Layered-MAC: An Energy-Protected and Efficient Protocol for Wireless Sensor Networks

Udoh, E. and Getov, Vladimir

This is an author's accepted manuscript of an article published in in Tang, D., Zhong, J. and Zhou, D. (eds.) EAI MobileWare 2021, Online, 22 - 24 Oct 2021, Springer. DOI:10.1007/978-3-030-98671-1_4 .

The final authenticated publication is available at Springer via:

https://doi.org/10.1007/978-3-030-98671-1_4

Layered-MAC: An Energy-Protected and Efficient Protocol for Wireless Sensor Networks[1]

Ekereuke Udoh[0000-0003-1655-8829] and Vladimir Getov[0000-0002-5597-7682]

Distributed and Intelligent Systems Research Group
University of Westminster, London, United Kingdom

ekereuke.udoh@qa.com
v.s.getov@westminster.ac.uk

Abstract. In wireless sensor networks, the radio of the wireless sensor node happens to be the highest source of energy consumption. Hence, there is a need to focus on the MAC layer, as it controls access to the radio. While there are several existing techniques to make sensors more energy-efficient, not many of them consider the security aspects of energy efficiency. By this we mean, protecting energy from external attacks. The existing protocols focus mainly on either duty-cycling (Sensor-MAC, Time-out MAC) or clustering (Gateway MAC), as a way of conserving energy. One of such attacks to energy is the denial-of-sleep (DoSL) attack which is a specific kind of denial-of-service attacks designed to drain the energy of battery-powered sensors in a Wireless Sensor Network. This paper explains the development of a new MAC-layer protocol called Layered-MAC aimed at not just energy efficiency but energy protection against DoSL attacks. The protocol is implemented on the OMNET++ and Castalia simulator. The results from the simulation are then compared with two representative existing duty-cycled protocols (Time-out MAC and Sensor-MAC) and significant improvements are present. One of the benefits of the developed protocol is that, not only does it attempt to save energy, but it protects energy from DoSL attacks. There are two main contributions from this research – the first is the additional layer of network metrics (RSSI and LQI) consideration, based on the premise that protection/security is not possible without some form of measurement of assets, and the cluster head rotation which adds an extra layer of energy protection while considering energy efficiency.

Keywords: MAC Layer, Denial-of-Sleep Attacks, Wireless Sensor Networks, Energy Efficiency, Energy Protection, OMNET++.

1 Introduction

This paper documents the development of a new MAC layer protocol which demonstrates an ability to tackle denial-of-sleep (DoSL) attacks better than the existing duty-cycled protocols. Battery-powered sensors usually have a network lifetime of 3.5 years [36]. However, a successful DoSL attack can reduce the lifespan of these sensors to 3

[1] This is a modified and extended version of previously published work [15].

days [6, 16-18]. Such significant loss of energy requires a deeper look into the problem, hence the need for a protocol that is energy-efficient and protects against these attacks. Our protocol is implemented on two different platforms: a simulated environment using OMNET++ [22] and a small proof-of-concept prototype using physical devices such as Sun SPOT [23]. More emphasis is placed on the simulation results rather than the real experiments using physical devices. This is because the simulation platform gives room for scalability analysis allowing a variety of bridge sizes and numbers of nodes, whereas the physical platform is limited to just 3 devices. Hence, the physical devices are used as a proof-of-concept to support the simulation results. The developed protocol also includes some inherent security features as part of the process of tackling DoSL attacks [25-26].

Our solution is evaluated based on how energy loss it eliminates as well as how it responds in the event of a DoSL attack. These sources of energy loss include overhearing, idle listening, control packets overhead and collisions [5]. The new protocol tackles each of these sources of energy loss in a unique and secure way. The research begins by identifying the requirements of the protocol, specifying these requirements and prioritizing them using a technique called MoSCoW which indicates four priority categories – a) Must have; b) Should have; c) Could have; d) Would have [24]. Furthermore, different designs of the semantics of the protocol are produced and discussed. Algorithms are then produced for the protocol. These algorithms are implemented on the OMNET++ simulator and on a small testbed with the Sun SPOT sensor devices. The language used for the simulation and device implementation are discussed critically based on different criteria. The paper also provides an evaluation of the Layered-MAC protocol in comparison to several existing MAC layer protocols.

The rest of this paper is organized as follows. Section 2 provides a summary of related work elsewhere. Section 3 lays emphasis on the software engineering aspects such as requirements gathering, functional and non-functional requirements of the protocol. Section 4 shows the results of the simulations including simulations under DoSL attack. Section 5 concludes the paper and discusses future work.

2 Related Work

It is pertinent to note that in the context of DoSL, several approaches exist to curb these attacks. However, most of them do not take energy efficiency into consideration and even when they do, throughput becomes a trade-off which could become counter-productive in the long run. The most notable existing approaches include Gateway-MAC (GMAC) [28], Hash-based scheme [29], Clustered adaptive rate limiting [3], Fake schedule switch scheme [30], Absorbing Markov chain (AMC) model [31], Secure wakeup scheme [32, 35], Zero knowledge protocol [33] and Cross layer mechanism [34].

One of the existing protocols that has geared towards energy efficiency as well as security is the GMAC protocol. GMAC protocol uses the idea of a central management where nodes are divided into clusters and each cluster has a gateway node. One of the

strategies used in tackling DoSL attacks is by understanding the impact of a failed node on the entire network lifetime. This is evidenced in [1] where the most critical node is assessed in terms of the impact of its elimination on the network lifetime. On the other hand, in [2] and [3], an intrusion detection scheme (IDS) is proposed whereby a DOS attacks are detected before it has any impact thereby making it preventive. In [11], focus is placed on creating hard-to-guess tokens/beacons which prevents attackers from easily guessing tokens that are aimed at depleting battery life. In [19], a cluster-based security protocol which uses digital signatures is proposed, however, this does not consider energy efficiency. Another protocol based on public-key cryptography is proposed in [20]. However, this protocol seems to introduce a lot of overhead that comes with key exchange and management.

