World Academy of Science, Engineering and Technology
International Journal of Electronics and Communication Engineering
Vol:14, No:12, 2020

Security in Resource Constraints Network Light Weight Encryption for Z-MAC

Mona Almansoori, Ahmed Mustafa, Ahmad Elshamy

***Abstract*—**Wireless sensor network was formed by a combination of nodes, systematically it transmitting the data to their base stations, this transmission data can be easily compromised if the limited processing power and the data consistency from these nodes are kept in mind; there is always a discussion to address the secure data transfer or transmission in actual time. This will present a mechanism to securely transmit the data over a chain of sensor nodes without compromising the throughput of the network by utilizing available battery resources available in the sensor node. Our methodology takes many different advantages of Z-MAC protocol for its efficiency, and it provides a unique key by sharing the mechanism using neighbor node MAC address. We present a light weighted data integrity layer which is embedded in the Z-MAC protocol to prove that our protocol performs well than Z-MAC when we introduce the different attack scenarios.

***Keywords*—**Hybrid MAC protocol, data integrity, lightweight encryption, Neighbor based key sharing, Sensor node data processing, Z-MAC.

I. Introduction

WIRELESS sensor network formed by joining different nodes to collect and process vast streams of data indented to form a decision on the state of the operations based on the functional requirements of the network. To process and collect data always have tradeoffs between data integrity and processing; both are equally important. Emphases on the data processing can lead to questions on the authenticity of the data processed by the network nodes, and giving higher precedence to integrity can lead to data losses or latency issues eventually leading to the observations being invalid for the intended data set collected by the nodes distributed across the network. Many attempts happened to reduce the issue of the data integrity and processing still tradeoffs, while some emphasize on the data integrity, the others take a hybrid approach leaving the decision for the network implementers to decide on the approach based on their resources and requirements. We take an approach to this issue, taking in all three considerations of using a hybrid approach to network access that can handle both relatively small and vast sets of data packets providing a balance between latency, data processing, and integrity. This paper presents a secure MAC protocol intended for sensor networks utilizing hybrid Z-MAC operations for maintaining consistent latency in network with lightweight encryption of elliptic curve and a unique HOP based distributed key sharing mechanism to maintain data integrity. We compared the proposed mechanism with Z-MAC operated network by introducing different attack vectors. We concluded that our solution could achieve almost 40% more efficient output in terms of utilizing the data at the center, properly encrypting the chains of packets from source to destination and keeping the latency low for both high and low contention networks without being compromised, achieving our goal of maintaining a real-time network to process and produce accurate near real-time results of the sensor data for the intended function of the network.

II. Security

As we have discussed previously that sensor nodes in a wireless networks operate in an open network and require data integrity using secure packet transfer using encryption and secure key sharing mechanism, taking advantage of light weight encryption and key sharing mechanism described by IHOP [3] we generate a hierarchical key sharing mechanism that encrypts every packet using eliptic curve. Next section will discuss what hierarchical keys are before we move to the proposed secure Z-MAC mechanism.

We built our secure Z-Mac [1] scheme on hierarchical identity based cryptography, because of the uniqueness of MAC address IHOP discussed above we generate shared keys between neighbors nodes when the neighbor discovery is started by the Z-Mac, below are the steps followed by our scheme when Z-MAC neighbor discovery starts

1. Discover every 1 hop neighbor as designed by the network base station
2. Sends a handshake hello message to its 1 hop neighbor
3. Calculate weighted average of the response of the network to maintain trust factor between nodes
4. Share the Mac address as the PKG(Private key generator) with neighbor node
5. Use IHOP to calculate SKG(Secret key generator)
6. Use the PKG and SKG defined by eliptic curve[2] to send and receive packets

As each node has unique Mac address, our private key cannot be compromised as any new node added will not share the same key with the participating node, even if user is able to access the Mac address of the node user cannot obtain the key as ensured by IHOP [3].

Next we discuss how these messages are transmitted using the proposed scheme.

