An Intra-Cluster Trust-Based Secure Data Aggregation Framework for Wireless Sensor Networks

Bhavna Arora Makin
Model Institute of Engineering & Technology, Jammu
and
Prof.Devanand
Department of Computer Science and IT, University of Jammu

The key security issues in the data aggregation for wireless sensor networks (WSN) are data confidentiality and data integrity. In this paper, a framework for secure data aggregation has been proposed. The framework is based on the existing concepts of clusters and key management. We compare the proposed scheme with the existing scheme SCAF (Secure Cluster-based Architecture Formation Scheme) and with the normal aggregation process on various parameters such as packet delivery ratio, throughput and drop. Our results show that the proposed scheme performs better compared to other related schemes.

Keywords: Secure Data Aggregation, WSN Clusters, Trust Evaluation

1. INTRODUCTION

Wireless Sensor Networks (WSNs) comprise an emerging technology which has received significant attention from the research community. A WSN is a self-organizing ad-hoc system comprising several sensors which observe the physical environment. The sensors in the network act as "sources" which detect environmental events and push relevant data to the appropriate subscriber "sinks". Thus, data needs to be forwarded towards the sink node in hop-by-hop manner. If the amount of data which needs to transmitted are reduced, then the energy consumption of the network is also minimized [Dorottya et al. 2007]. A sensor network consists of one or more sinks which subscribe to specific data streams by expressing interests or sending out queries. Several challenges are created due to the characteristics of the wireless sensor networks like limited power resources, processing capabilities and low bandwidth [Gregory et al. 2004] Many sensor systems are deployed in unattended and often adversarial environments. Hence, security mechanisms that provide confidentiality and authentication are critical for the operation of many sensor applications. Providing security is particularly challenging in wireless sensor networks due to the resource limitations of sensor nodes [Akyildiz et al. 2002]

In wireless sensor networks, in order to reduce the medium access layer contention and for conserving energy, data aggregation is considered as one of the fundamental distributed data processing procedures [Zhenzhen et al. 2007].This scheme combines data from different sources and eliminates its redundancy, thus reducing the number of transmissions and saving energy [Bhaskar et al. 2002]. The main idea behind data aggregation is that, rather than sending individual data items from sensors to sinks, multiple data items are aggregated as they are forwarded by the sensor network. Data aggregation is application dependent, i.e., depending on the target application, the appropriate data aggregation operator (or *aggregator*) will be employed.

Security is one of the major challenges for any data aggregation scheme, since many sensor systems are deployed in unattended and often adversarial environments. The following security issues persist in performing data aggregation in a WSN [Sang et al. 2007]:

- **Data Confidentiality:** There is a clear need to protect sensitive transmitted data from the passive attacks such as eavesdropping. Hence, cryptography based solutions are typically employed to alleviate this shortcoming. However, the sensor's power can be used quickly by the complicated encryption and decryption methods like multiplications of large numbers in public key based cryptosystems.
- **Data Integrity:** The integrity of the data in a WSN needs to be ensured even in the presence of malicious or compromised sensors. Since sensor nodes lack expensive tamper-resistant hardware, it is easy for them to be physically altered or reprogrammed.

To overcome these challenges, traditionally two mechanisms have been proposed, namely:

- **Hop-by-Hop Encrypted Data Aggregation:** In this scheme, the transmitted data is encrypted by the sensing nodes and decrypted by the aggregator nodes. The aggregator nodes then aggregate the data and encrypt the aggregation result again. At last the sink node gets the final encrypted aggregation result and decrypts it [Sang et al. 2007].
- **End-To-End Encrypted Data Aggregation:** In this method, the intermediate aggregator nodes do not have the decryption keys and can only do aggregations on the encrypted data. The aggregators aggregate the encrypted sensor readings without decrypting them, so the end-to-end privacy is achieved using homomorphic cryptosystems [Sang et al. 2007].

The framework for hop-by-hop encrypted data aggregation is more efficient than the framework for the end-to-end one, but in the former the sensor readings may be leaked to the adversary if the aggregator is compromised. In this paper, we propose an efficient hop-by-hop protocol based framework that is resilient and secure. It provides both data integrity and confidentiality to the aggregated data and helps in malicious node detection.

The paper is structured as follows. In Section II we study the related background work done by other researchers which forms the basis of our research. In Section III we detail the proposed framework. In Section IV, we present the Cluster Based Secure Data Aggregation Protocol (CBS). Section V presents the performance analysis of CBS in comparison to SCAF and normal aggregation schemes using simulations, while Section VI concludes the paper.

2. RELATED WORK

Several researchers have studied problems related to security of data aggregation in wireless sensor networks. However, most of the existing work is driven by the performance issues without considering the possibility of the presence of attackers. Only in the past few years, have several papers been published which address security issues of the aggregation protocols. In this framework we have used the existing concepts of clustering and key management that have already been proposed by various researchers.

Yang et al.[Yang et al. 2008] have proposed a Secure Architecture Formation Scheme (SCAF) which uses a bidirectional evaluation mechanism to securely form a hierarchical architecture for WSNs. By using the CH-MN evaluation, the malicious nodes are filtered before the network nodes begin to elect the new CHs, thus avoids the malicious nodes becoming the CHs. Further more, even though some malicious nodes are elected to be CHs by little chance, they can be detected and excluded by MN-CH evaluation mechanism.

Bekara et al.[Bekara et al. 2007] have presented a new secure aggregation protocol for cluster-based WSN, which does not require trusted aggregator nodes. Their protocol is resilient to nodes compromising including aggregator nodes, and introduces an acceptable communication and transmission overheads. Their protocol allows the BS to verify the authenticity and the validity of the aggregation results, even if all aggregator nodes and part of the sensors are compromised in the network.