A generic framework that optimizes the performance of existing clustering protocols such as UHEED by using Simulated Annealing and K-Beam algorithms is proposed in [21]. However, this is mainly aimed at clustering and routing protocols. In [22], the relationship between node density and certain network parameters such as the Received Signal Strength Indicator (RSSI) and the Link Quality Indicator (LQI) are analysed, with reference to DoSL attacks.

With regards to clustering, one protocol which considers energy efficiency is GMAC [13] and is also intended to guard against broadcast attacks which target the MAC layer. GMAC uses a cluster-based approach, thereby enhancing security, however the clustering limits the network architecture thereby making it have a low level of autonomy with regards to network architecture. The clustered adaptive rate limiting approach uses a host-based intrusion detection system to limit the rate of activity by the radio as a way of conserving energy and curbing against the broadcast attack with a downside if reducing throughput [4].

Furthermore, the hash-based scheme also uses clustering in addition to hashing but focuses on reducing the overhead involved in selecting a cluster head better than random vote scheme and round robin scheme [8].

Fake schedule switch scheme [12] as the name implies creates a false schedule and uses an offensive approach rather than a defensive approach by sending the wrong schedule when an acknowledgement is not received but one limitation is that it only applies to protocols that support acknowledgement-based communication.

The secure wakeup scheme [9] appears to have a mechanism where the packet can be inspected while the node is still in a low-power sleep state. This is achieved by the radio being able to hold a list of tokens and carry out an authentication. It would have been good to highlight how much energy is spent.

The Absorbing Markov Chain [14] works by attempting to predict the expected death-time of a sensor and using that as a benchmark to detect a DoSL attack by monitoring the network traffic. The limitation of this technique is that it is detective in nature and not corrective or preventive.

Anomaly detection is the technique used by Hierarchical collaborative model [2]. It attempts to achieve this not with one node but multiple nodes thereby balancing out the load. The downside is that achieving this requires a lot of packet overhead.

Zero-knowledge protocol [10] uses RSA key generation, hash generation and distribution and interlock protocol to achieve security, however with little or no focus on the energy costs of the technique. It does help against man-in-the-middle and replay attacks but no evidence to curbing against DoSL attacks.

In [37], to curb DoSL attacks, a combination of firefly algorithm, Hopfield neural network and RSA are applied in additional to the considering mobility of the sink node. The mobility of the sink node is based on the premise that in fixed-sink networks, nodes that are closer to the sink tend to drain energy faster as they act as a proxy for the rest of the nodes, and that mobility of the sink node will solve this problem.

In [38], k-nearest neighbour classification algorithm is implemented using Python libraries such as sci-kit learn, Numpy and Pandas. This machine learning algorithm uses data from incoming traffic to a node to detect a DoSL attack. Four traffic-related features are used to train the model in addition to heuristically determined rules and the results of the confusion matrix show an 87% accuracy.

In [11], the vulnerabilities present in MAC layer protocols such as SMAC, TMAC and B-MAC are highlighted. For SMAC, an attacker can use false SYN messages with longer time than the transmission frames. For B-MAC, an attacker can take advantage of the preambles. For TMAC, an attacker can take advantage of the adaptive timeouts.

3 Requirements Analysis

First, performance tuning was done on existing protocols to understand the impact of certain parameters like duty-cycling, beacon interval fraction and transmit power on metrics like energy consumption, number of transmitted packets. The performance tuning was done using a protocol called TunableMAC which was created using two languages – NED and C++ – and runs on OMNET++ framework as part of the Castalia simulator. NED was used to define the network including its parameters and gates while C++ was used to define the behaviour of the MAC protocol. The platform for these languages is OMNET++ and this was used alongside a framework for wireless sensor networks called Castalia. The C++ codes consisted of two files- a header file which contained a declaration of the variables and methods and another file which contained an initialization of the variables and implementation of the methods. The reverse engineering was done to understand the sequence and effect of the methods as well as the states of the variables. Hence a sequence diagram and state diagram are produced for the TunableMAC protocol. The diagrams provide a better understanding of where to insert the algorithms for the new protocol.

In building the new protocol, a traditional software development life cycle (SDLC) was used, particularly the **incremental model/iterative model**. This involved building the protocol in small increments. Each increment involved all the stages of the SDLC which are described briefly below:

- Requirements gathering/analysis. This stage involves understanding the problem and deciding on what needs to be done. In some cases, these two stages are split individually but considering the scope of this protocol, not many requirements are required. Hence the two stages can be combines into one. The requirement is then clearly specified.
- Design. This stage involves producing a blueprint of the internal workings of the system be it high-level or low-level design. One design could be a flow chart showing the flow of information in the protocol. Another design could be a sequence diagram showing the sequence of method calls for the new protocol. A class diagram showing the methods and variables of the new protocol is also an important design to include.
- Implementation. At this stage, the coding will be done either for the simulator in C++ and NED or for the Sun SPOT sensor in Java [27]. This stage involves testing the codes to first check that they meet the requirements and that they perform better than existing protocols at tackling DoSL attacks.

3.1 Requirements Identification

Problem Statement. DoSL attacks can have a strong negative impact on the life span of battery-powered wireless sensors. Considering that the radio is the major source of energy loss, these attacks take advantage of the MAC layer, which is responsible for access to the radio, and use certain techniques to prevent the radio from sleeping thereby reducing the lifespan of the sensor. While there have been proposed solutions and techniques to tackling these attacks, only one of these solutions (GMAC) has been incorporated into a protocol and tested on a real device. There is therefore a need for more MAC layer protocols that have a form of security against DoSL attacks while aiming at maintaining the same or similar level of throughput and latency as protocols that do not have these security measures.

