	Yes	Yes	Yes	Yes	Yes	No	No	No
Data integrity	No	No	No	Yes	Yes	No	Yes	Yes	Yes
Data authentication	No	No	No	No	Yes	No	Yes	Yes	Yes
Two-Hop verification mechanism	No	No	No	No	No	No	Yes	No	Yes
Algorithm used	Domingo-Ferrer	Additive homomorphic encryption	Additive homomorphic encryption	Elliptic curve ElGamal	Cryptographic hash function	ECC	–	–	–
Prevention of eavesdropping	Yes	Yes	Yes	Yes	Yes	Yes	No	No	No
Prevention of false packet injection	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes
Prevention of impersonation attack	No	No	No	Yes	–	No	No	–	No
Prevention of replay attack	Yes	No	No	No	–	–	No	–	No
Detection of selective forwarding attack	No	No	No	No	–	–	No	–	No

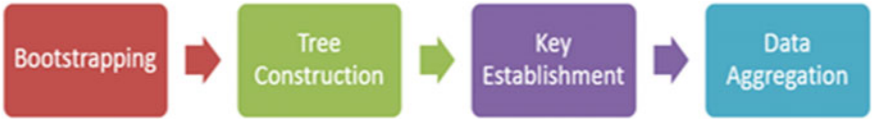


Fig. 2 Phases of protocol

Table 2 Rationale used

Rationale	Description
ID_A	Represents id of sensor node A
$N1_A$,$N2_A$	Nonce generated by node A
F	Any hash function
MK_A	Master key for node A
K_{IN}	Initial key stored in all nodes
$KE_{A,B}$	Key shared between nodes A and B used for encryption/decryption
$KM_{A,B}$	Key shared between nodes A and B used for MAC generation

are set-up phases. These phases are executed at once for set-up. Then multiple rounds of aggregation take place over the constructed aggregation tree. Each phase is described in Fig. 2 (Table 2).

3.1 Bootstrapping

Before deployment each node is preloaded with an initial transitory master key K_{IN}. This key is stored in nonvolatile storage. On boot-up each node first copies this key to volatile memory and then erases this key from nonvolatile memory. After the key is copied to a volatile storage each node performs steps to derive its own master key. Each node A uses Eq. (1) to derive its master key using initial key K_{IN}.

$$\mathrm{MK_A} = F(K_{\mathrm{IN}}, \mathrm{ID_A}) \tag{1}$$

3.2 Tree Construction

Here we assume that each message contains the sender id, destination id, sequence number, and other metadata in the header for each message. Each message described in the protocol contains a header that is not explicitly specified in each message for easy description of the protocol. TreeBeacon, TreeJoinRequest, and TreeJoinSucess only denote message types. Base Station (BS) sends the TreeBeacon message to join it for tree construction. Other nodes in range receive the TreeBeacon message and request to join the tree. On receipt of TreeJoinRequest from any node it sends a TreeJoinSuccess message. The node also adds the accepted node to its child list. It can avoid sending TreeJoinSuccess messages if it already has a number of children up to a defined tree degree.

3.3 Key Establishment

Each node A sends a KeyExchange message to its parent and grandparent. The message designated to the grandparent must be forwarded by the parent. This message contains two nonce generated by node A. This message is encrypted by K_{IN}. Furthermore, MAC is generated using K_{IN}. On receipt of this message from child node A, parent or grandparent node B calculates a shared key with the child.

$$\begin{aligned} \mathrm{MK_A} &= F(K_{\mathrm{IN}}, \mathrm{ID_A}) \\ \mathrm{KE_{A,B}} &= F(\mathrm{MK_A} \oplus \mathrm{MK_B}, \mathrm{N1_A} \oplus \mathrm{N1_B}) \\ \mathrm{KM_{A,B}} &= F(\mathrm{MK_A} \oplus \mathrm{MK_B}, \mathrm{N2A} \oplus \mathrm{N2B}) \end{aligned} \tag{2}$$

After Node B (i.e., parent or grandparent calculated shared key) it sends its own nonce as reply to the child node so it can calculate the same shared key with the parent.

$$\begin{aligned} \mathrm{MK_B} &= F(K_{\mathrm{IN}}, \mathrm{ID_B}) \\ \mathrm{KE_{A,B}} &= F(\mathrm{MK_B} \oplus \mathrm{MK_A}, \mathrm{N1_B} \oplus \mathrm{N1_A}) \\ \mathrm{KM_{A,B}} &= F(\mathrm{MK_B} \oplus \mathrm{MK_A}, \mathrm{N1_B} \oplus \mathrm{N1_A}) \end{aligned} \tag{3}$$

By this calculation the child shares a key with the parent and another key with the grandparent.