Huang et al.[Huang et al. 2007] has presented a security protocol for cluster based wireless sensor networks combined with an energy efficient clustering algorithm, a secure key management,

and a loosely synchronized broadcasting scheme into a new protocol. He combined the existing researches related to energy efficiency with security service that have been studied separately.

Li Qing et al. [Li Qing et al. 2006] have proposed and evaluated a new distributed energy -efficient clustering scheme for heterogeneous wireless sensor networks, called as DEEC. They proposed that the cluster heads are elected by a probability based on the ratio between residual energy of each node and the average energy of the network. The nodes with high initial and residual energy will have more chances to be the cluster heads than the nodes with low energy.

Pathan et al.[Pathan et al. 2007] have proposed an efficient approach of secure clustering in distributed sensor networks. The clusters or groups in the network are formed based on offline rank assignment and pre-distribution of secret keys. Their approach uses the concept of weakly connected dominating set (WCDS) to reduce the number of cluster-heads in the network. The formation of clusters in the network is secured as the secret keys are distributed and used in an efficient way to resist the inclusion of any hostile entity in the clusters.

The researches done by other researchers till now have usually focused one parameter alone. The main contribution of this paper is the application of a trust evaluation mechanism to the nodes in a clustered environment to effectively track and filter the malicious nodes. The performance of the network under application of the proposed scheme is measured in the presence of malicious nodes and for varying packet sizes.

3. THE PROPOSED FRAMEWORK

3.1 Overview

The proposed security framework is aimed at overcoming the shortcomings of existing solutions proposed by various researchers. Since a single measure on its own cannot provide complete security, there is need for a solution that can overcome multiple hurdles before the network would be compromised. In this paper we propose a framework with the features of key management, secure routing and malicious node detection as a more practical and realistic solution for resource constrained nodes. While these components have been applied independently to achieve certain level of security, we propose their collective deployment as a framework, which is more robust against known attacks providing a high degree of confidentiality and integrity.

3.2 Phases of Framework

The proposed framework has nine phases as follows:

(1) **The Cluster Forming Phase -** In this phase, a WSN is divided into aggregation groups or clusters using a probabilistic grouping technique [Yang et al. 2008] and each node joins only one cluster. In this clustering hierarchy, first the information from the cluster nodes are gathered by the cluster heads. Then it transmits the aggregated data directly to the base station. A hierarchical network with n nodes that are deployed uniformly within $N \times N$ square region is considered. We assume that the network topology is fixed and it does not vary over time.

(2) **The Aggregator Election Phase -** In this phase, one of the sensors will be elected as the cluster head based on energy efficiency of the nodes [Qing et al.2006] These cluster heads act as aggregation points. The clustering is based on the initial energy (E_I) and residual energy (E_R) and all nodes use these values to select the cluster heads. The residual energy E_R can be given as

$$E_R = E_I - E_c \tag{1}$$

where E_c is the consumed energy The average energy (E_A) of i^th iteration is based on parameters such as the message size, the number of clusters, multipath energy, free space energy, data aggregation cost expended in the cluster heads, the average distance between

the cluster head and the base station, and the average distance between the cluster members and the cluster head.

(3) **The Bootstrapping/ Initialization Phase -** This phase focuses on key distribution among cluster heads and nodes. The key distribution consists of three phases: (1) key pre-distribution, (2) shared-key discovery, and (3) path-key establishment [Eschenauer et al. 2002]. In the pre-distribution phase, a large $key - pool$ of K keys and their corresponding identities are generated. During the key-discovery phase, each sensor node finds out which neighbors share a common key with itself by exchanging discovery messages. If two neighboring nodes share a common key then there is a secure link between two nodes. In the path-key establishment phase, a path-key is assigned to the pairs of neighboring sensor nodes who do not share a common key but can be connected by two or more multi-hop secure links at the end of the shared-key discovery phase. The establishment of the path can be through one or more aggregators. When cluster heads and sensor nodes communicate with each other securely, they use keys to identify and validate each other. Various keys can be used from a large key pool of keys that are generated in this phase. In this framework, two keys, namely the Node Key (K) and Network Key (NK) are used. The former is utilized by the individual sensor nodes for the encryption and decryption purposes while the latter is used by the key server node to unicast the node keys to the sensor nodes. When a sensor node transfers data to the cluster head in the cluster, it first encrypts the data using the key k_ch, i (where ch is the cluster head and i is the node). The transmission of the data from the aggregator to the sink takes place by using key k_ch, s (where ch is the cluster head and s is the sink) . The sink can decrypt and read the data using this key. This prevents the adversaries from misusing the data. Sensor nodes agree on the following system parameters. These include Global Key pool and Group Key Pool.

   (a) **Global Key Pool :** Defined as a pool of random symmetric keys from which a group key pool is generated. Keys are generated using one way hash function F, where n is chosen to be large.

$$K_i = F(K_i + 1), \quad i = 1, 2, 3 \ldots n \tag{2}$$

   (b) **Group Key Pool :** Defined as a subset of Global key pool for a given group.

   (c) **Key Ring :** Defined as a subset of group key pool, which is independently assigned to each sensor node. Primarily G_i $(i = 1, 2 \ldots k)$ group key pools are produced using global key pool S. After this, for each sensor node in a group, a key ring from a group key pool G_i is assigned along with a variable.

