International Journal of Latest Trends in Engineering and Technology
Vol.(7)Issue(3), pp. 589-597
DOI: http://dx.doi.org/10.21172/1.73.5779
e-ISSN:2278-621X

Data stratified collation Wireless Sensor Networks

R Shiva Shankar[1], K V S Murthy[2], V MNSSVKR Gupta[3] and D Ravibabu[4]

ABSTRACT
Wireless sensor networks are typically unnecessary, becausestrength of monitoring is needed. An aggregator node aggregates the data from multiple sensors that in turn send the only aggregated values to base station. Currently, because of sensor nodes computing power and resources of energy are limited, very easy algorithms are used for aggregating the data for example averaging. But the above used averaging algorithm was identified to be very risk to faults, and more significantly to malicious attacks.As the attackers get access completely to the information that is stored in compromised nodes, cryptographic methods were not a remedy. WSN. Because of this enhanced and much more complicated algorithms are in need for aggregating the data in future WSN. The below presented algorithm allows the base station to figure out the predicate count and Sum securely still the attack present at that time. The attack-resilient computation algorithm calculates the true aggregate by sorting out the inputs of compromised nodes in the hierarchy of the aggregation. Systematic theoretical analysis and wide-ranging simulation study demonstrate that the algorithm do better than other existing methods.

INTRODUCTION

WSNs are significantly used in many of the real-world applications in the last decade, like wild habitat monitoring, volcano and fire monitoring, urban sensing, and military surveillance. In many of the situations, the sensor nodes shape a multi-hop network as the base station (BS) operates as the Central point of the control. Usually, sensor nodes computing power and resources of energy are limited. The BS wishes to gather the sensed information from the network.

The familiarpath is to permitevery sensor node to send its reading to the BS, probablythroughthe other intermediate nodes. Lastly, the BS progresses the data that is received from the individual nodes. On the other hand, this procedure is prohibitively costly in terms of communication overhead. Data aggregation in a networkis able todecrease the amount of communication and therefore the energy consumed, particularly in big WSNs. The most importantplan is to merge partial outcomes at intermediate nodes throughout message routing. One of the approaches is to build a spanning tree rooted at BS, and nextcarry out in-network aggregation along the tree. The significant aggregates taken by the research community comprise Count, and the Sum. It is simple to simplify these

[1] *Dept. of CSE, SRKR Engineering College, Bhimavaram, West Godavari, AP, India*
[2] *Dept. of CSE, SRKR Engineering College, Bhimavaram, West Godavari, AP, India*
[3] *Dept. of CSE, SRKR Engineering College, Bhimavaram, West Godavari, AP, India*
[4] *Dept. of CSE, SRKR Engineering College, Bhimavaram, West Godavari, AP, India*

aggregates to predicate Count (e.g, number of sensors whose reading is greater than 10 units) and Sum. Additionally, Average will also be calculated from Count and Sum.

We can also simplyenlarge a Sum algorithm to calculate Standard Deviation and Statistical Moment of every order. Though, communication losses resultant from node and transmission failures, those areusual in WSNs, can badly affect tree-based aggregation methods. In order todeal with this problem, we can take use of multi-path routing procedures for sending the sub-aggregates. For duplicate insensitive aggregates for instance Min and Max, this procedureoffers a fault-tolerant resolution. Unluckily, for duplicate sensitive aggregates, for example Count and Sum, multi-path routing directs to double-counting of sensor readings.

In recent times, many of the researchers have offeredintelligent algorithms to resolve this double-counting problem. A robust and scalable aggregation framework called synopsis diffusion has been provided for computing duplicate-sensitive aggregatesfor example Count and Sum. This procedure makes use of a ring topology in which a node can have multiple parents in the hierarchy of aggregation. In addition, every sensed assessment or sub-aggregate is signified by a duplicate-insensitive bitmap named as synopsis. The opportunity of node compromise brings in many of the challenges since most of the present in-network aggregation algorithms have no provisions for security.