III. Z -MAC

Z-MAC has a setup stage in which it runs the accompanying tasks in arrangement: neighbor discovery, opening task, nearby node discovery, and local time

Mona Almansoori is with the Suez Canal University, Egypt (e-mail: mnmalmansoori@gmail.com).

Open Science Index, Electronics and Communication Engineering Vol:14, No:12, 2020 publications.waset.org/10011634.pdf

synchronization. These tasks run just once amid the setup stage and do not keep running until a critical change in the system topology (for example, physical migration of sensors) happens. The thought is that the underlying forthright expenses for running these tasks are amortized by enhanced throughput and vitality effectiveness amid information transmission. In this area, we initially depict how we execute these setup stage tasks and after that talk about how they are coordinated with transmission control in Z-MAC.

A. Neighbor Discovery and Slot Assignment

As a node begins up, it first runs a straightforward discovery protocol where it intermittently communicates a ping to its one-hop neighbors to accumulate its one-jump neighbor list. A ping message contains the current rundown of its one-hop neighbors. In our usage, every node sends one ping message at an arbitrary time in each second for 30 s. Through this procedure, every node accumulates the data got from the pings from its one-hop neighbors, which basically establishes its two-jump neighbor data. The two-jump neighbor list is utilized as contribution to a schedule slot task calculation. The present execution of Z-MAC utilizes DRAND [4], a conveyed usage of RAND [5], to allot schedule slots to each node in the system. DRAND guarantees a communicate plan where no two nodes inside a two-jump correspondence neighborhood are allotted to a similar slot. This task ensures that no transmission by a node to any of its one-hop neighbors meddles with any transmission by its two-hop neighbors. Note that a communicate timetable can deal with any directing changes among its one-hop neighbors. The execution of DRAND is versatile in light of the fact that it does not rely upon the system measure, however on the nearby neighborhood size of every node. The convention creates an extremely effective time plan where the slot number relegated to a node does not surpass the measure of its nearby two-hop neighborhood by and large, considerably less than that. The running time and message multifaceted nature of DRAND is additionally limited. In this manner, its vitality cost is directly corresponding to the span of the nearby neighborhood. At the point when just few new nodes are joined late, DRAND can likewise perform limited availability task without adjusting the schedule slots officially appointed to the current nodes.

If we consider a new attacker node here it can introduce itself as a neighbor and join the network to transmit invalid information or perform different kinds of attacks like DDOS, MAN in the middle, eavesdropping. An attacker can use node replication attack to replicate a node as valid node as no integrity checks are done to verify or declare a node as trusted sender or receiver. As discussed in Section V when neighbor discovery starts, we exchange the MAC address of the node and utilize the IHOP explained in Section IV to generate primary key and shared keys between nodes, it then shares the key between n+1 nodes making creating a hierarchal key where each node shares its primary key with higher or lower nodes in the setup, every time a neighbor is discovered it is verified against the base station configuration as shown in Fig. 1. In case of attacks like DDOS (dynamic denial of service), nodes will not accept any packet that is not encrypted or sent from the node they share key with, in case of Man in the middle attack attacker node will not be able to decrypt and modify any kind of packet as receiving node will always verify the MAC of the sender node as explained in IHOP.

Introducing the IHOP based key generation in the discovery function of Z-Mac will secure the integrity of the data sent by the node and elliptic curve will encrypt each packet to send it securely over the transmission link to the receiving node.

We will use the time synchronization of ZMAC to create trust between nodes to very it as valid node during initial network setup only known nodes will have access to information exchange, time synchronization concept is discussed in next section.