(4) **Trust Evaluation Phase -** In this phase, trustworthiness of various nodes in a cluster are evaluated. The two major tasks that are performed in this phase are the Trust Evaluation and the Trust Estimation. Each sensor node has a Combined Trust Value (**CTV**) based on the following trust evaluation factors:

   (a) **Identification :** A unique number. is assigned to the node that contains information about the node's location and is a controlling factor in the network.

$$ID_i = SID_i; \tag{3}$$

   where $1 \leqslant i \leqslant k$. k-no.of nodes, SID - Sensor node unique ID.

   (b) **Sensing Result :** This factor represents sensing results for detected events. This factor consists of sensing data and sensing time for the events.

$$SR_i =< SD_i, ST_i > \quad : \text{sensing result value of node} \quad I; \quad \text{where}, 1 \leqslant i \leqslant k., k-\text{no.of nodes} \tag{4}$$

SD_i: sensing data of node i
ST_i: sensing time of node i

(c) **Consistency :** Consistency represents the uniformity and stability of a node in the data gathering process. Based on this factor, we can identify malicious or compromised nodes, and filter out their data. We define a difference threshold DF, which is the upper boundary of the difference between two nodes' sensing results for a same event. i.e If r_1 is the sensed value of sensor s_1 and r_2 is the sensed value of sensor s_2, then the difference d_1 should satisfy.

$$d_1 = abs(r_1 - r_2) < DF. \tag{5}$$

It is assumed that every sensor node knows this threshold DF and makes use of it to check the consistency of its neighbor nodes' sensing data. If the above condition is satisfied, then the sensor node s_1 increments the consistent sensing counters of s_2 by 1. Otherwise it increments the inconsistent sensing counter by 1. Thus, the Consistency Value (CV_i) is given by,

$$CV_i = \frac{CCs_i - ICs_i}{CCs_i + ICs_i} \tag{6}$$

where $-1 \leqslant CV_i \leqslant 1$
CV_i: consistency value of node i, where $1 \leqslant i \leqslant k$,
CCs_i: consistent sensing count of node i
ICs_i: inconsistent sensing count of node i

(d) **Trust Estimation :** Trust estimation involves an assignment of weights to the trust factors that are evaluated and quantified in trust evaluation step. Weight W_i is a weight which represents importance of a particular factor from 0, unimportant, to +1, most important. The weight is dynamic and dependent on the application. Hence the Combined Trust Value **CTV** for node i is computed by the following equation:

$$CTV_i = \frac{W_1 ID_i + W_2 SR_i + W_3 CV_i}{\sum_{i=1}^{3} W_i} \quad \text{where} \quad 0 < W_i \leqslant 1 \tag{7}$$

As the time elapses, trust values for neighbor nodes change dynamically and continuously. If a node makes some trivial and contemporary mistakes in communication or sensing events, such mistakes have little influence on the trust value which is evaluated by its neighbor nodes. Else if a node broadcasts inconsistent data steadily or seldom communicates with its neighbor nodes, trust value for that node is decreasing and convergent to -1. Therefore, some malicious or compromised nodes that broadcast inconsistent or deceitful data continuously can be detected and classified in this phase.

(5) **Secure Data Transmission between Nodes and Cluster Head -** In this phase; the data is transmitted from the nodes to the cluster heads. Since the data needs to be communicated securely, it is required that the keys deployed in the bootstrap phase are used. Each sensor nodes communicate with the cluster head using a symmetric key $K_{ch,i}$. The sensor nodes send the encrypted data using this key to the cluster head. Then the cluster head receives the encrypted data and decrypts the data using the same key $K_{ch,i}$.

(6) **Data Aggregation Phase -** This phase is designed with the goal that within each cluster the desired aggregate value is calculated, in a way that no individual node knows the data values of other nodes. The sensor nodes send encrypted data to the cluster heads. After the encrypted data are received, cluster heads calculate intermediate aggregate values and further aggregate them to the sink. Let $CTV_1, CTV_2, \ldots$ be the initial trust values of the

nodes $n_1, n_2, \ldots$ along the route from a source S to the Sink D. Since the node does not have any information about the reliability of its neighbors in the beginning, nodes can neither be fully trusted nor be fully distrusted. Each cluster head keeps track of the number of packets it has aggregated through a route using a counter (Ct). Each time, when the cluster head CH_k receive data packets along with the trust value CTV_i from a node n_i, then CH_k checks the value of CTV_i. If $CTV_i < CTV_{thr}$, then the data packets from the node n_i will not be aggregated, where CTV_{thr} is the minimum trust threshold value. If $CTV_i > CTV_{thr}$, then it increments the counter Ct_k as,

$$Ct_k = Ct_k + \alpha \tag{8}$$

where α is the number of packets successfully aggregated by CH_k

(7) **Secure Data Transmission Phase to sink -** In this phase the cluster head CH_k generates a random hash value by computing the MAC over the aggregated data and Ct_k with a key shared by the cluster head and the sink and transmits the MAC to the sink D.

$$CH_k \xrightarrow{[MAC(agg, Ct_k)]} D \tag{9}$$

Similarly each cluster head determines its MAC value and finally all the aggregated data reach the sink D.

(8) **Data Verification Phase -** In this phase the data is verified at the sink D. When the aggregated data from all the cluster head reaches the sink, it checks the counters of the cluster heads, before verifying their MAC. The cluster heads are considered as well behaving if the counters are greater than a credit threshold C_{thr}. On the other hand, the cluster heads are considered as misbehaving if the counters are less than C_{thr}. The verifications of the MAC are made only to the misbehaving cluster head instead of verifying all the cluster heads, which reduces the control overhead.