3.4 Data Aggregation

Each node of the aggregation tree sends its reading to the parent node and the grandparent node. As each node sends the same readings to the parent node and the grandparent node, both the parent and grandparent can compute the same aggregate value. Each message contains reading and subaggregate values. The subaggregate value denotes the aggregation value for the subtree rooted at that node. The message designated for the grandparent contains two MACs: one generated using the key shared with the grandparent node and other with the key shared with the parent node. Two MACs are required because the grandparent may not be in direct communication range and the parent may need to forward this packet to the grandparent. The parent node receives this packet and checks the integrity and source authenticity by verification of MAC generated with the shared key with the source node. It will compute the aggregate value. Furthermore MAC that is designated for the parent is truncated by the parent node during forwarding of the message to the grandparent. This helps in reduced packet size for transmission. Once a node receives nodes from its child and grandchild it computes a subaggregate of the subtree rooted at that node. It verifies source authenticity and integrity of all messages and decrypts all packets. The grandparent compares the aggregates of aggregation of all grandchildren with the subaggregate field of all child nodes. If they differ, the node rejects that aggregation from the child and recovers the actual aggregate value by aggregation of values from all grandchild nodes. The node may send a notification message about this misbehavior. It also sends an alert message to the parent node that the aggregate value is recovered. Thus the parent node can reject a message from its grandchild nodes to complete recovery.

4 Performance Comparison and Analysis

We have analysed our protocol on the basis of the attacks it overcomes. The comparison between the existing and proposed protocols is presented in Table 3.

We have chosen AES for encryption of the data for the reason that it has been proved to be the most secure algorithm and is the current encryption standard. We analyze our protocol on the basis of the security and attacks possible on the proposed protocol. Here we observe that our protocol is highly secure, whereas SEDAN [1] can be easily attacked. Therefore from Table 3 we conclude that our proposed protocol will give better results than SEDAN [1].

We implemented the proposed framework in TinyOS for MicaZ and TelosB motes. First we used Eclipse for programming the TinyOS application and simulated the protocol in a Cooja simulator. We simulated a system of 50 nodes and one base station. We collected energy consumption of MicaZ motes by using the Avrora simulator.

We have analyzed our results and measured the performance of our proposed scheme. We have analyzed memory consumption and energy consumed. We have also calculated average energy consumption in the CPU, receive mode, and transmit

Table 3 Comparison of security analysis between protocols

Attacks	SEDAN [1]	Proposed protocol
Eavesdropping	No	Yes
False data aggregation	Yes	Yes
Replay attack	No	Yes
Selective forwarding attack	No	Yes
Impersonation attack	No	Yes

Fig. 3 Average energy consumed in transmit mode by protocols

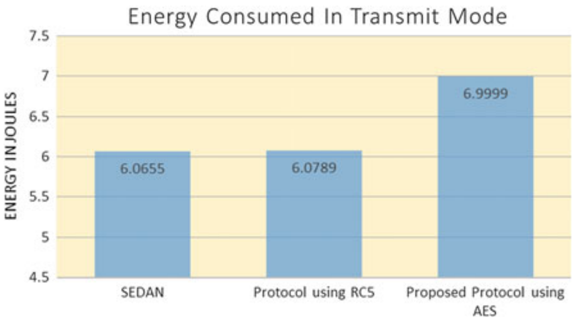


Fig. 4 Average energy consumed in receive mode by protocols

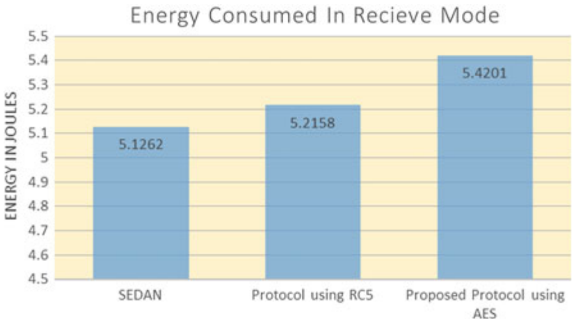
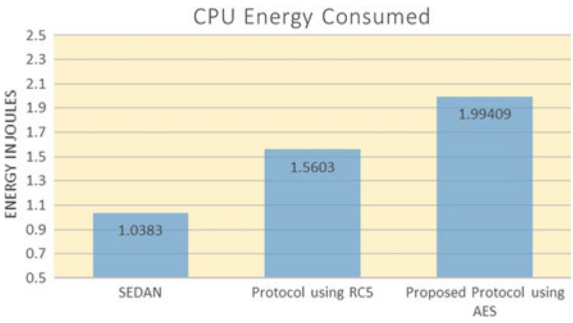


Fig. 5 Average CPU energy consumed by protocols



mode. By the data collected, we conclude that the proposed protocol consumes an acceptable energy rise, at the same time providing high security. The results are shown in Figs. 3, 4, 5, 6, 7 and 8.

Furthermore, in Table 4, we present the benefits of the proposed protocol and the key features.