A compromised node mayeffort to prevent the aggregation procedure by introducingmany attacks, like eavesdropping, jamming, message dropping, message fabrication, and many others. This workmainly focuses on a subclass of those attacks wherein the oppositiontargets to cause the BS to get an aggregate which is incorrect. By communicating a false sub-aggregate to the parent node, a compromised node mightsupply a huge amount of fault to aggregate. Such as, while processing Sum computation algorithm a compromised node X might inject an arbitrary amount of fault in the last estimate of Sum by misrepresent X’s personal sub-aggregate. We represent to this attack as misrepresent sub-aggregate attack.

In this work, provide an algorithm in order to compute aggregates more securely, like Count and Sum despite the misrepresent sub aggregate attack. Particularly, the presented algorithm which is called as the attack-resilient computation algorithm contains two phases. The majordesign is as follows: (i) during the firststage, the BS derives a preliminary estimate of the aggregate depending on the nominal authentication information received from nodes. (ii) In the last stage, the BS requests extra authentication information from barely a subset of the nodes during this subset is determined by the estimate of the first stage. In conclusion, the BS is capable to filter out the wrong contributions of the compromised nodes from aggregate.

The key observation we develop here is toreduce the communication overhead is that to validate the accuracy of the last synopsis (representing the aggregate of the entire network) the BS don’t need to obtain authentication messages from all the nodes. We tested the performance of the proposed algorithm bymutually thorough theoretical analysis and also extensive simulation. The per-node overall communication overhead in the proposed algorithm is max (O (mlogA); O (mt)) where in m is O (1 e2 log 1 d), A is the aggregate value, and t compromised nodes are present in the network. Note that m items are calculated in parallel fashion to result in an (e;d)-approximate aggregate as described.

When Count is calculated, A = N in which N is the totality nodespresent in the network; while Sum is calculated, O (logA) = O (logN) + O (log v) in which v is maximum value of an individual node. The already available attack-resilient algorithm [10] acquires communication overhead in the worst case as O (N), which is much morelarger than the proposed in this given t <<N, and the unit of sensed values are like that log(v) << N. Moreover, the proposed algorithm requiresO(1) latency during the further existing algorithm takes O(logN) latency while both the algorithms (i.e. proposed and) basicallydeserve the similar communication overhead to guarantee the same approximation error guarantee..

Even though our earlier work takes the similar aggregation framework (i.e. synopsis diffusion) and same attack situation (i.e. falsified sub-aggregate attack) the goal (in addition to the result) is very unlike. Our earlier work proposes only a verification algorithm, which would have been failed to calculate the aggregate in the occurrence of an attack during our proposed work will be unbeaten in doing so. Additionally, we earlierproposedother attack-resilient aggregation algorithm for the synopsis diffusion framework;however the algorithm proposed in thiswork is much more efficient ad capable.

We stressed that the design principle of our proposedwork is much moreunlike from that in, e.g., by using a basic round to get an estimate of the aggregate which assistsin decreasing the communication overhead of the entire process is out. Furthermore, the theoretical analysis methodproposed in the current work is also dissimilar from that in. It is to be in notice that as our algorithm was designed by having WSNs in mind, it is simple to extend our solution for more secure aggregation query dealing out in a large-scale distributed system like a distributed database system above the Internet.

LITERATURE SURVEY

Approximate Aggregation Techniques for Sensor Databases:

In the up-and-coming area of sensor-based systems, a considerable challenge is to build up scalable, fault-tolerant procedures to extract valuable information from the data the sensors assemble. One of the approaches to this data management trouble is making use of sensor database systems, demonstrated by TinyDB and Cougar that ispermittingconsumers to carry out aggregation queries for instance MIN, COUNT and AVG on a network of sensors. Because of the limitations in power and range, methods which are centralized are usually not practical;consequentlythe majority systems use in-network aggregation to decrease network traffic. On the other hand, aggregation strategies are becoming bandwidth-intensive duringcombining with the fault-tolerant, multi-path routing proceduresfrequently used in this kind of environments. Such as, duplicate-sensitive aggregates like SUM cannot be calculatedaccuratelyby utilizing substantially fewerbandwidths than explicit enumeration. Thefollowing are the contributions: 1) simplify well recognized duplicate-insensitive sketches for approximating COUNT to handle SUM,2) presented and examined techniques for using sketches to generate accurate outcomes with low communication and computation overhead, and 3) proposed an extensive experimental validation of the methods.