(9) **Malicious node Detection -** After the data verification phase, it becomes quite clear as to which nodes and cluster heads are well behaving and which ones are not. The cluster heads with counters less than C_{thr} are prohibited from further transmissions. Gradually in the process, we can identify the malicious nodes in the network and can prevent them from transmitting further. This reduces the possibility of the network being completely compromised.

4. CLUSTER BASED SECURE DATA AGGREGATION PROTOCOL

The framework can be implemented using a protocol that is referred to as CBS (Cluster Based Secure Data Aggregation Protocol).The overview of the algorithm of the protocol is as under-

5. EXPERIMENTATION AND SIMULATION RESULTS

5.1 Simulation Setup

The performance of our proposed scheme is evaluated through NS2 [Network Simulator-2 2009] simulator. A random network deployed in an area of $350 \times 350m$ is considered. Initially 30 sensor nodes are placed in square grid area by placing each sensor in a 50×50 grid cell. 4 phenomenon nodes which move across the grid (speed 5m/s) are deployed to trigger the events. 4 aggregators are deployed in the grid region according to our protocol. The sink is assumed to be situated 100 meters away from the above specified area. In the simulation, the channel capacity of mobile hosts is set to the same value: 2 Mbps. The distributed coordination function (**DCF**) of IEEE 802.11 is used for wireless LANs as the MAC layer protocol. The simulated traffic is **CBR** with **UDP** source and sink. The number of sources is fixed as 4 around a phenomenon. Table 1 summarizes the simulation parameters used.

Algorithm 1 Cluster Based Secure Data Aggregation Protocol for Wireless Sensor Networks

BEGIN
for all sensor nodes in the network do **do**
  form clusters in the network
  Select cluster head for each node //Each node joins only one cluster
end for
Select the Cluster Head CH_j based on the energy values of the sensor nodes
for each sensor node in the network do **do**
  Measure the identification factor ID_i.
  Measure the sensing result SR_i.
  Measure the consistency value CV_i.
  Compute the combined trust value CTV_i.
end for
for each sensor node **do**
  Compute Symmetric Key $K_{ch,i}$ //using key pool values.
end for
for each sensor node **do**
  Encrypt data along with CTV_i using the symmetric key $K_{ch,i}$
  Send encrypted data to it CH
end for
for each Cluster Head CH_j,$j = 1, 2, \ldots x$ **do**
  Receive the data packet from node S_i,
  Decrypt the data using the symmetric key $K_{ch,i}$
  Evaluate trust value CTV_i
  if $CTV_i < CTV_{thr}$ **then**
    No_packet_aggregation //CH_j will not aggregate the packet
  else
    Aggregate_packet
    $Ct_j = Ct_j + \alpha$ //Increment counter Ct_j, where α is the number of packets successfully aggregated by CH_j
  end if
  Generates random hash value MAC(agg , Ct_j)]
  Encrypt the MAC(agg , Ct_j)] //by the symmetric key $K_{ch,s}$.
  Transmit to the sink
end for
for all aggregated data from CH_j **do**
  Sink decrypts data using symmetric key $K_{ch,s}$. //data from CH reaches the sink
  Check counter value Ct_j.
  if $Ct_j > C_{thr}$ **then**
    CH_j is well behaving
  else
    CH_j is misbehaving //malicious node detection
  end if
end for
Prohibit CH_j from further transmissions.
END

For cryptography, **RC5** algorithm has been used. **RC5** is a fast symmetrical block cipher and is a well-known cipher algorithm used since 1995 without showing any significant weaknesses [Law et al., 2006]. In our simulation a word size of 64 bytes, 128-bit Key length and 12 rounds have been used. The weight values used are w_1=0.4, w_2=0.3, w_3=0.3, while the DF threshold

No. of Nodes	30
Area Size	350 X 350
Mac	802.11
Routing protocol	DSDV
Simulation Time	20 sec
Traffic Source	CBR
Packet Size	100 bytes
Rate	50 bytes
Transmission Range	150m
No. of events	4
Speed of events	5 m/s

Table I: Simulation Parameters

and the CTV_i values have been kept at 100 and 0.5 respectively. We have selected the above parameters keeping in mind that in real life scenarios, the maximum weight age is given to the node's location in the network .In this case, the sensing results and the level of consistency have equal weight age. These parameters are application dependent.

5.2 Performance Metrics

The performance of CBS protocol is compared with the Secured Cluster-Based Architecture Formation (SCAF) [Yang et al. 2008] and non-secure normal aggregation scheme without applying the CBS protocol (hereafter referred as NoCBS). Performance is evaluated as per the following metrics.

- **Average Packet Delivery Ratio :** It is the ratio of the number .of packets received successfully to the total number of packets transmitted.
- **Throughput :** It is the number of packets received successfully per unit time.
- **Drop :** It is the number of packets dropped by the legitimate nodes.

5.3 Results

5.3.1 ***Based on Attackers.*** In this experiment, we vary the number of attackers and monitor the environment keeping packet size as constant. A maximum of 10% of sensor node population have been considered as malicious i.e. upto 3 attackers.

The throughput is the most important parameter to analyze the performance of the network. To get better throughput the errors in the network should be corrected, instead of retransmitting the packet. In this algorithm the attackers create the errors in the network by dropping packets. This greatly affects the throughput of the network. Generally, with higher throughput and lesser no. of packet drops the delivery ratio will increase, but in the presence of more no. of attackers it decreases. Fig.1 gives the packet delivery ratio when the number of attackers increases. It shows that the proposed CBS protocol achieves better delivery ratio when compared to NoCBS and SCAF. Fig.2 shows the throughput obtained with proposed CBS protocol compared with NoCBS and SCAF protocol. It shows that the throughput is significantly more than the NoCBS and SCAF, as the number of attackers increase. Fig.3 shows the results of packets dropped for the increasing misbehaving nodes. From the results, we can see that CBS protocol has lesser no. packets dropped than the NoCBS and SCAF.

