Authenticating the Data Aggregation in Wireless Sensor Networks using Synopsis Diffusion Method

[1]Asha Bai L, [2]Manjunatha P, [3]Sharath M

[1]M.Tech. (DECS), PG scholar, Dept. of ECE, JNNCE, Shimoga, Karnataka, India
[2]Professor, HOD of ECE, JNNCE, Shimoga, Karnataka, India
[3]Assistant Professor, Dept. of ECE, JNNCE, Shimoga, Karnataka, India

***Abstract*— Wireless sensor networks plays an important role in many applications where the data sensing and transmitting to the sink needs authentication. Applications Wireless sensor networks applications are Volcano and fire monitoring, perimeter surveillance and urban sensing etc . Because of Physical tampering of nodes, information may be compromised and may cause large amount of errors in the aggregation computed at the base station. When a node is compromised it takes more time for transition and causes loss of information. The technologies such as multipath routing, spanning tree have lot of limitations while routing the information. Thus in-network data average technique reduces the communication overhead, time consumption and power consumption[1]. Thus an in-network data aggregator called "Synopsis Diffusion" method, which is robust and scalable against errors caused by the attacker nodes. Thus Synopsis Diffusion algorithm provides security in wireless sensor networks. An "Attack-Resilient algorithm" generates the true aggregation by eliminating the falsified bits in the aggregation hierarchy .**

***IndexTerms*— Data Aggregation, Aggregation Hierarchy, In-network Aggregation, Sensor Network Security, Synopsis Diffusion, Attack-Resilient.**

I. INTRODUCTION

Sensor networks applications are increasing nowadays to control and monitor physical and environmental conditions such as humidity, temperature, pressure etc. There are different applications of Wireless Sensor Networks(WSNs) like military surveillance, wild habitat monitoring, volcano and fire monitoring[2],[3]. Sensor nodes uses multi hop transmission for transmitting the in data and thereby save energy and time. Data is aggregated at some intermediate node and is selected as cluster head. The cluster head selection depends on how that node provides services to all its member nodes and also minimum hops from the BS. In data aggregation technique each node transmits the data directly to the BS, rather than transmitting through an intermediate nodes. Physical tampering of nodes is a common problem occur in network communication. Physical tampering results in a loss of information since it injects the errors. Spanning tree, multipath routing are the different data routing techniques and are not secured over errors caused by compromised nodes. Spanning tree communicate over a shortest path. The disadvantage of spanning tree is that if any node in the path is compromised , the erroneous data only reaches the BS and there is no particular algorithm to find the error bits. Since data is routed in multiple paths in Multipath routing scheme, it results in double counting of data since it is not secure towards count and sum algorithms. The disadvantages of multipath and spanning tree techniques introduce a new technology called as "Synopsis Diffusion" method[4]. For simulation of Nodes, network simulator (NS2) is used.

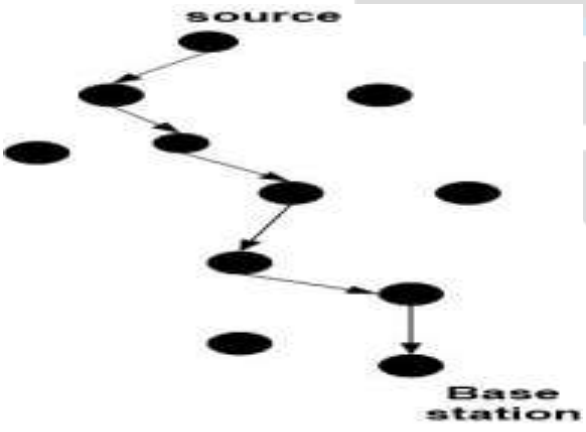


Fig1: Spanning tree

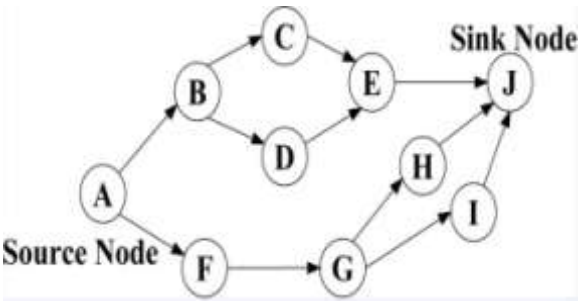


Fig2: Multipath Routing

Network Simulator simulates both wired and wireless networks and is an event driven simulation tool. For wireless sensor mobile ad hoc networks "Ad hoc On-Demand Distance Vector (AODV)" is used as a routing protocol. AODV establishes a route whenever there is a demand from a node, hence it is a reactive routing protocol.