Protocol Requirements. Only two levels of headings should be numbered. The protocol should be able to detect a DoSL attack and take measures to reduce its impact. In a case where the protocol is not able to detect the DoSL attack on time, it should take measures to reduce the other sources of energy loss that are not because of an attack. In this way the sensor can have enough energy to continue functioning until it detects the attack. To detect the attack, the first step is to understand the possible attack strategies that could be used:

- Attack from an unauthorized authenticated node – In this scenario, the node's identity is verified and valid, however the action of the node is not authorized.
- Attack from an authorized and authenticated node – This is a more dangerous scenario as it is more difficult to detect such a node. In this case the entire identity has been compromised. Sybil node attacks fall under this category.
- Attack from an unauthenticated and unauthorized node – This is the least dangerous of the three strategies.

Then, it is important to identify the target because an attack on a sink node would have more impact than an attack on a cluster head. Similarly, an attack on a cluster head would have more impact than an attack on a normal node. After identifying the target, the next step is to get some data about the attacker node beginning with its address and RSSI and LQI for that node. After the node has been identified, the next step is to isolate the attacker and make the network inaccessible by that node.

The life cycle of the MAC layer is divided into four stages as follows:

- The start-up stage involves initializing the variables with information about the packets, sensors, and communication as well as getting the node to sleep if there is no information from the radio layer or there is nothing left in the buffer to send to the network layer. At this stage, the cluster heads will also be set up.
- The transmit stage involves transmitting information received from the network layer to the radio layer or transmitting information received from the radio layer to the network layer.
- The carrier sensing stage is before transmitting, when a node may want to apply some CSMA techniques or use request-to-send (RTS) or clear-to-send (CTS) packets to avoid collisions and overhearing. While RTS/CTS could be helpful in avoiding collisions, it has one disadvantage of increasing the control packet overhead which further increases the energy consumption. CSMA on the other had has some back-off techniques that work based on probability and may not always be accurate and could lead to deadlock problems where a node is not able to transmit because if waiting endlessly for an opportunity to transmit.
- The receive stage involves staying in a receive mode and waiting for information from the radio layer which is coming from another node. The data received must be checked to know the type of data (control packet or actual data).

3.2 Functional Requirements

As mentioned earlier, the MoSCoW technique is used to prioritize the requirements based on the following four categories:

Must Have. The sink node should be able to get the RSSI and LQI values of every sensor it receives data from. Each node should know how far it is from the sink node and use that to decide who becomes a cluster head. The protocol should be able to adjust the duty cycle at run-time based on the traffic. The protocol should allow cluster heads to be appointed and rotated at intervals if need be. The protocol should allow for a node to be isolated from the network when it has been discovered to be an attacker node.

Should Have. Nodes should be able find the least expensive route to communicate their data. Cluster heads should be able to communicate using code division multiple access.

Could Have. Supervised learning could be applied on the data collected from the base station.

Will not Have. Nodes cannot be powered by Solar energy.

3.3 Algorithm Design

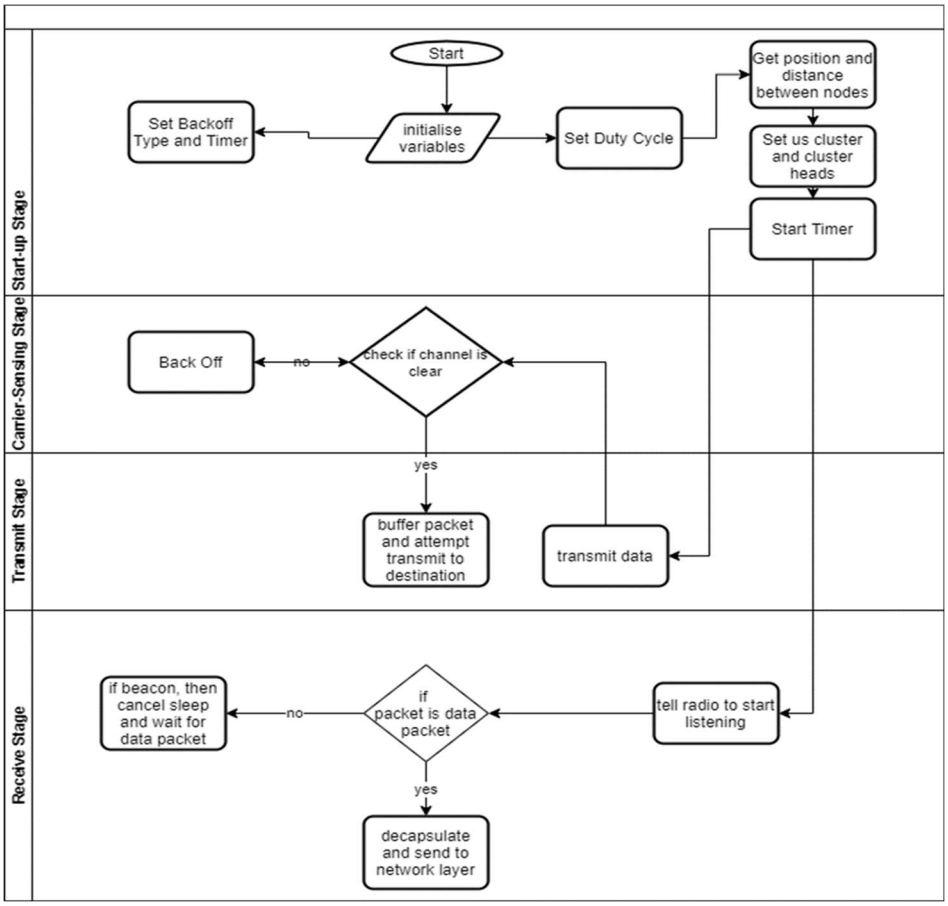

Fig. 1. Flow chart showing the Layered-MAC communication.