Computing Aggregates for Monitoring Wireless Sensor Networks

Wireless sensor networks engaged byhuge numbers of small, low-power, wireless devices. In this work, in briefexplainedarchitecture for sensor network monitoring system, then focus on one aspect of this architecture: incessantly computing aggregates (sum, average, count) of network properties (loss rates, energylevels etc., and packet counts). The contributions are two-fold. First, we proposed a novel tree building algorithm that allows energy-efficient calculation of various classes of aggregates. Finally, we demonstrated through actual accomplishment and experiments that wireless communication artifacts in yetcomparatively benign environments can considerably impact the calculation of these aggregate properties. In several occasions, without cautiousconcentration to detail, the relative fault in the computed aggregates can be to the extent 50%. BUT, by suspiciouslyremoving links with heavy packet loss and asymmetry, we can increase accuracy by an order of magnitude.

Diffusion for Robust Aggregation in Sensor Networks

Earliermethods for calculating duplicate-sensitive aggregates in sensor networks (e.g., in TAG) enclosed a tree topology, so as topreserve energy and to evade double-counting sensor readings. But, a tree topology is not robust in opposition to node and communication crashes, those are common in sensor networks. In this work, we propose synopsis diffusion, a common framework in order to considerablyextra accurate and reliable responses by grouping energy-efficient multi-path routing designs with methods that evade double-counting. Synopsis diffusion removes double-counting during the use of order- and duplicate-insensitive (ODI) synopses that efficientlyreview intermediate outcomes during in-network aggregation. This workoffers amazinglyeasy test that constructs it simple to verify the rightness of an ODI synopsis. This work demonstrates that the properties of ODI synopses and synopsis diffusion generate implicit acknowledgments of packet delivery. We show that this property can, in turn, enable the system to adjust message routing to dynamic message loss circumstances, still in the occurrence of asymmetric links. At last, this workexemplify, using extensive simulations, the significant robustness, ac- curacy, and energy-efficiency developments of synopsis dif- fusion over earliertechniques.

Proof Sketches: Verifiable In-Network Aggregation

The work in currenttimes on distributed and in-network aggregation presuppose a benign population of contestants.Unexpectedly, modern distributed systems are overwhelmed by malicious contestants. During this work, provide a starting step headed forconfirmable yet well-organized distributed, in-network aggregation in adversarial settings. This work illustrates a common framework and risk model for the trouble and then providesproof sketches, a compact confirmation mechanism that groups cryptographic signatures and Flajolet-Martin sketches to assure acceptable aggregation fault bounds with more probability. This wokgain proof sketches for calculate aggregates and enlarge them for the random sampling, this can be used to offerdemonstrable approximations for a broad variety of data analysis queries, e.g., quintiles and heavy hitters. Lastly, this workassess the realistic use of proof sketches, and examine that adversaries can regularly be decreased to greatly smaller violations in practice than previous worst-case bounds propose.

EXISTING SYSTEM

With the purpose of improving the performance of IF algorithms next to the aforementioned attack scenario, this work provide a robust early estimation of the honesty of sensor nodes to be used in the starting iteration of IF algorithm. Many of the conventional statistical estimation techniques for variances engage use of the sample mean. Consequently, suggesting a robust variance estimation technique in the scenario of skewed sample mean is important part of this methodology. Therefore, during this work,recommended a novel attacker detection methos in order to supplementary diminish the impact of the compromised nodes. This work will describe the proposed collusion detection method and then talk about the proposed compromised nodes revocation method.

PROPOSED SYSTEM

Iterative Filtering (IF) algorithms are smartalternative for WSNs since they solve two problemsmutually- data aggregation and data trustworthiness estimation by using a repeatablepractice. Such trustworthiness estimate of every sensor is stand on the distance of the readings of such a sensor from the estimate of the correct outcomes, gained in the earlier round of repetition by some structure of aggregation of the readings of all the sensors. This aggregation is generally a weighted average; sensors whose readings considerablyvary from such estimate are handoversfewer trustworthiness and as a result in the aggregation procedure in the current round of repetition their readings are specified a lower weight. The main aim of this work is to allow BS to get the ‘true’ estimate of aggregate (which BS would calculate if there were no compromised nodes) still in the occurrence of attack. More frequently, goal (a) is to identify if ˆB, the synopsis gained at BS is the similar as the ‘true’ last synopsis B, and goal (b) is to calculate B from ˆB and further received information. With no loss of simplification, this workdescribesthe algorithms in the situation of Sum aggregate. As Count is a special scenario of Sum, in which each and every node tells a unit outcome, these algorithms are readily appropriate to Count aggregate too.