The synopsis diffusion is secure against the erroneous data caused by attacker nodes and attacker node causes errors by launching several attacks like message dropping, eaves dropping, jamming, message fabrication etc. An attack-resilient computation algorithm generates the MAC for each bit in the data thereby computes true average by filtering out the errors caused by the error nodes in the aggregation hierarchy.

II.SYNOPSIS DIFFUSION ALGORITHMS

Synopsis diffusion is a method of getting true average by eliminating errors in the data. Synopsis diffusion is based on the star topology and star topology helps in increased scalability and reduced energy consumption. When all nodes receives aggregation Query from BS, they first arrange in a ring and selects cluster head and cluster selection depends on minimum no. of hops from BS[5]. If the node is at T_j means that the ring consist of the nodes which are j hops away from BS. In every aggregation period, each node generates and broadcasts a self synopsis and it starts from the outermost ring. The synopsis generation SG(v), where v is the value of a sensor relevant to the BS's current query. A cluster head in each round computes the synopsis or fused synopsis SF() by performing logical OR of data received from its child nodes and then broadcast the updated synopsis. The fused synopses is broadcasted to BS level-by-level. Finally, BS uses the synopsis evaluation function SE() to translate the final synopsis as a answer to the query. Residual energy of the neighbouring nodes are accessed first and higher energy node is selected as CH and CH changes every time since energy of the node decreases every time.

When node receives hello message, it checks its id and the parent id mentioned in the hello message. If both ids matches, then CH adds that node to its member list. Each node is aware of child and its member list. The synopsis diffusion algorithms are based on Flajolet and Martin's probabilistic algorithm[6] . Each node X generates a information called as synopsis Q^X which is a binary vector. The different functions of the synopss diffusion method are as follows.

1. **Synopsis Generation**: A synopsis generation function SG(.) converts sensed value into binary data (111101010000).
2.**Synopsis Fusion:** A synopsis fusion function SF(.,.) performs logicl OR operation of two synopsis and generates new synopsis.
Eg: Node R's synopsis is 111101010000, S's synopsis is 111110011000 ,then the synopsis fusion function will be 11111101100.
3.**Synopsis Evaluation:** Synopsis Evaluation function translate the binary information into the sensed final answer which is at the BS.

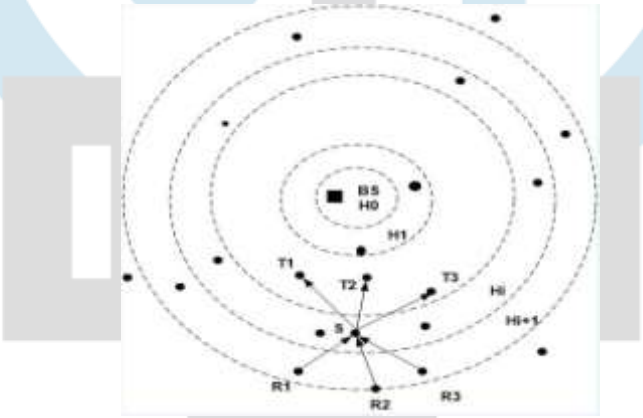


Fig3: Synopsis Diffusion over a Ring Topology.The base station (BS) is at ring H_0.

Definition: A node S's fused synopsis, B^S, is recursively defined as follows. If S is a leaf node that is present in the outermost ring, B^S is its self synopsis Q^S. If S is a non-leaf node, B^S is the logical OR of S's local synopsis Q^S , with S's children's fused synopsis.
Let B^{R1}, B^{R2},……. , B^{Rd} are the synopses received by node S from d child nodes R1, R2, ….., Rd respectively, then S computes the fused synopsis B^S as follows:

$$B^S = Q^S || B^{R1} || B^{R2} || \ldots\ldots || B^{Rd}$$

where || is the bitwise logical OR operator. Note that B^S represents the sub-average of node S, including its derivative nodes. B^{BS} is same as the final synopsis B which is at the BS. The different algorithms of synopsis diffusion method are given below.