Algorithm for the Layered-MAC Protocol. The algorithm flow chart depicted in Figure 1 highlights the steps involved in setting up the MAC layer before the communication process starts. More specifically, the main steps include the following actions:

- MAC layer receives the number of nodes from the application layer.
- Sink node gets the distance of all nodes.
- Sink node appoints the node with closest proximity as a cluster head.
- If a cluster head has more than 5 nodes assigned to it, then another cluster head is appointed.
- Nodes must only communicate to their cluster heads not to other nodes.

- The cluster head then passes the information to the sink nodes. If the sink node is too far from the cluster head, then the data is passed to other cluster heads closer to the sink node.
- After every 5 minutes, a new cluster head is appointed to ensure security and to also manage the energy efficiency.

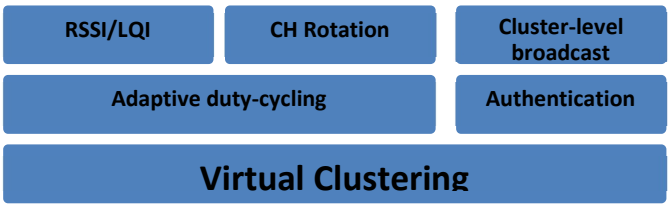

Fig. 2. Component architecture model of the Layered-MAC protocol.

Figure 2 shows the conceptual design of the Layered-MAC protocol. The lowest layer is virtual clustering which involves grouping the sensors based on their proximity. The benefits of keeping the clusters virtual is that if the position of the sensor is changed, then the cluster can be reconfigured [7]. The adaptive duty cycling is adopted form TMAC whereby the duty cycle automatically adjusts to the amount of traffic.

Algorithm for Position and Distance of Nodes. Getting the positions and distance between nodes involves using a points-based system/GPS to get the x and y coordinates of the sensor. After getting the x and y coordinates for each node, the next step would be to calculate the distance between the two nodes. The distance between the nodes is calculated using the following formula based on Pythagorean theorem:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (1)$$

Where d is the distance; x_2 is the x coordinate for node 2; x_1 is the x coordinate for node 1; y_2 is the y coordinate for node 2; y_1 is the y coordinate for node 1. However, this method may not work for sensors located in areas where the GPS does not work. Another way to achieve this is by using the Castalia framework to get the RSSI and the LQI.

The algorithm for position and distance of nodes involves the following steps:

- Each node waits for a random time and makes a broadcast.
- The broadcast packet contains the schedule, and each node follows the schedule it receives.
- Each node also keeps the RSSI/LQI of the packet it receives.
- Get position of nodes as described above.
- Base station creates a map of the distance between nodes using RSSI.
- Identify best distance from each node.
- Create virtual clusters based on best distance.

- Use CDMA to communicate between cluster heads.
- Stop.

Creating a map of the distance between the nodes by the base station involves creating a matrix that maps the distance between each node. For example, if there are ten nodes, each node maps the distance with 9 other nodes. The mapping is based on the RSSI and LQI values of the sensor nodes. Based on the map, the best distance for each node is then calculated. The reason for using distance is to enhance the energy efficiency in the nodes when transmitting data. Using the best distance, the clusters are then created with cluster heads managing nodes within the closest distance. Only the cluster heads communicate directly with the base station/sink node. The cluster heads will be changed at intervals to increase security. Thus, the algorithm for cluster creation involves the following three steps:

- If nodes have the same schedule, they belong to the same cluster.
- Cluster nodes can only communicate with their cluster head.
- Cluster heads then communicate with the sink node.

Finally, CDMA is used only for communication between cluster heads to ensure security and prevent DoSL attacks. This stage has more to do with the physical layer.

4 Protocol Simulation

4.1 Test Plan

Table 1. Test cases and corresponding actions.

ID	Test Case	Description	Actions
1	Collision	This checks that the protocol plays a role in reducing reduction	Compare the number of transmitted and received packets.
2	Control Overhead	This checks that the protocol reduces the control overhead	Track the size of data used for control overhead
3	Idle Listening	This checks that the protocol plays a significant role in reducing	Measure how long a node stays idle before transmitting.
4	Overhearing	This checks if the protocol reduces the chances of a node hearing a packet meant for another node.	Measure how much energy is wasted listening to packets meant for other nodes.

4.2 Experimental Results

This subsection presents the OMNET++ simulator results. Furthermore, the results for the Layered-MAC protocol are then compared with the results from SMAC and TMAC.