B. Local Framing

When a node picks a schedule vacancy, every node needs to settle on the period in which it can utilize the availability for transmission. This period is known as the time allotment of the node. The customary way of thinking is that all nodes must keep a similar time period while all nodes synchronize to have their vacancy 0 in the meantime. Be that as it may, this requires spreading the greatest slot number (MSN) to the whole system and is likewise not versatile to nearby availability changes. At the point when new nodes are added to the system, DRAND can run nearby space task while keeping up the current task. On the off chance that this task makes the MSN be changed, that change must be spread again to the whole system. This could cause staggering expense for adjusting to a little change in the system topology. (Note that organized topology changes by precarious radio channel conditions are taken care of by the inalienable activity of Z-MAC so it does not bring about new task, yet new node joining or node redeployment can cause slot changes.) We present another plan where every node keeps up its very own nearby time allotment that accommodates its neighborhood measure, yet maintains a strategic distance from any contention with its battling neighbors. Fig. 1 explains the concept of local framing, Once the neighbor discovery starts it will share the list of neighbor nodes with the chain of participating sensor nodes, once node list is shared, every neighbor will send a sync message with MAC address of node to each hop i.e. next neighbor node on the node setup, their shared MAC will be XOR using IHOP mechanism for neighbor based key generation and they will have a shared key to encrypt packets using eliptic curve [3] keeping the processing power low and maintaining the integrity of data shared between sensor nodes. Each node has its own schedule to transmit and receive the data packets, as seen in Fig. 1 each node has its own available slots assigned by transmitting schedule of Z-MAC. i.e. 1, 2 for node 1 we use the same slots to transmit the data after encrypting the data packets as explained above in local framing concept.

C. Receiving Schedule of Z-MAC

DRAND characterizes just the transmission timetable of nodes for each available slot in the node. In Z-MAC, a node

Open Science Index, Electronics and Communication Engineering Vol:14, No:12, 2020 publications.waset.org/10011634.pdf

can transmit in any available slot. Then again, Z-MAC does not characterize an accepting slot for nodes. Rather, it depends on the LPL method of B-MAC for accepting slots. Consequently, the vitality utilization of Z-MAC for inactive listening particularly under low obligation cycles is practically identical to that of B-MAC.