A .Count

In Count algorithm each node generates a self synopsis denoted as Q^X, which is a binary vector of length η>logN', where N' is the upper bound on count. CoinToss(X,η) will be performed in order to generate the self synopsis of each node. To generate the self synopsis of a node it has to perform coinToss(X, η), where X is the node's identifier and total length η. Count algorithm uses Hash function to generate the MAC or security key for each '1' bit in the data. CoinToss(X, η) returns the total number of iterations, until the first head occurs or η+1 if all of η tosses have been tail. The synopsis generation function of the count algorithm generates self synopsis Q^X and the bits for which MAC will be generated set as 1 while all others set as 0.Thus Q^X is a bit vector of the form $0^{(i-1)}.1.0^{(\eta-i)}$.... with probability 2^{-i}. The i^{th} bit of Q^X for which correct MAC received is set to '1' while all other bits are '0',thus Q^X is a bit vector of the form $0^{(i-1)}.1.0^{(\eta-i)}$ with probability 2^{-i}.

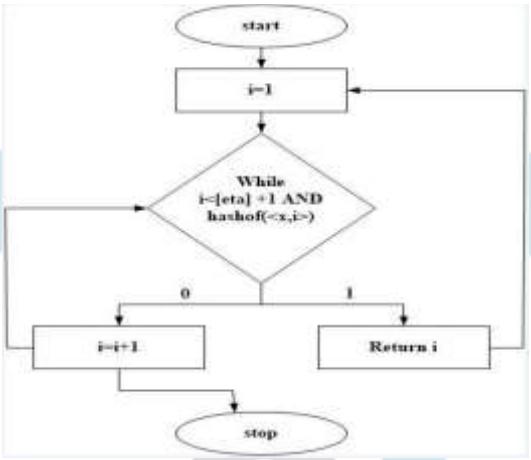


Fig4: Count Algorithm

Then here the case considering is changing '0' become '1' and not '1' as '0'. Bs can count the no. of nodes whose value is more than 10 units, using synopsis evaluation SE() function. Thus no. of nodes in the network can be counted by $2^{Z-1}/0.7735$. While computing coinToss(X,η), no node in the network has set the z^{th} bit there by the number of sensor nodes is proportional to 2^{Z-1}.

B.Sum

Count algorithm counts the no. of nodes whose value is greater than 10 units but sum algorithm aggregate the data received from all cluster heads. Sum algorithm computes aggregate at BS and aggregated data is then transformed into final answer or sensed data by synopsis evaluation function. Synopsis fusion SF() and synopsis evaluation SE() are same as that of count, where as synopsis generation SG() of sum algorithm is different from that of count algorithm. Sum algorithm generates the self synopsis Q^X which is also the sensed value V_X, Sum has to perform coin toss function V_X times and thereby verifies the Message authentication code for each '1' bit of the data. In the i^{th} supplication ($1 \le i \le V_X$), node X executes the function CoinToss(X_i, η) where key_i is constructed by adding node's ID and integer value. The process of Sum algorithm is given in below flow chart.

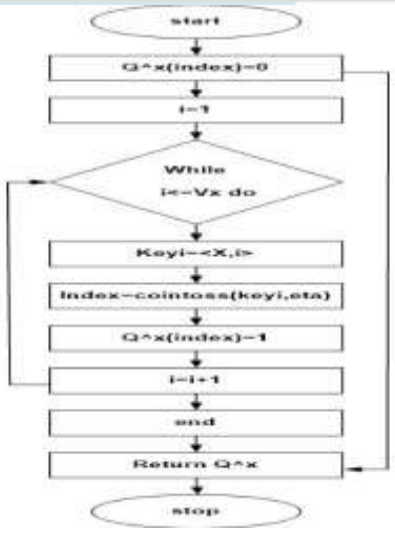


Fig.5: Sum Algorithm

III.ATTACK RESILIENT PROTOCOL

An Attack resilient algorithm generates MAC for each 1 bit in the information and there by verifies each bit and computes the true average by eliminating the falsified errors at BS.