By simulation results, we conclude that even in the presence of 3 attackers (i.e 10% of the total no.of nodes), the packet drop of NoCBS is almost 94% ,and that of SCAF is 41% more than CBS. The throughput of CBS is about 57% more as compared to NoCBS and 26% more as compared to SCAF. The deliver ratio of CBS is better than NoCBS by almost 10% and that with SCAF is 7%.

5.3.2 ***Based on Packet Sizes.*** In the second experiment, we vary the packet size as 100, 200, 300, 400 and 500 bytes and keep the no. of attackers constant as 3.

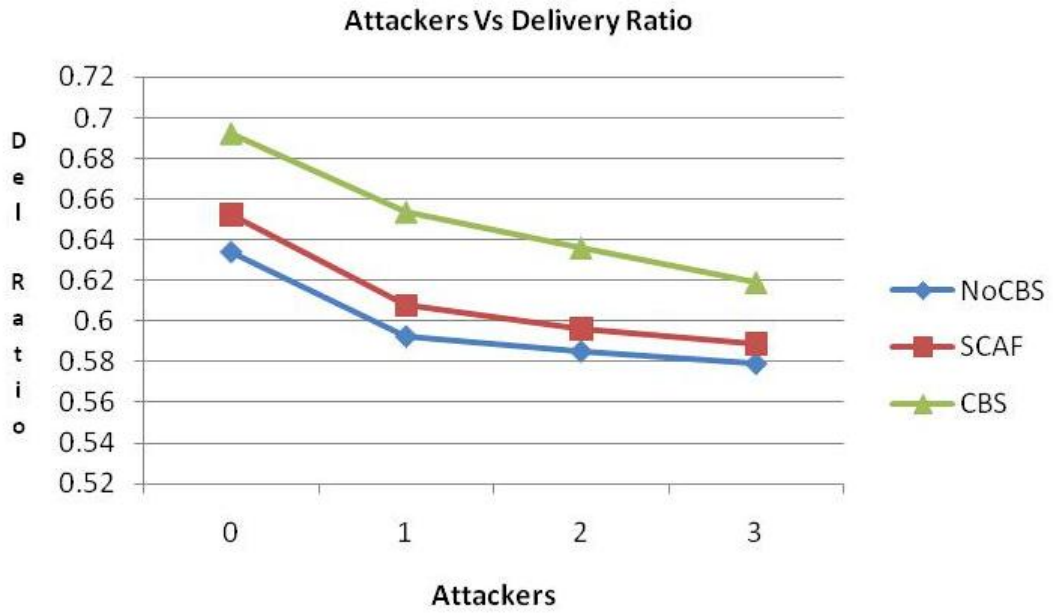

Figure. 1: Number of Attackers vs Delivery Ratio

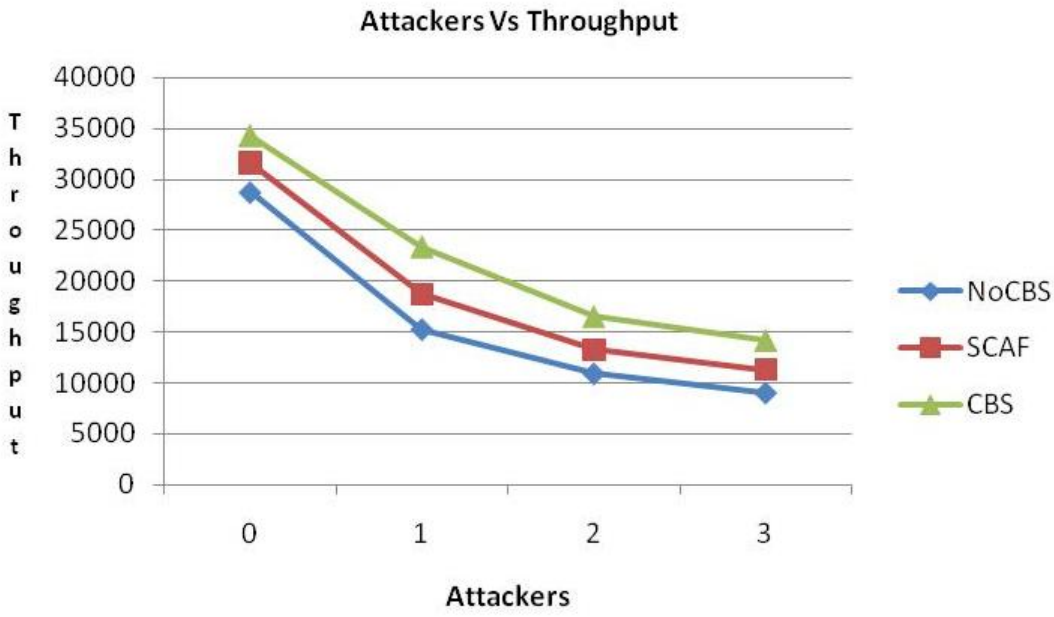

Figure. 2: Number of Attackers vs Throughput

Intuitively, the increase in packet size results in lesser no. of packets sent. If more number of packets is transmitted without loss then the throughput, packet delivery ratio shall increase. Fig.4 gives the packet delivery ratio when the packet size is increased. The throughput degrades as the size of packets sent increases. Fig.5 shows the throughput obtained with the proposed CBS protocol compared with NoCBS and SCAF. Fig.6 shows the no. of packets dropped as the packet size increases, and it is evident from the graph that the least no. of packets dropped is