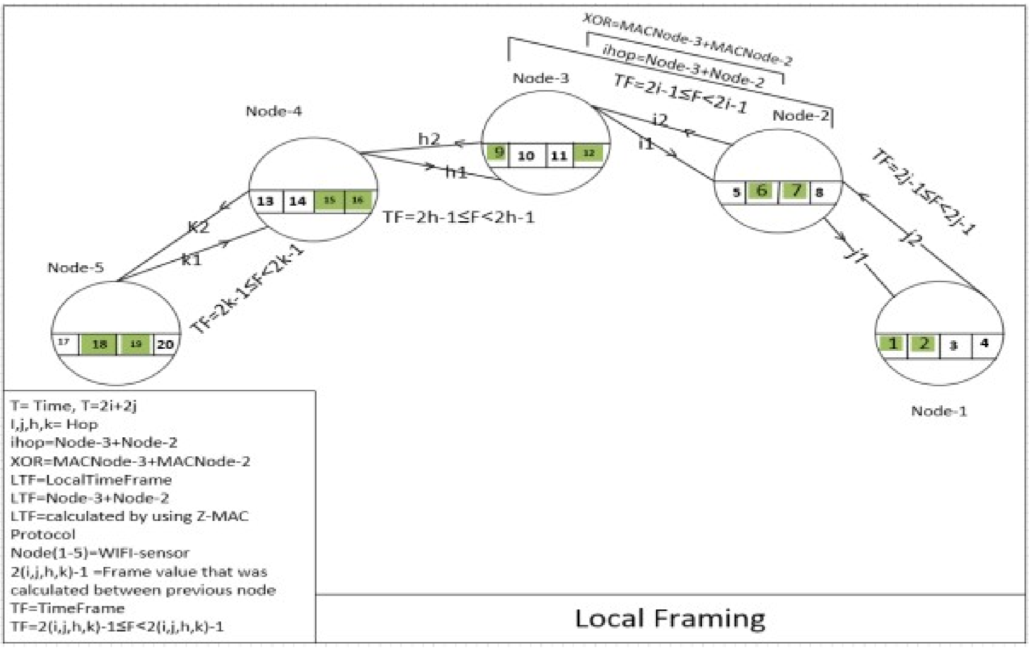

Fig. 1 Key exchange and generation mechanism

D. Local Time Synchronization

Z-MAC requires clock synchronization under high dispute to execute HCL. In Z-MAC, a node can be in one of two modes: low contention level (LCL) or high conflict level (HCL). However, we take note that synchronization is required just among neighboring senders and when they are under high dispute. This offers us a brilliant chance to streamline the overhead of clock synchronization since synchronization is required just locally among neighboring senders, and the recurrence of synchronization can be balanced by the transmission rates of senders so senders with higher information rates transmit more successive synchronization messages. In this plan, collectors inactively synchronize their tickers to the senders' timekeepers and do not need to send any synchronization messages.

IV. Experimental Setup

To assess the presentation of Z-MAC, we actualized Z-MAC in ns-2. We use ns-2 recreation to contrast the exhibition and existing conventions whose Tinos usage does not exist at the hour of fixing this work. In spite of the very fact that our exhibition assessment does not cover all the accessible sensor MAC conventions, we accept that the assessed conventions comprise an honest portrayal of existing conventions. Except if indicated else, we utilize the default settings of B-MAC as referred in [6]. Since Z-MAC is implemented on top of B-MAC, we use the identical packet format as B-MAC. We use three benchmark setups in our experiment: one-hop, two-hop benchmarks.

A. One-Hop Benchmark

This benchmark is reproduced from [6] — nodes are put equidistant from a collector around transmit as fast as conceivable with full transmission power. Prior to each run, we guaranteed that all nodes are in a one-jump separation to one another so that there are no concealed terminals. This benchmark is utilized to gauge the feasible throughput of various MAC conventions for various degrees of dispute inside a one-bounce neighborhood. All nodes are put in any event 2 feet separated and the separation to the recipient was roughly 2 m. The arrangement is set in an open meeting room with no hindrance. Ns-2 one-jump reproduction follows a similar arrangement. Fig. 2 depicts the scenario.

B. Two-Hop Benchmark

We make this benchmark to test the presentation of various conventions when concealed terminals are available. We arrange nodes into two groups where seven and eight sending nodes are situated in each bunch individually. The two bunches are set roughly 5 m separated in a house with drywall. A beneficiary hub (or steering hub) is put in the two groups. Nodes inside a similar group are set around 2 feet separated. In this condition, we cannot get a sharp limit of obstruction however we guarantee that all senders discover the recipient as a one-jump neighbor and all nodes are reachable by two-bounce interchanges. Then again, ns-2 reenactment of the two-hop benchmark can be characterized from of the two set of experiments with the goal that they become consistently two - bounce to one another. We expanded the system hub size by 5 nodes consistently to test the hub arrangement and time surrounding calculation, as we push ahead we presented assault vectors talked about above in the nodes effectively flooding the system and expanding the defer time among nodes and constraining the base station to drop the parcels. Fig. 3 depicts the scenario.