An Authentication Mechanism

A attacker node X sends the falsified sub-average attack by inserting one or more false 1's in its fused synopsis. There is an obvious solution towards these kind of attack is as follows. BS broadcasts an average query message which includes a initial value "Seed" associated to the current query. In the resultant aggregation phase, along with the fused synopsis B^X, each node also sends a MAC to BS authenticating its each bit of sensed value V_X. MAC is computed by considering Seed value and node's own ID[9]. Thereby, BS is able to detect and filter out any false '1' bits which generates false MACs while answering the query. Specifically, if node X contributes to bits $b_1,b_2,.....,b\zeta$ in its self synopsis Q^X, it generates a MAC, M = MAC(K_X, L), where K_X is the key that node X shares with BS that may be a genuine key or malicious key and the content of L is $<X,V_X,b_1,b_2,.....,b\zeta,Seed>$. Each node X sends a message (L',M), where $L'=< X,V_X,b_1,b_2,.....,b\zeta >$ might be needed by BS in order to regenerate the MAC which is required for the verification.

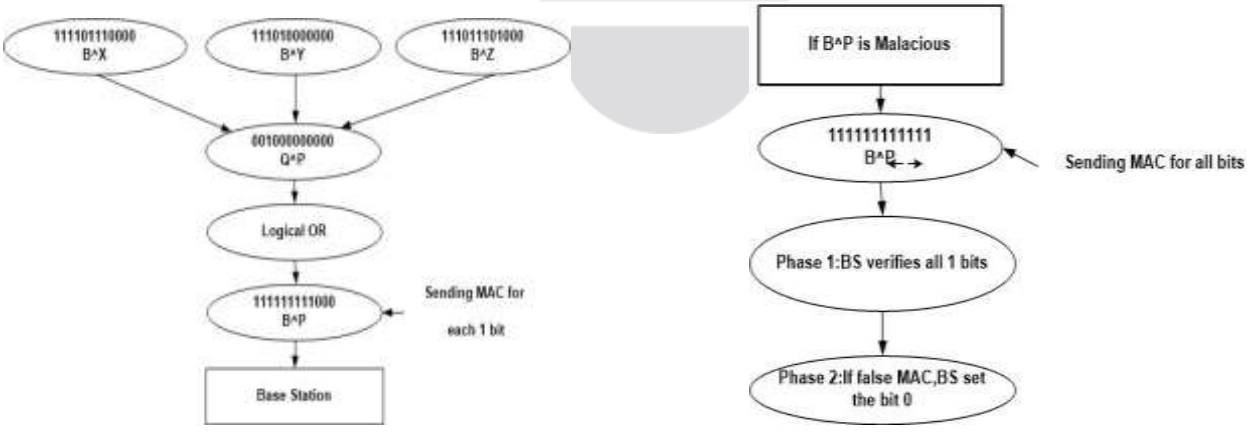
An attack-resilient algorithm reduces total number of MACS and thereby overhead. Throughout this dissertation, MAC means that the corresponding L' is attached to M[7]. False MAC can be associated either to a false '1'or to a non-false '1' bit. Specifically, An attacker node can generate a pseudo MAC in four ways—(i) by injecting error at L, (ii) by altering the key K_X, (iii) by falsifying L and K_X ,(iv) by randomly sending bogus array of bits. As BS re-executes the MAC of each bit for each received MAC, any false MAC will be detected by BS and will be discarded from the synopsis. M_i^X denotes the MAC generated by node X, authenticating the i^{th} bit of its self synopsis Q^X. Note that M_i^X is required to be generated only if $Q^X[i] = 1$, which means that no MACs are generated for 0 bits. As an example, if two nodes X1 and X2 set bit i to be '1' in their self synopsis, then M_i corresponds to either M_i^{X1} or M_i^{X2}. The nodes aggregate their self synopsis with the synopsis obtained from child nodes and send some authentication messages to BS after receiving the query message in the following two phases.