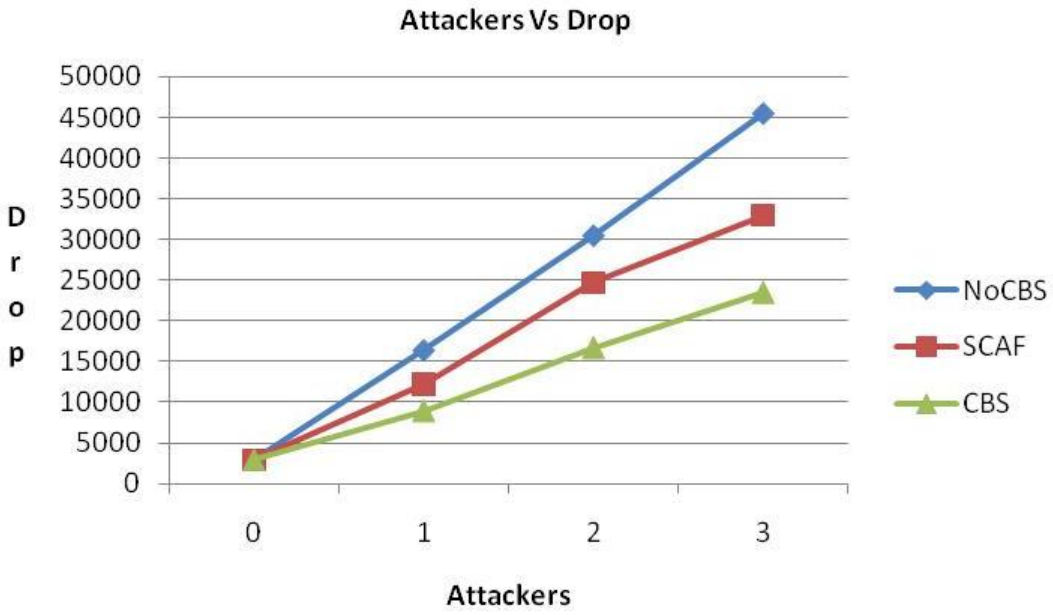

Figure. 3: Attackers vs Drop

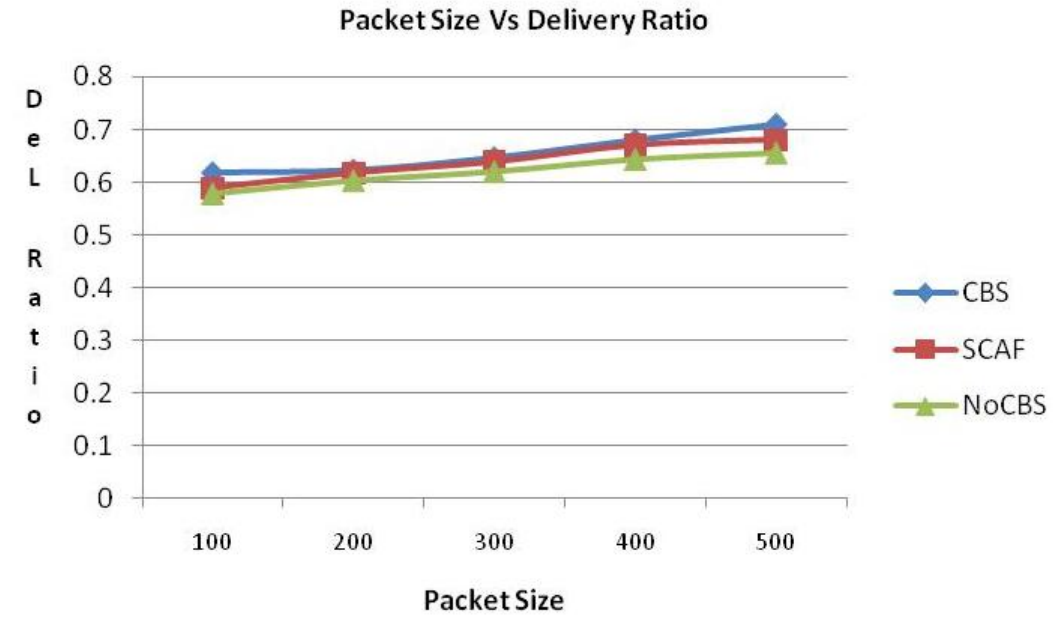

Figure. 4: Packet Size Vs Delivery Ratio

by CBS as compared with NoCBS and SCAF. By analyzing the simulation results, we conclude that in the presence of attackers and at packet size 500 the deliver ratio of CBS is about 9% more than NoCBS ,and about 5% more than SCAF. The throughput of CBS increases by about 59% and 25% when compared to NoCBS and SCAF respectively. The packet drop of NoCBS is approx. 169% more than CBS and that of SCAF is 81% more than CBS

From the above figures, it can be concluded that the proposed protocol CBS has better delivery