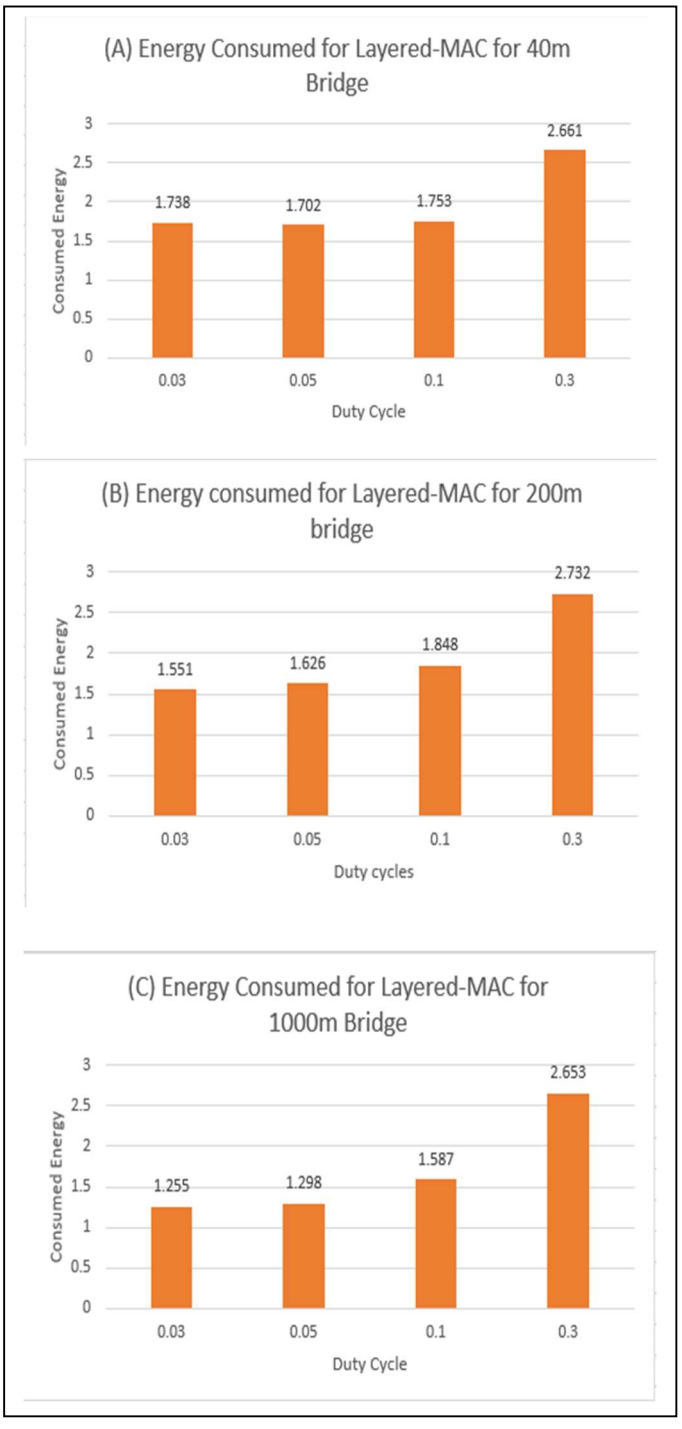

Fig. 3. Energy Consumption for Layered-MAC on 40m, 200m and 1000m Bridge.

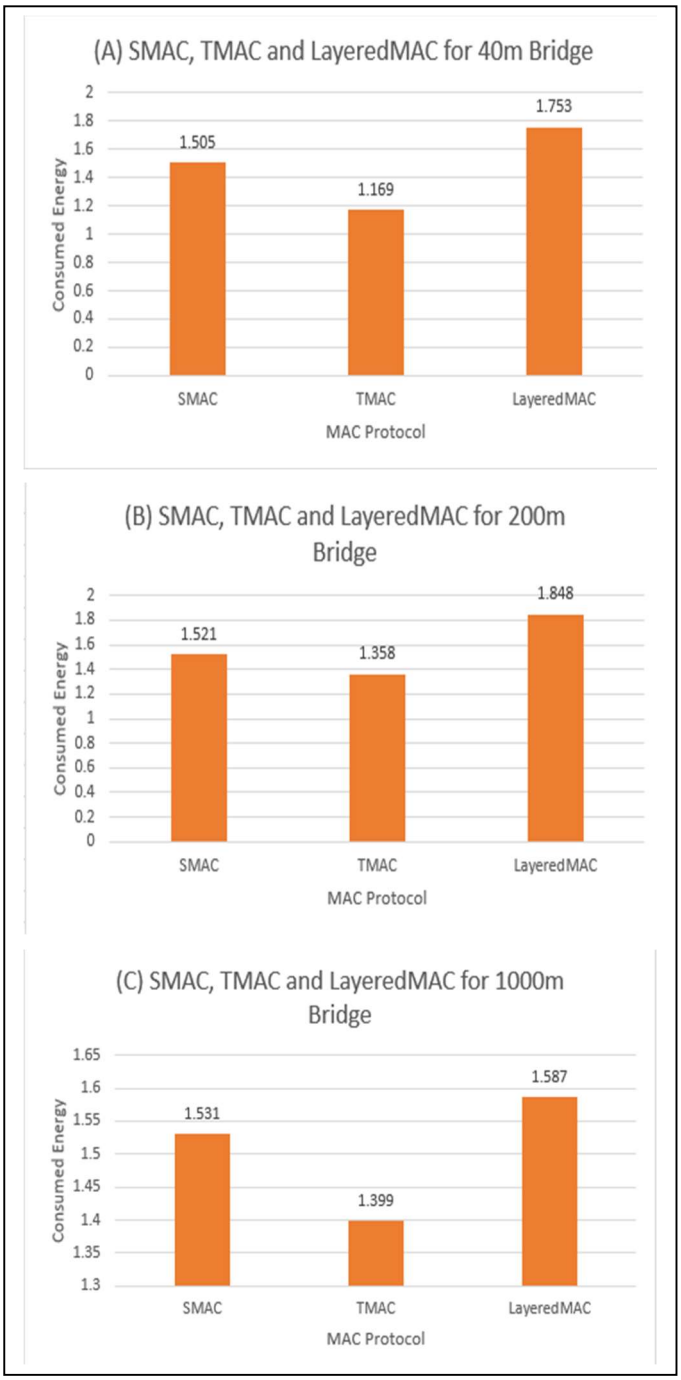

Fig. 4. Comparison for SMAC, TMAC and Layered-MAC on 40m, 200m and 1000m Bridge.