1**) Phase 1:** BS broadcast the initial query message BS is as follows:
BS→: ("Phase1", "Sum",Seed, η), where "Phase1" is an initialization of first query, and η is the synopsis length.

2)**Phase 2:** BS requests the sensor nodes which are reason for bits i, i>r^, in the synopsis to send back the corresponding true MACs. The message sent by BS to corresponding nodes is as follows:
BS→: ("Phase2", r^),
where "Phase2 indicates that phase 2 is going to initialize.

Example

Let us take an example to explain the whole process of synopsis diffusion algorithm . Let the child nodes be X, Y, and Z of a CH node P. In phase 1, each node is supposed to send MAC for each '1' bit in the synopsis while sending the information thereby authenticating each bit. So, X sends 111101110000 as its synopsis, and few MACs corresponding to each '1' such as M_i, $1\le i\le 4$, M6, M7 and M8. X may have received some of these MACs from its child nodes and not sure whether the received MACs are true or false. Likewise, Y sends 111010000000 with MACs M_i, $1\le i\le 3$, M5 and Z sends 111011101000, M_i, $1\le i\le 3$, $5\le i\le 7$ and M9. If P is not compromised, then P should compute its fused synopsis as 111111111000, and should randomly choose single MAC for each '1' bit from the received MACs. Thus P must forward Mi, $1\le i\le 9$. However, if P is injecting some errors, then it forwards 111111111111 as its fused synopsis, and M_i, $1\le i\le 12$, where M10, M11, and M12 are false MACs. BS will be able to detect that M10, M11, and M12 are false MAC, if no node is able to send a valid MAC for these three bits. Furthermore, for the 9^{th} bit the MACs received by BS from different nodes are false[8]. Thus, BS's estimation r^ can be 8 and it requests the network to start the second phase. In phase 2, the node which generates and sends valid MAC for bit 9, 10, 11 or 12, and all of these MACs should reach BS and thereby BS can decide the correct status of each bit in the synopsis.



IV. SIMULATION RESULTS

The simulation study showed the performance and correctness of the Synopsis Diffusion and attack resilient algorithms. The evaluation of results is done based on performance metrics, such as closeness of the simulation results to the theoretical predictions and communication overhead.

Results and Discussion

Nodes are initialized given with different IDs. Nodes are initialized with different energies and set the topology. Node 0 acts as Base station. Routing protocol uses distance property to select the neighboring nodes. Residual energy of all nodes are accessed, high energy node is selected as cluster head and it is attached with hello message.