IMPLEMENTATION

- **Service Provider**

During this module, the Service Provider computes the less distanced path to target, the shortest-path routing above the Internet BGP-based router. The Service provider gets the file that is required and imports their corresponds data files to the Particular End User (A, B, C, D) along with their DIP (Destination IP) of End User.

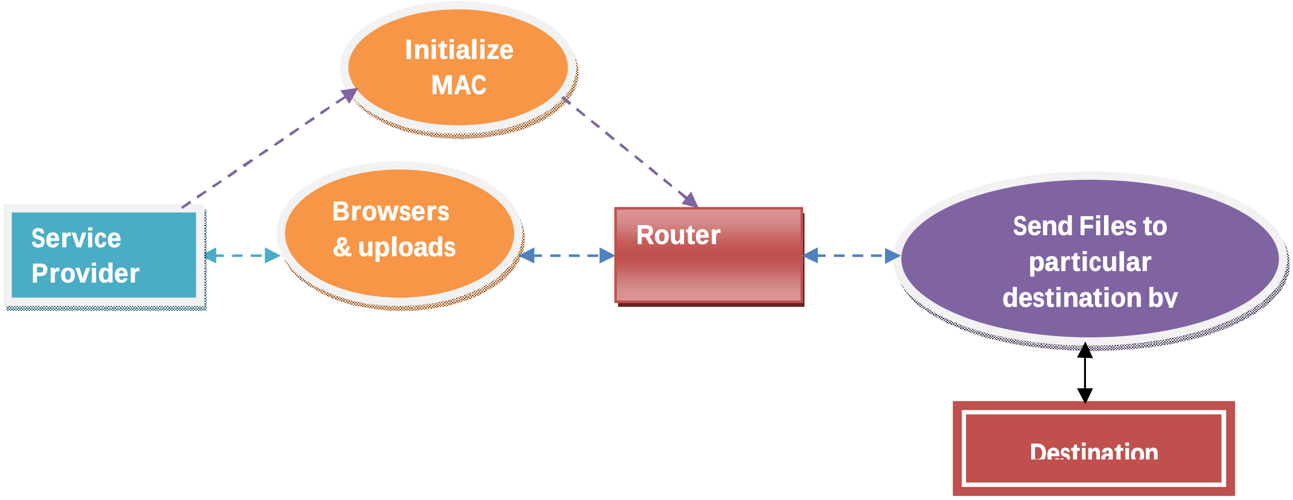


Figure 1: Service Provider

- **Router**

The main responsibility of the Router is to forward the file to the destination that was specified, the routing technique is the group of all the smallest physical paths make simpler the implementation of the system, and discovering a shortest path to the specified destination using routing, one individual can carry out routing through shortest paths, the router as well responsible for handing over the cost and as well can see the price of the nodes with their tags From the node (from), To the node (to) and the cost. Although the router is routing the path, if some of attackers identified then it will be localized using kalman filtering algorithm. The attackers may be fake injected data or IP spoofing attackers.