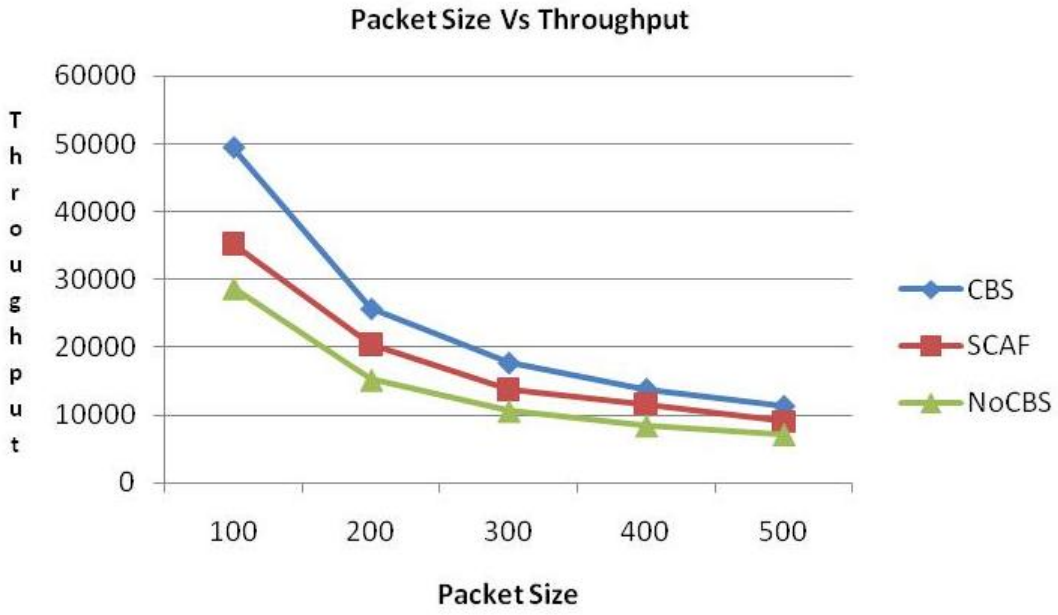

Figure. 5: Packet Size Vs Throughput

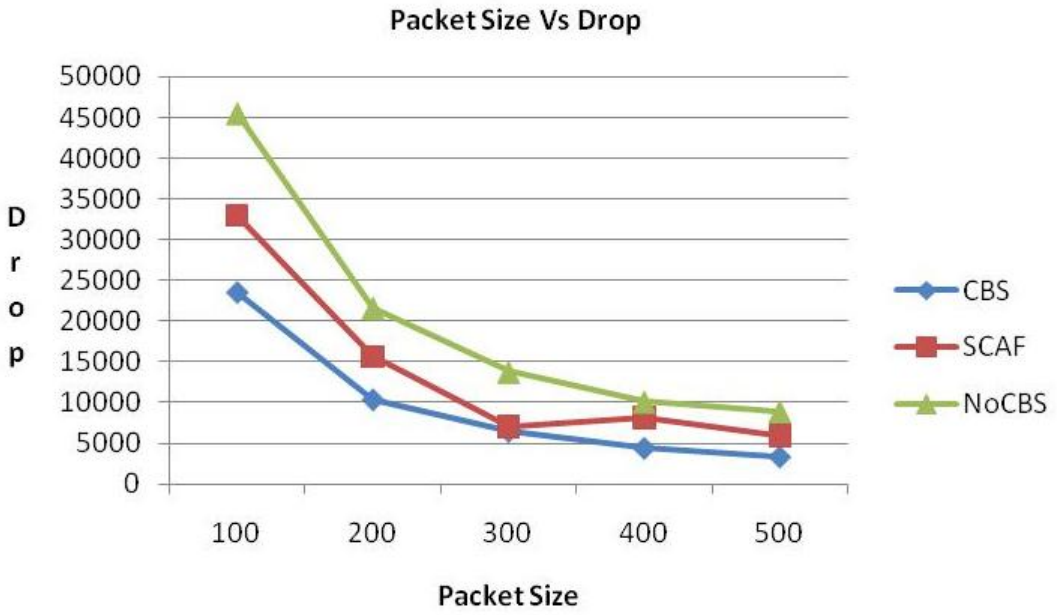

Figure. 6: Packet Size Vs Drop

ratio, throughput and lesser no. of packets drops with increase in packet sizes as compared to SCAF and NoCBS.

6. CONCLUSION AND FUTURE WORK

We have proposed a framework for Wireless Sensor Networks that uses the existing concept of cluster formation and key management for WSN. The early results obtained show that the

proposed framework and the underlying protocol achieves better delivery ratio, throughput and lesser number of packet drops when compared to existing protocols like SCAF and normal data aggregation. While the current work assumes a static network topology, future work can involve extending the intra-cluster trust based secure data aggregation framework to mobile sensor nodes and multiple sinks.

7. ACKNOWLEDGEMENTS

The authors would like to express their gratitude to the anonymous reviewers at IJNGC for their valuable feedback and comments which have greatly helped in improving the quality of this paper. The support extended by the Editor-in-Chief Prof. Vijay Kumar and Managing Editor Dr. Ankur Gupta is also greatly appreciated.

REFERENCES

Akyildiz, I.F. , W. Su, Y. Sankarasubramaniam, And E. Cayirci, 2006. Wireless Sensor Networks: A Survey, Computer Networks,393-422.

Bekara, Chakib And Maryline Laurent-Maknavicius, 2007. A Secure Aggregation Protocol For Cluster-Based Wireless Sensor Networks With No Requirements For Trusted Aggregator Nodes, In Proceedings Of The International Conference On Next Generation Mobile Applications, Services And Technologies.

Bhaskar, Krishnamachari, Deborah Estrin And Stephen Wicker, 2002,The Impact Of Data Aggregation In Wireless Sensor Networks,In Proceedings Of The 22nd International Conference On Distributed Computing Systems.

Culpepper, B. J. ,L. Dung, And M. Moh., Jan 2004, Design And Analysis Of Hybrid Indirect Transmissions (Hit) For Data Gathering In Wireless Micro Sensor Networks, In Acm Sigmobile Mobile Computing And Communications Review, Vol. 8, Pp. 61-83.