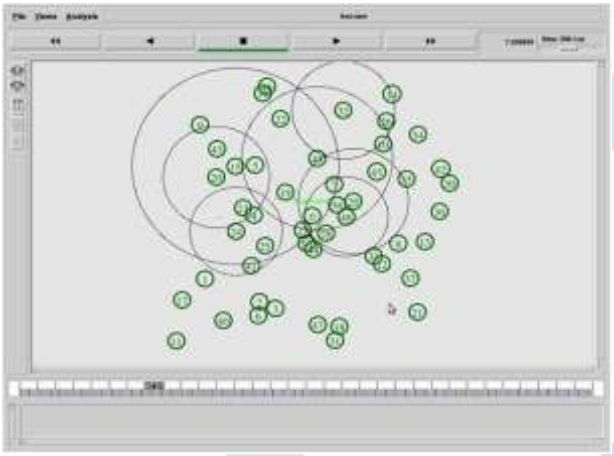


Fig6: Node initialization

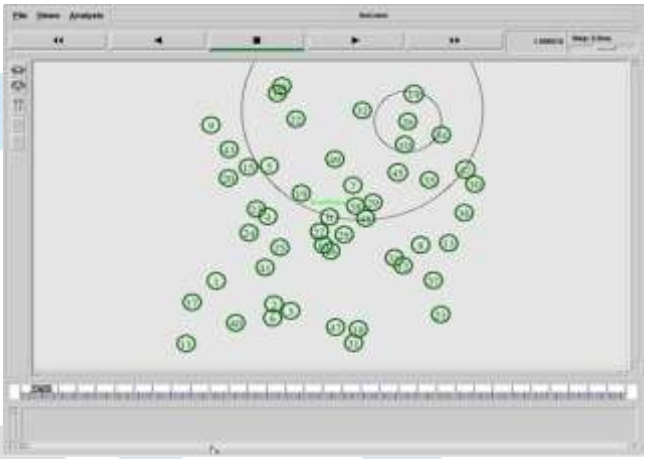


Fig7: False Data Injection Attack by Sensor Member

When child nodes receives hello message, first checks its id and the parent id mentioned in the hello message. If both cluster head id and child id matches, cluster head adds that node as child node in its member list. Each node is aware of child and its member list. Bit-string of sensed temperature is computed for every sensor and transmitted to parent (CH). Every sensor generates Message Authentication Code(MAC) for each '1' bit using its genuine key.

Cluster Head may also be an attacker node. BS verifies received MAC and filters an unauthenticated bit from final fused synopsis. Initially BS verifies the cluster head temperature . If cluster head node is an attacker then BS collects the temperature of all sensor nodes about temperature of cluster head node. CH calculates the average temperature of sensor members.

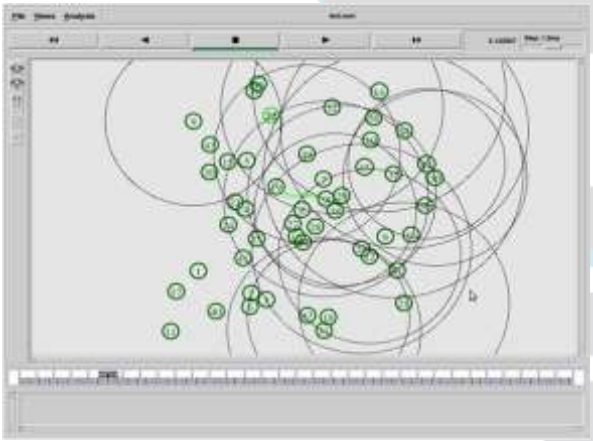


Fig8: False Data Injection Attack by Parent

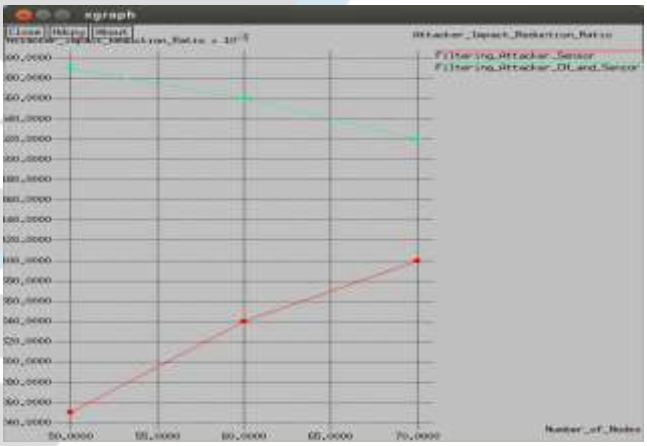


Fig9: Attacker Impact Reduction Ratio

Then, calculates the difference between cluster head temperature and average sensor temperature. If the difference temperature value greater than the threshold then it is an attacker node. Otherwise, an average temperature for BS will be calculated.

An attacker impact reduction ratio is decreased with increased number of nodes. The filtering attacker cluster head provides increased reduction ratio compared to filtering attacker sensor.