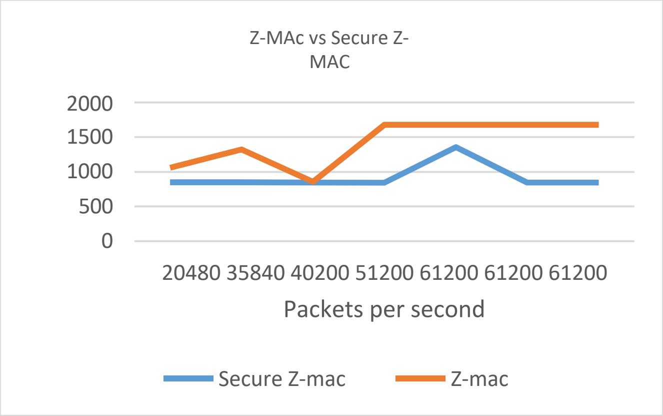

Fig. 2 One hop Benchmark

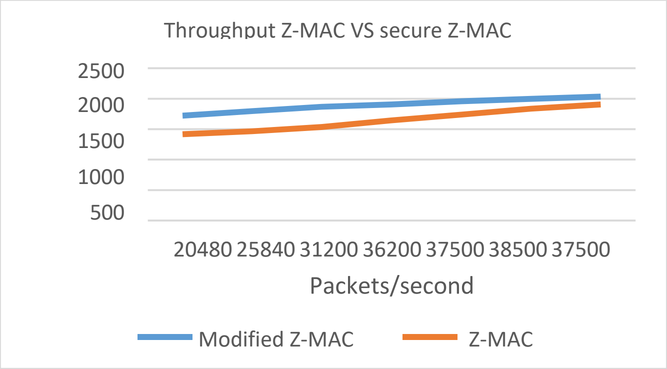

Fig. 3 Two Hop Benchmark

We can see the consistent in the diagram, depicting the proficient of the Z-MAC in Fig. 1, but in Fig. 2 the attack vectors presented, that its exhibit was affected. In a few seconds the through-put was expanded and then started to influence the exhibition in the remote system to get more information which was driven choices on base stations. We improved the exhibition by presenting the Ihop component examined in the above segment with key age through the MAC address strategy, where every hub shares the key through XOR of MAC on each progression, thus making chain square figure text used by eliptic curve. We found that the seat mark is not affected by altering the key sharing and age instrument with IHOP neighbor revelation.

V. Conclusion

The wireless sensor networks are more prone to several attacks like the entrance of black nodes that affect the reliability of network security. Therefore, to maintain these issues it must be done through published node authentication to ensure that the black node cannot enter the network completely. The main concern is to maintain network reliability and integrity. So only the authenticated unit will be able to send the data to another unit, if the unit is not authenticated then the data not forwarded. The evaluation of parameters like less key generation time, end to end delay [7], average delay, energy consumption, cipher text size, encryption time and increased throughput [8] of the network should be achieved to show improved results.

References

[1] Rhee, A. Warrier, M. Aia, J. Min and M. L. Sichitiu, "Z-MAC: A Hybrid MAC for Wireless Sensor Networks," in IEEE/ACM Transactions on Networking, vol. 16, no. 3, pp. 511-524, June 2008, doi: 10.1109/TNET.2007.900704.

[2] Lopez, Julio, and Ricardo Dahab. "An overview of elliptic curve cryptography." (2000).

[3] Zhu, Sencun, et al. "An interleaved hop-by-hop authentication scheme for filtering of injected false data in sensor networks." *IEEE Symposium on Security and Privacy, 2004. Proceedings. 2004*. IEEE, 2004.

[4] Rhee, Injong, et al. "DRAND: Distributed randomized TDMA scheduling for wireless ad hoc networks." *IEEE Transactions on Mobile Computing* 8.10 (2009): 1384-1396.

[5] Lu, Tao, Shunchao Jia, and Yinchang Li. "A modified RAND algorithm for multi-buyer Joint Replenishment Problem with resource constraints." *The 2nd International Conference on Information Science and Engineering*. IEEE, 2010.

[6] S. Ganeriwal and M. Srivastava, "Reputation-based framework for high integrity sensor networks", In Proceedings of the 2nd ACM Workshop on Security on Ad Hoc and Sensor Networks, Washington DC, USA, 2004

[7] N. M. A. Latiff, C. C. Tsimenidis and B. S. Sharif, "Energy-Aware Clustering for Wireless Sensor Networks using Particle Swarm Optimization," 2007 IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications, Athens, 2007, pp. 1-5, doi: 10.1109/PIMRC.2007.4394521.'

[8] N. Gura, A. Wander, H. Eberle, and S. Shantz, "Comparing elliptic curve cryptography and RSA on 8-bit CPUs", In Proceedings of Workshop on Cryptographic Hardware and Embedded Systems (CHES '04), August 2004.