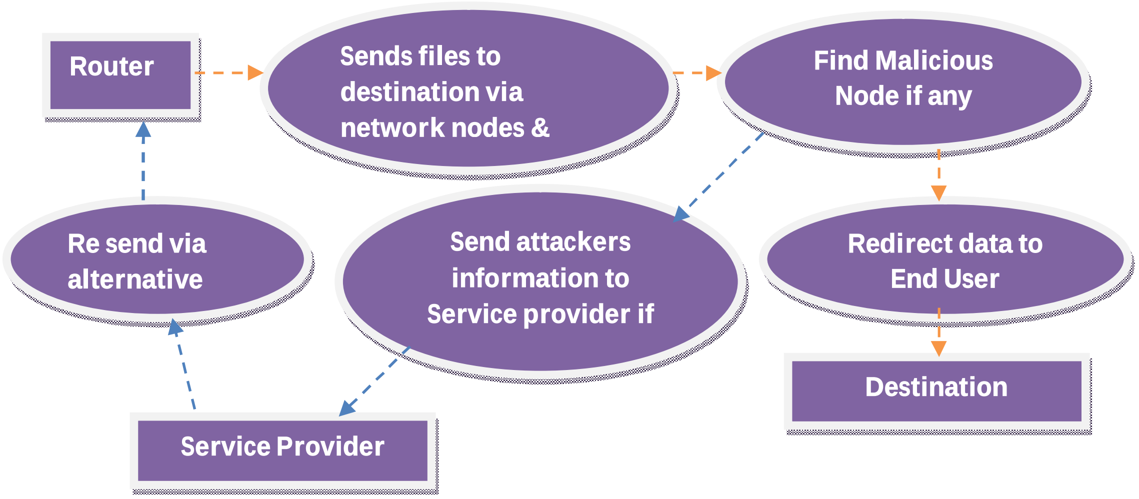

Router
Sends files to destination via network nodes &
Find Malicious Node if any
Re send via alternative
Send attackers information to Service provider if
Redirect data to End User
Destination
Service Provider

Figure 2: Router

- **Secure Aggregation Techniques**

A number of secure aggregation algorithms were been anticipated presuming that the BS is the single aggregator node in the network. The proposed works cannottake into consideration in network aggregation. In recent times only, the research community has been giving concentration to the security problems of hierarchical aggregation.The starting attack-resilient hierarchical data aggregation protocol wasbuilt in this system. But, this technique is more secure at time of only one malicious node is there in the network. A tree-based verification algorithm was built in this system by using that the BS can identify if the last aggregate, Count or Sum, is falsified.

- **Attacker**

Attacker is one who is instilling malicious data to the resultant node or IP spoofing to the respective node. The attacker will inject falsedata to the specific node. After attacking the nodes, data will interchanged into a router.

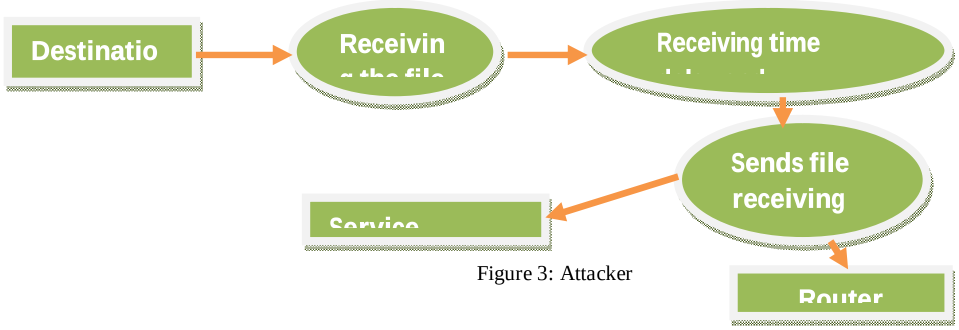
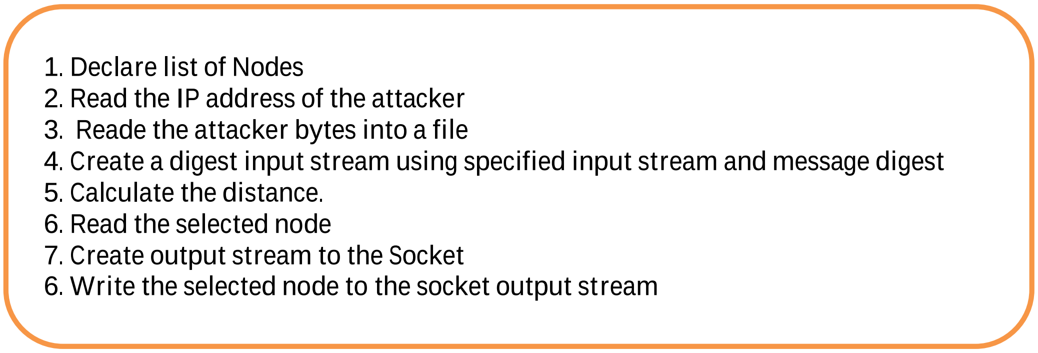