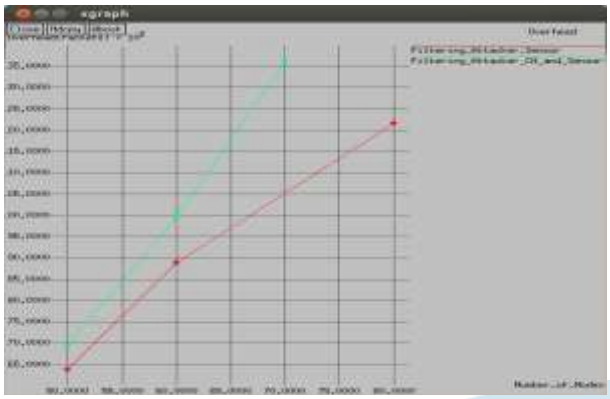


Fig10: Overhead

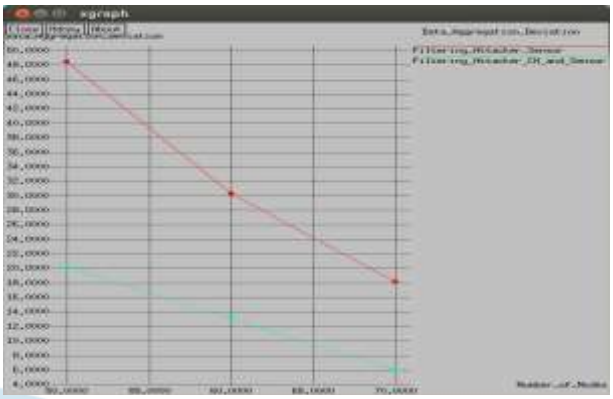


Fig11: Data aggregation deviation

When the number of nodes increased overhead is decreased. The filtering attacker cluster head provides increased overhead compared to filtering attacker sensor.

When the number of nodes increased attacker data aggregation deviation is decreased. The filtering attacker cluster head incurs decreased data aggregation deviation compared to filtering attacker sensor.

V. CONCLUSION

Synopsis diffusion algorithms plays an important role in computing secured aggregation at BS using count and Sum algorithms. The security issues of in-network aggregation algorithms are useful for computing aggregates such as predicate Count and Sum. Error can be injected by a single node and also cluster head, a small amount of error may cause large amount of error at BS. Thereby changing the Base Station's estimate of the aggregate. An Attack resilient computation algorithm helps in getting correct aggregates at each CH and also at BS. Thereby guarantee the successful computation of the aggregate even in the presence of the attack.

Acknowledgment

I like to express sincere gratitude to Dr. Manjunatha P.,Professor and Head of Department, JNNCE of Engineering, Shimoga and thankful to Mr. Sharath M, Assistant Professor, Dept. of ECE, JNNCE, Shimoga for his continuous support, comments and guidance throughout the implementation of proposed method.

REFERENCES

[1] Sankardas Roy , Sanjeev Setia, and Sushil Jajodia. Attack-resilient hierarchical data aggregation in sensor networks. In Proc. of ACM Workshop on Security of Sensor and Adhoc Networks (SASN), 2006 .

[2] Teresa Ko, Josh Hyman, Eric Graham, Mark Hansen, Stefano Soatto, and Deborah Estrin. Embedded imagers: Detecting, localizing, and recognizing objects and events in natural habitats. Proceedings of the IEEE: Special issue on sensor network applications., 98(11) :1934–1946, 2010.

[3] Peter Corke, Tim Wark, Raja Jurdak, Wen Hu, Philip Valencia, and Darren Moore. Environmental wireless sensor networks. Proceedings of the IEEE: Special issue on sensor network applications., 98(11) :1903– 1917, 2010.

[4] S. Madden, M. J. Franklin, J.M. Hellerstein, and W. Hong. TAG: A tiny aggregation service for ad hoc sensor networks. In Proc. of 5th USENIX Symposium on Operating Systems Design and Implementation, 2002.

[5] J. Considine, F. Li, G. Kollios, and J. Byers. Approximate aggregation techniques for sensor databases. In Proc. of IEEE Int'l Conf. on Data Engineering (ICDE), 2004.

[6] S. Nath, S. Seshan, and Z. Anderson. Synopsis diffusion for robust aggregation in sensor networks. In Proc. of the 2nd international conference on Embedded networked sensor systems (SenSys), 2004.

[7] M. Garofalakis, P. Maniatis and J. M. Hellerstein. Proof sketches: Verifiable in-network aggregation. In Proc. of the 23rd Int'l Conference on Data Engineering (ICDE), 2007.

[8] Haifeng Yu. Secure and highly-available aggregation queries in large scale sensor networks via set sampling. In Proc. of the Int'l Conference on Information Processing in Sensor Networks, 2009.