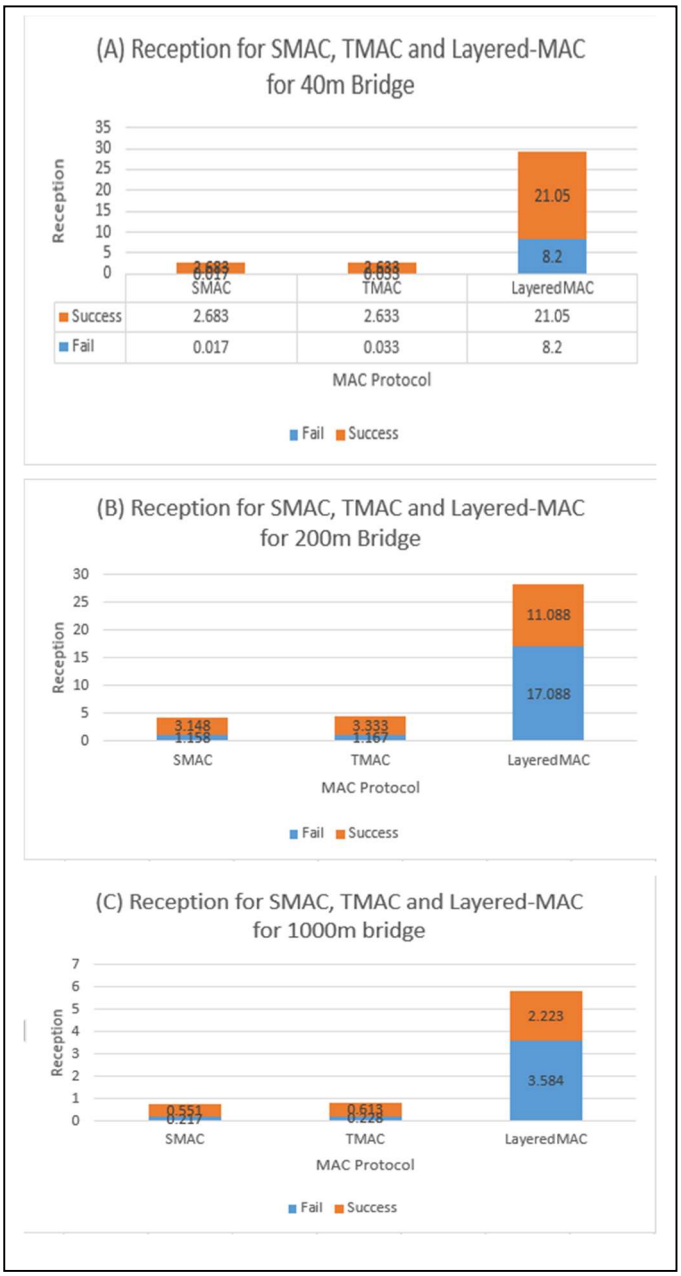

Fig. 5. Reception Comparison for SMAC, TMAC and Layered-MAC on 40m, 200m and 1000m Bridge.

Consumed Energy. The first graph in Figure 3(a) shows the energy consumed for Layered-MAC under different duty cycles in a 40m bridge. The consumed energy increases as the duty cycle increases. The second graph in Figure 3 shows the energy consumption

for Layered-MAC in a 200m bridge. The energy consumed in the 200m appears less that the energy consumed in 40m bridge. The third graph in Figure 3 shows the energy consumption for the 1000m bridge for which energy consumption is the lowest.

Energy Comparisons. The graphs above show a comparison of the Layered-MAC with two other protocols – TMAC and SMAC. The comparison is based on the energy consumption of the sensors.

The graph in Figure4(a) shows that TMAC consumes the least energy while Layered-MAC consumes the most energy. However, this does not take into consideration the packet reception at the sink.

In the second graph, energy is compared on a 200m bridge for SMAC, TMAC and Layered-MAC and Layered-MAC consumes the most energy at 1.848 while TMAC consumes the least energy at 1.358.

Reception Comparisons. The second graph in Figure 5(b) shows that the Layered-MAC has a better reception at the sink in terms of data throughput with success of 11.088 and failure of 17.088 compared to SMAC and TMC which are significantly lower. The third graph (Figure 5c) shows that the Layered-MAC has a better reception at the sink in terms of data throughput. There is a greater amount of succeeded and failed packets than in TMAC and SMAC.

In conclusion, while the energy consumption in total is higher for the Layered-MAC than TMAC and SMAC, the reception results show that the Layered-MAC provides better performance and even energy consumption when measured in terms of energy per successfully received packets.

Another interesting observation is that although more energy is spent as the bridge size increases, the successfully received packets show a downward trend for both TMAC and SMAC which is slightly different for the Layered-MAC.

Hence, on a 40m bridge for Layered-MAC, 1.752 is spent for 21.05 received packets. On a 200m bridge, 1.848 is spent for 11.088 received packets and 1.587 for 2.223 received packets on a 1000m bridge. Summing up the received packets for TMAC and SMAC put together still does not get up to half the reception for Layered-MAC.

4.3 Simulation results for the protocols under attack

Simulation Scenario. Only two levels of headings should be numbered. The Layered-MAC protocol is used alongside two other protocols, SMAC and TMAC in the simulation to understand the energy consumption and reception under an attack.

For this scenario, a 200m bridge is used with 3 of the nodes as compromised malicious nodes which use a broadcast attack to stop nodes from sleeping. The broadcast attack is carried out by continuously flooding the network with broadcast messages from these three nodes. The simulation is about the structural health monitoring of a bridge. Sensing nodes are placed in a grid with a sink node in the middle. A car moves on the bridge every five minutes and triggers nodes along its path.

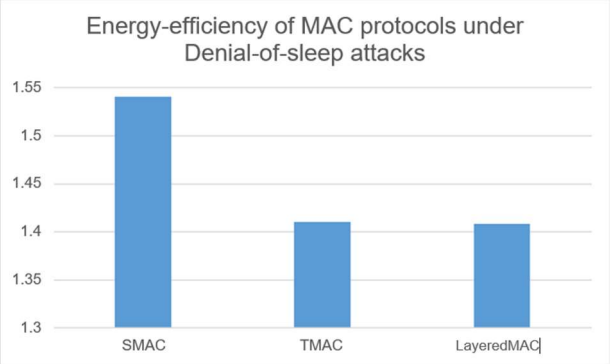

Fig. 6. Energy efficiency of MAC protocols under DoSL attack simulation.