Figure 3: Attacker

- **End User(Destination)**

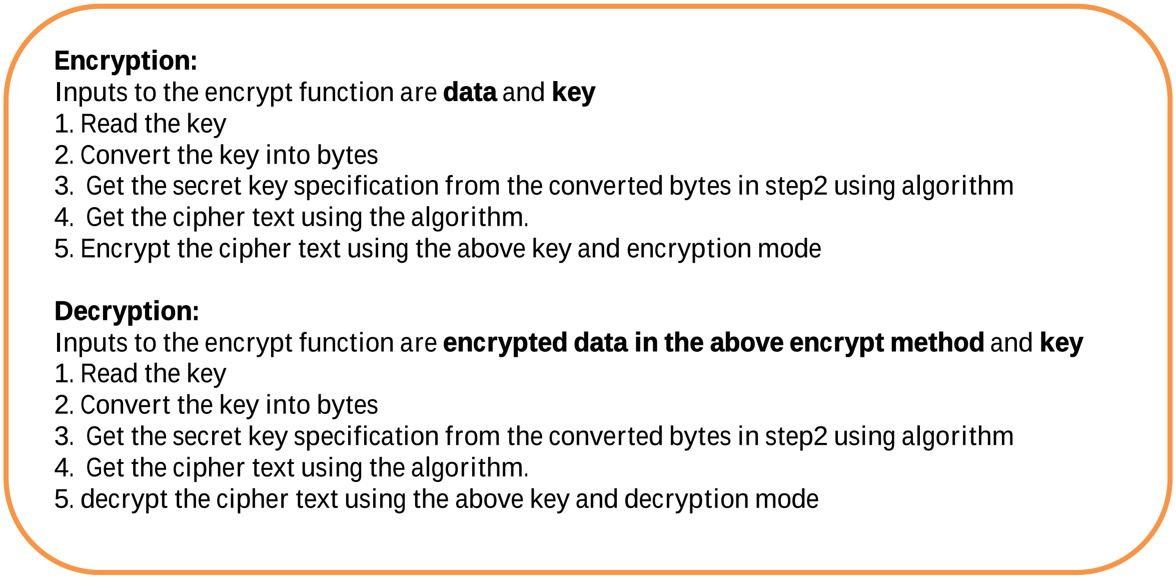
During this module, the End user (Node A, Node B, Node C, Node D) is answerable to get the file from the Service Provider in the shortest-path routing involving the source and destination nodes, the system contain one-to-many relationship. In which end User derives file from a individual source to destination (Node A, Node B, Node C, Node D).

ALGORITHMS



```
1. Declare list of Nodes
2. Read the IP address of the attacker
3.  Reade the attacker bytes into a file
4. Create a digest input stream using specified input stream and message digest
5. Calculate the distance.
6. Read the selected node
7. Create output stream to the Socket
6. Write the selected node to the socket output stream
```

Figure 4. Algorithm for Hierachical



Encryption:

Inputs to the encrypt function are **data** and **key**

```
1. Read the key
2. Convert the key into bytes
3.  Get the secret key specification from the converted bytes in step2 using algorithm
4.  Get the cipher text using the algorithm.
5. Encrypt the cipher text using the above key and encryption mode
```

Decryption:

Inputs to the encrypt function are **encrypted data in the above encrypt method** and **key**

```
1. Read the key
2. Convert the key into bytes
3.  Get the secret key specification from the converted bytes in step2 using algorithm
4.  Get the cipher text using the algorithm.
5. decrypt the cipher text using the above key and decryption mode
```

Figure 5. Algorithm for **AES**

The synopsis diffusion framework solitary cannot comprise any requirements for security. In order to prevent nodes which are not authorized from interfering in communications between honest nodes, this work can expand the aggregation framework by using some of standard authorization and encryption protocols. Consequently, we observed that there is no need to take the attacks into consideration approaching from unauthorized nodes in the rest of this work. But, cryptographic techniques do not stop attacks initiated by compromised nodes since the opponent can get the cryptographic keys through the compromised nodes. Compromisednodes may effort to prevent the aggregate computing procedure in many of the ways Many of the researchers [25] previously projected privacy-preserving aggregation algorithms, and we are not taking this difficulty during the rest of work. Following are some we described other possible issues and detect the scope of this work.