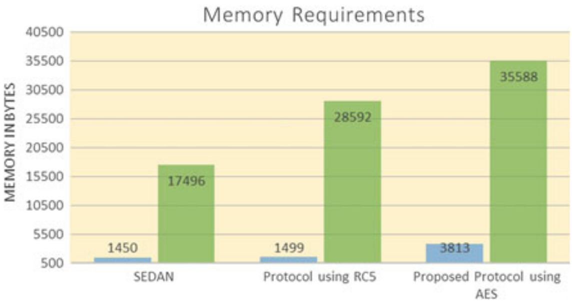


Fig. 6 Memory requirement of protocols

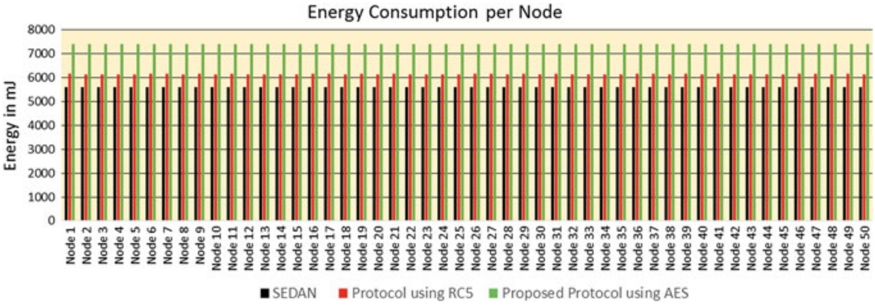


Fig. 7 Energy consumed per node

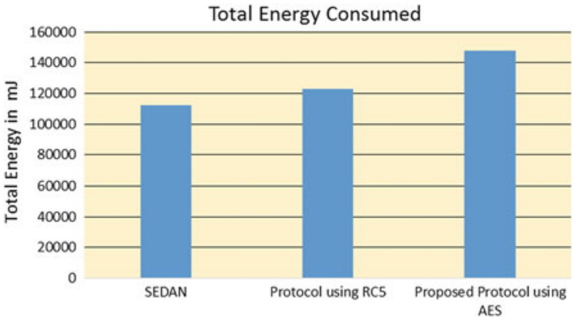


Fig. 8 Total energy consumed by protocols

Table 4 Benefits of proposed protocol

Security schemes	Major features
Tremendous high security	With the use of AES
Eavesdropping attack	Because of symmetric key cryptography
No false data injection	Because of two-hop mechanism use
No replay attack	Using timestamp
No selective forwarding attack	Dropped packets can be detected by two-hop neighbor
No impersonation attack	Because no attacker can capture node's key

5 Conclusion

In this paper we introduce a new protocol providing data confidentiality, authentication, and integrity. To securely exchange the data, we have used the AES encryption algorithm which is more secure and faster than asymmetric algorithms and hence prevents eavesdropping. With this we have used a two-hop verification mechanism that prevents a false data aggregation attack and detects a selective forwarding attack. We have used a timestamp to overcome a replay attack. The results show that our protocol provides a high level of security for aggregated data. We consider this work as a novel step towards incorporating all the basic security in a single protocol.

References

1. Bagaa, M., Challal, Y., Ouadjaout, A., Lasla, N., Badache, N.: Efficient data aggregation with in-network integrity control for WSN. J. Parallel Distrib. Comput. **72**(10), 1157–1170 (2012)
2. Ozdemir, S., Xiao, Y.: Secure data aggregation in wireless sensor networks: a comprehensive overview. Comput. Netw. **53**(12), 2022–2037 (2009)
3. Sen, J.: A survey on wireless sensor network security. CoRR vol. abs/1011.1529 (2010)
4. Przydatek, B., Song, D., Perrig, A.: SIA: secure information aggregation in sensor networks. In: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (2003)
5. Yick, J., Mukherjee, B., Ghosal, D.: Wireless sensor network survey. Comput. Netw. **52**(12), 2292–2330 (2008)
6. Mahimkar, A., Rappaport, T.S.: SecureDAV: a secure data aggregation and verification protocol for sensor networks. In: Global Telecommunications Conference, 2004. GLOBECOM'04. IEEE (2004)
7. Alzaid, H., Foo, E., Nieto, J.: Secure data aggregation in wireless sensor network: a survey. In: Proceedings of the Sixth Australasian Conference on Information Security-Volume 81 (2008)
8. Hu, L., Evans, D.: Secure aggregation for wireless networks. In: 2003. Proceedings. 2003 Symposium on Applications and the Internet Workshops (2003)
9. Girao, J., Westhoff, D., Schneider, M.: CDA: concealed data aggregation for reverse multicast traffic in wireless sensor networks. In: 2005 IEEE International Conference on Communications, 2005. ICC 2005 (2005)

10. Castelluccia, C., Mykletun, E., Tsudik, G.: Efficient aggregation of encrypted data in wireless sensor networks. In The Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2005. MobiQuitous 2005. (2005)
11. Chen, C.-M., Lin, Y.-H., Lin, Y.-C., Sun, H.-M.: RCDA: Recoverable concealed data aggregation for data integrity in wireless sensor networks. IEEE Trans. Parallel Distribut. Syst. **23**(4), 727–734 (2012)
12. Yang, Y., Wang, X., Zhu, S., Cao, G.: SDAP: A secure hop-by-hop data aggregation protocol for sensor networks. In: Proceedings of the 7th ACM International Symposium on Mobile Ad hoc Networking and Computing (2006)
13. Rodhe, I., Rohner, C.: n-LDA: n-layers data aggregation in sensor networks. In: 28th International Conference on Distributed Computing Systems Workshops, 2008. ICDCS'08. (2008)