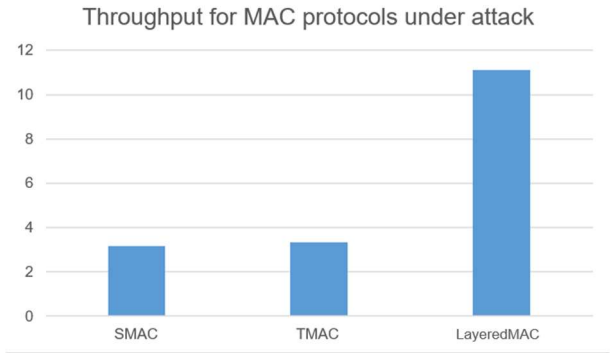

Fig. 7. Throughput of MAC protocols under DoSL attack.

Figures 6 and 7 show the energy efficiency and throughput for 3 MAC protocols respectively, including the new Layered-MAC protocol, under a DoSL attack. GMAC has not been used in the comparisons for two reasons. Firstly, because there is no model of it in the Castalia simulator and secondly because GMAC does not give any considerations to reception(throughput) at the sink.

Figure 6 shows that under the DoSL broadcast attack, SMAC consumes the highest amount of energy at 1.54 Joules. TMAC consumes much lower energy than SMAC at 1.41 Joules. The Layered-MAC consumes the least energy slightly below TMAC at 1.408 Joules. The fixed duty cycling for SMAC justifies the relatively high energy consumption. TMAC on the other hand supports adaptive duty-cycling, hence there is better energy efficiency than SMAC. The Layered-MAC on the other hand goes a step further than just adaptive duty-cycling to also detect signal strength and link quality, hence the slightly better energy-saving than TMAC.

In terms of throughput, Figure 7 shows significant difference in throughput between the Layered-MAC and TMAC and SMAC combined. This is partly because of the

Layered-MAC's ability to detect a malicious node and adjust duty cycling to by-pass/isolate the malicious node.

5 Conclusion and Future Work

It is important to look at how the research questions discussed in the introduction have been addressed. CSMA with collision avoidance is used to prevent collisions when two nodes are trying to communicate at the same time. Acknowledgements are not required to reduce the number of control packets. Minimizing the number of broadcasts by applying distance measurements and using a routing-by-rumor approach helps reduce overhearing. Idle listening is reduced by adaptive duty cycling. One of the benefits of this new protocol is that it is multi-layered and touches on different aspects. First it deals with virtual clustering [21], authentication, RSSI and LQI measurements which helps with security. It also measures the distance between nodes and supports adaptive duty-cycling which helps with energy consumption. Furthermore, our Layered-MAC protocol is tested alongside two other protocols under DoSL attacks and the performance as well as the energy efficiency are significantly better.

One of the areas for future work is to investigate how machine learning techniques can be applied to data collected by the sensor nodes to further enhance energy efficiency and security against DoSL attacks. The algorithms will be run on the machine connected to the base station and then the output from the learning is then passed across to the nodes as an update.

Acknowledgement

This work was partially supported via a doctoral research scholarship grant by the University of Westminster.

References

1. Yuksel, A., Uzun, E., Tavli, B.: The impact of elimination of the most critical node on Wireless Sensor Network lifetime. In: IEEE Sensors Applications Symposium, Proceedings, pp. 1-5, IEEE, Zadar, Croatia (2015).
2. Wang, J., Jiang, S., Fapojuwo, A. O.: A protocol layer trust-based intrusion detection scheme for wireless sensor networks. In: Mauri, J., Han, G. (eds.) Smart Communication Protocols and Algorithms for Sensor Networks, vol. 17, no. 6, Sensors, Switzerland (2017).
3. Vaseer, G., Ghai, G., Patheja, P. S.: A novel intrusion detection algorithm: An AODV routing protocol case study. In: Proc. 2017 IEEE Int. Symposium on Nanoelectronic and Information Systems, *iNIS 2017,* Vols. 2018-Febru, pp. 111-116, IEEE, Bhopal, India (2018).
4. Uher, J., Mennecke, R. G., Farroha, B. S.: Denial of Sleep attacks in Bluetooth Low Energy wireless sensor networks. In: Proc. IEEE Military Communications Conference MILCOM, pp. 1231-1236, IEEE Baltimore, MD, USA (2016).
5. Sinha, P., Jha, V. K., Rai, S. K., Bhushan, B.: Security vulnerabilities, attacks and countermeasures in wireless sensor networks at various layers of OSI reference model: A survey.