1. Falsifying the local value: A compromised node *C* can misrepresent its personal sensor reading with the target of pressurizing the aggregate result.

There are three cases.

Case (i): Suppose the local value of a truthful node preserve to be *any* value; subsequently a compromised node might act as if to sense *any* value. During this situation, there is no method to identify the fallacious local value attack (as also confirmed in [17]). We left Case (i) as out of possibility of this work.

Case (ii): If the local value of truthful node is bounded, and the compromised node falsify

The local value inside the bound, there was no answer for identifying an attack such as in Case (i). We only verified that in Case (ii), the impact of these kinds of attacks is restricted as described in the Appendix.

Case (iii): The local value of a truthful node is bounded, and compromised nodes falsify the local value exterior the bound.

2. Falsifying the sub-aggregate: A compromised node *C* can misrepresent the sub-aggregate that *C* is believed to calculate based on the messages derived from *C*'s children nodes. It is demanding to protect in opposition to this attack, and dealing with this dispute is the major focus of this work. We presume that if a node was compromised, all the information that it holds will be compromised. We conservatively regard that all malicious nodes can join together or will be below the control of a sole attacker. We make use of Byzantine fault model, in which the opponent can inject several messages via the compromised nodes. Compromised nodes might perform in arbitrarily malicious methods, that means that the *sub-aggregate*of a compromised node will be arbitrarily produced. But, we presume that the attacker cannot initiate *DoS*attacks, e.g., the multi-hop flooding attacks [11] with the target of making the entire system *unavailable*.

RESULT ANALYSIS

The performance of the protocol mainly based on the loose cut of the estimate, rˆ which is gained in first phase. In addition, the highest deviation in estimate rˆ from correct r (which is gained in phase two) depends on how several compromised nodes take part in the false MAC injection attack while execution of phase one.

Worst-case communication overhead:

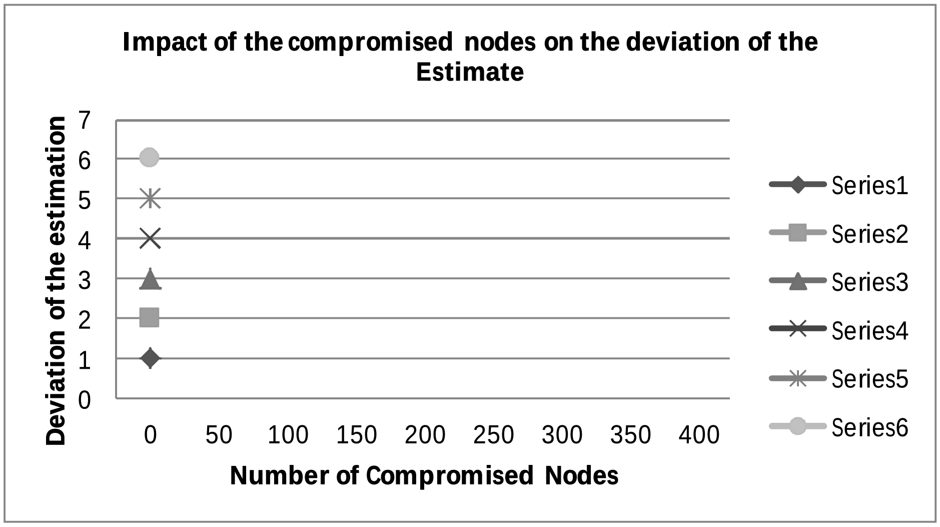


Fig.6: Impact of the compromised nodes on the deviation of the Estimate

We cannot decrease this overhead for the reason that (in the worst case) the compromised nodes can constantly inject wrong MACs for each and every of the η bits. Alternatively, in the second phase a node (in the worst case, i.e., near BS) requires to send O(t) MACs based on our investigation, in which t is the total of compromised nodes. Fig. 6 represents the total of distinctive MACs forwarded entire the entire network while second phase as a function of t. The 99% assurance intervals are contained by ±20% of the reported results. We examined that the total number of MACs adds in linear fashion with t, which proves our analysis.