Dorottya,Vass, Attila Vidacs, July 2007, Distributed Data Aggregation With Geographical Routing In Wireless Sensor Networks, Pervasive Services, IEEE International Conference.

Eschenauer, L. And V. D. Gligor, Nov 2002 , A Key-Management Scheme For Distributed Sensor Networks, In Proceedings Of The 9th ACM Conference On Computer And Communications Security, Pp. 4147

Fan, Kai-Wei , Sha Liu, And Prasun Sinha, 2007, Structure-Free Data Aggregation In Sensor Networks, IEEE Transactions On Mobile Computing.

Gregory, Hartl, Baochun Li., 2004, Loss Inference In Wireless Sensor Networks Based On Data Aggregation, Ipsn

Heinzelman, W., A. Chandrakasan, And H. Balakrishnan, Vol. 2 Jan 2000, Energy-Efficient Communication Protocol For Wireless Microsensor Networks, 33rd Annual Hawaii International Conference On System Sciences.

Hu L. And D. Evans, 2003. Secure Aggregation For Wireless Networks, Symposium On Applications And The Internet Workshops, pp.384

Law, Y., Doumen, W., J. And Hartel, P., Feb 2006.Survey And Benchmark Of Block Ciphers For Wireless Sensor Networks, Acm Transactions On Sensor Networks 2(1), 65-93.

Li, Qing , Qingxin, Zhu, Mingwen Wang, 2006. Design Of A Distributed Energy-Efficient Clustering Algorithm For Heterogeneous Wireless Sensor Networks, Computer Communications 29, 22302237.

Lindsey, S., And Raghavendra ,C., Mar. 2002. Pegasis: Power-Efficient Gathering In Sensor Information Systems , Ieee Aerospace Conference, Vol. 3,1125-1130.

Lindsey, S., Raghavendra, C. And Sivalingam, K.M, Vol. 13, Sep. 2002.Data Gathering Algorithms In Sensor Networks Using Energy Metrics. IEEE Transactions On Parallel And Distributed Systems, 924-935.

Lu, Huang, 2007, A Security Protocol Scheme For Cluster-Based Wireless Sensor Networks, Graduate School Of System And Information Engineering

Makin, Bhavna Arora And Devanand, 2010, A Trust Based Secure Data Aggregation Protocol For Wireless Sensor Networks Sept 2010.The Iup Journal Of Information Technology, Vol. Vi, No. 3, 7-22

Network Simulator: *http://www.isi.edu/nsnam/ns*

Pathan, Al-Sakib Khan And Hong ,Choong Seon, 2007. Secure Clustering In Dsn With Key Pre-Distribution And Wcds, Ieee Military Communications Conference, Ieee Milcom

Qing ,Li , Qingxin Zhu, Mingwen Wang, 2006. Design Of A Distributed Energy-Efficient Clustering Algorithm For Heterogeneous Wireless Sensor Networks , Computer Communications 29, 2230-2237.

Sang, Yingpeng , Hong Shen, Yasushi Inoguchi, Yasuo Tan And Naixue Xiong, 2006. Secure Data Aggregation In Wireless Sensor Networks: A Survey. Seventh International Conference On Parallel And Distributed Computing, Applications And Technologies.

Yang ,Wencheng , Zhang Yiying, Keebum Kim, Junghwan Kim And Myong-Soon Park, 2008. Scaf: A Secure Cluster-Based Architecture Formation Scheme For Wireless Sensor Network, IEEE International Conference On Circuits & Systems For Communications.

Yang Yi , Xinran Wang, Sencun Zhu, And Guohong Cao, 2008. SDAP: A Secure Hop-By-Hop Data Aggregation Protocol For Sensor Networks, Acm Trans. Inf. Syst. Secur., Vol. 11, No. 4.1-43.

Zhang, Junqi And Varadharajan Vijay, 2008.A New Security Scheme For Wireless Sensor Networks, IEEE "Globecom".

Zhao, L. , Hong,X. And Liang,Q., Nov 2004. Energy-Efficient Self-Organization For Wireless Sensor Networks: A Fully Distributed Approach,47th Annual Ieee Global Telecommunications Conference, Vol. 5, 2728-2732.

Zhenzhen, Ye, Alhussein, A. Abouzeid And Jing ,Ai , 2007 , Optimal Policies For Distributed Data Aggregation In Wireless Sensor Networks, Draft Infocom.

Zhu, S.,Setia, S., Jajodia S.And Ning P., 2004. An Interleaved Hop-By-Hop Authentication Scheme For Filtering Of Injected False Data In Sensor Networks. Ieee Symposium On Security And Privacy,259-271.

Bhavna Arora Makin is an Assistant Professor in the department of Information Technology Engineering at the Model Institute of Engineering and Technology, Jammu. She is currently pursuing her PhD in the area of Wireless Sensor Networks from the University of Jammu, J&K, India. She has over 13 years of work experience in the industry and academia. Her current research interests include network security, wireless sensor networks and mobile agents.

Dr. Devanand is currently the Head and Professor in the Department of Computer Science and Information Technology, University of Jammu, J&K, India. He has done his MSc. in Physics, M.Phil and Ph.D in the area of High Energy Physics from Jammu University. He has about 22 years of research experience. He has participated in various collaboration research projects at CERN-(Geneva),Tufts Univ.-Medford(Boston),University of California - Berkley, TIFR Bombay, VECC-Calcutta. His areas of interest include Computer Simulation and Optimization Techniques.