Average communication overhead:

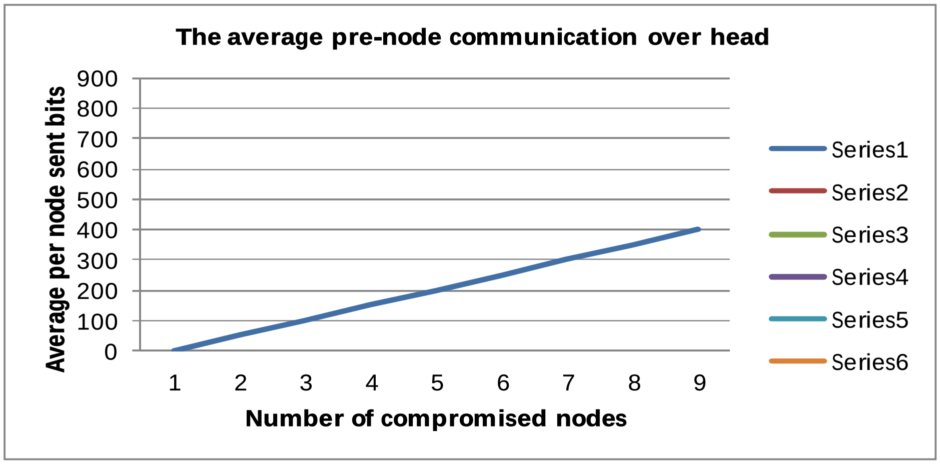


Fig.7: The average pre-node communication overhead

To account the total (over first and second phases) average (over all nodes) communication overhead, we included in Fig. 7 that describes how it (in bits) varies with t. This diagram also independently plots the per-node overhead above first phase. We examined that first phase took the most important share of the per-node overhead. We presume that every MAC is of size 32 bits and it is forwarded together with the generator node Id of length 16 bits. In addition, in individual experiments, we verified that the per-node communication overhead cannot increase by the network size (varied across 900, 1600, 2500, and 3600), that describes that our algorithm is scalable one.

CONCLUSION

In this talked about the security problems of in-network aggregation algorithms to calculate aggregates like predicate Count and Sum. Specifically, the work showed the fallacious sub-aggregate attack commenced by a less compromised nodes be able to inject arbitrary amount of fault in the base station’s estimate of an aggregate. This workprovided an attack-resilient computation algorithm that would assure the thriving computation of the aggregate still in the attendance of an attack.

REFERENCES

[1] Mingyan Liu, Neal Patwari, and Andreas Terzis.Scanning the issue. Proceedings of the IEEE: Special issue on sensor network applications., 98(11):1804–1807, 2010.

[2] SumanNath, Haifeng Yu, and Haowen Chan. Secure outsourced aggregation via one-way chains. In Proc. of the 35th SIGMOD international conference on Management of data, pages 31–44, 2009.

[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. Zhao, R. Govindan, and D. Estrin. Computing aggregates for monitoring sensor networks. Sensor Network Protocols and Applications, 2003.

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

[7] S. Nath, P. B. Gibbons, 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.

[8] M. Garofalakis and P. Maniatis. Verifiable in-network aggregation . In Proc 23rd Int’l Conference on Data Engineering (ICDE), 2007.

[9] Y. Yang, X. Wang, S. Zhu, and G. Cao. SDAP: A secure hop-by-hop data aggregation protocol for sensor networks. In Proc. of ACM MOBIHOC, 2006.

[10] Haifeng Yu. Secure and highly-available aggregation queries in largescale sensor networks via set sampling. In Proc. Int’l Conference on Information Processing in Sensor Networks, 2009.

[11] M. B. Greenwald and S. Khanna. Power-conservative computation of order-statistics over sensor networks.In Proc. of the 23th SIGMOD Principles of Database Systems (PODS), 2004.

[12] L. Hu and D. Evans. Secure aggregation for wireless networks. In Proc. of Workshop on Security and Assurance in Ad hoc Networks., 2003.