In: Proceedings of IEEE International Conference on Signal Processing and Communication, ICSPC 2017, Vols. 2018-Janua, no. July, pp. 288-293, IEEE, Coimbatore, India (2017).
6. Shakhov, V., Koo, I., Depletion-of-battery attack: Specificity, modelling and analysis. In: Spec. Issue on Security in IoT Enabled Sensors, Sensors, vol. 18, no. 6, Switzerland (2018).
7. Isaiadis, S., Getov, V., Integrating Mobile Devices into the Grid: Design Considerations and Evaluation. In: Proc. of Euro-Par 2005 Conference, LNCS, vol. 3648, pp. 1080-1088. Springer (2005).
8. Qiu L., Jiang W., Zhang W, Li P.: Wireless Injection Attacks Based on Fake Data Injection in TinyOS, In: Proceedings - International Symposium on Parallel Architectures, Algorithms and Programming, Vols. 2016-Janua, pp. 236-242, PAAP, Nainjing, China (2016).
9. Pawar P., Nielsen R. H., Prasad N. R., Prasad R.: GSHMAC: Green and Secure Hybrid Medium Access Control for Wireless Sensor Network. In: Wireless Personal Communications, vol. 100, no. 2, pp. 267-281, 2018.
10. Osanaiye, O. A, Alfa, A. S., Hancke G. P.: Denial of Service Defence for Resource Availability in Wireless Sensor Networks. In: IEEE Access, vol. 6, pp. 6975-7004, IEEE (2018).
11. Islam, M.N.U., Fahmin, A., Hossain, M.S. et al. Denial-of-Service Attacks on Wireless Sensor Network and Defense Techniques. Wireless Pers Commun 116, 1993–2021 (2021). doi: 10.1007/s11277-020-07776-3
12. Krentz K. F., Graupner H.: Denial-of-Sleep-Resilient Session Key Establishment for IEEE 802.15.4 Security: From Adaptive to Responsive. In: Proc. 2018 Int. Conference on Embedded Wireless Systems and networks, pp. 25-36, EWSN, Madrid, Spain (2018).
13. Krentzn, K.F., Graupner H., Meinel C.: Countering Three Denial-of-Sleep Attacks on ContikiMAC, In: Proc. 2017 Int. Conference on Embedded Wireless Systems and networks, pp. 108-119, Junction Publishing US (2017).
14. Krentz K. F., Meinel C.: Denial-of-sleep defenses for IEEE 802.15.4 coordinated sampled listening (CSL), Computer Networks, vol. 148, pp. 60-71, Elsevier (2019). doi: 10.1016/j.comnet.2018.10.021
15. Udoh, E.: An Energy Aware and Secure MAC Protocol for Tackling Denial of Sleep Attacks in Wireless Sensor Networks. PhD Thesis, University of Westminster (2019)
16. Gelenbe E., Kadioglu Y. M.: Battery Attacks on Sensors. In: Proc. Int. Symposium on Computer and Information Sciences, Security Workshop, pp. 1-10, Springer (2018).
17. Gehrmann, C., Tiloca M., Hoglund R.: SMACK: Short message authentication check against battery exhaustion in the Internet of Things. In: Proc. 2015 12th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON), pp. 274-282, IEEE (2015) doi: 10.1109/SAHCN.2015.7338326
18. Rani S., Naidu S.K.: Mitigation of Energy Depletion in Wireless Ad-hoc Sensor Networks through Path Optimization. In: Int. Journal of Computer Networks and Applications, vol. 2, no. 1, pp. 1-11, IJCNA (2015).
19. Ferng H. W., Khoa N.M.: On security of wireless sensor networks: a data authentication protocol using digital signature. In: Wireless Networks, vol. 23, no. 4, pp. 1113-1131, Springer (2017).
Capossele A. T., Cervo V., Petrioli C., Spenza D.: Counteracting Denial-of-Sleep Attacks in Wake-up-radio-based Sensing Systems. In: Proc. 2016 13th Annual IEEE Int. Conference on Sensing, Communication, and Networking (SECON), 2016, pp. 1-9, IEEE, London, UK (2016). doi: 10.1109/SAHCN.2016.7732978
20. Udoh, E. Getov, V.: Performance Analysis of Denial-of-Sleep Attack-Prone MAC Protocols in Wireless Sensor Networks. In: Proc. 2018 UKSim-AMSS 20th Int. Conference on Computer Modelling and Simulation, pp. 151-156, IEEE, Cambridge, UK (2018). doi: 10.1109/UKSim.2018.00038

21. Boulis, A.: Castalia A Simulator for Wireless Sensor Networks and Body Area Networks User's Manual, Version 3.2. NICTA (2011).
22. Horveliur C. M.: Sun SPOTs: A Great Solution for Small Device Development. https://www.oracle.com/technetwork/server-storage/ts-4868-1-159029.pdf last accessed 2018/11/20
23. Agile Business Consortium (2014). The DSDM Agile Project Framework. Available at: https://www.agilebusiness.org/content/moscow-prioritisation. Last accessed 2018/11/20.
24. Udoh E., Getov, V.: Performance and Energy-Tuning Methodology for Wireless Sensor Networks Using TunableMAC. 2020 Int. Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI), pp. 1-5, IEEE, Sharjah, UAE (2020). doi: 10.1109/CCCI49893.2020.9256744.
25. Udoh E. Getov V.: Proactive Energy-Efficiency: Evaluation of Duty-Cycled MAC Protocols in Wireless Sensor Networks. In: 2018 International Conference on Computer, Information and Telecommunication Systems (CITS), pp. 1-5, IEEE, Colmar, France (2018). doi: 10.1109/CITS.2018.8440194
26. Oracle Labs, "Sun SPOT Programmers Manual," 2011. Available at: https://bit.ly/2DlFFiy last accessed 2019/01/01.
27. Brownfield, M. I.: Energy-efficient Wireless Sensor Network MAC Protocol. In: PhD Thesis Virginia Polytechnic Institute and State University (2006).
28. Pirretti M., Zhu S, Vijaykrishnan N, Mcdaniel P, Kandemir M., Brooks R.: The Sleep Deprivation Attack in Sensor Networks: Analysis and Methods of Defense, In: Int. J. Distrib. Sens. Networks, vol. 2, no. 3, pp. 267–287, SAGE (2006).
29. Chen C., Hui L., Pei Q., Ning, L., Qingquan P.: An effective scheme for defending denial-of-sleep attack in wireless sensor networks. In: 5th Int. Conference on Information Assurance and Security, IAS. IEEE Xi'an China (2009).
30. Bhattasali T., Chaki R., AMC Model for Denial of Sleep Attack Detection. In: Journal of Recent Research Trends (JRRT) Cornell University arXiv Prepr. arXiv1203.1777 (2012).
31. Falk R., Hof H.J.: Fighting insomnia: A secure wake-up scheme for wireless sensor networks. In: Proc. Third International Conference on Emerging Security Information, Systems and Technologies, pp. 191-196, (2009). doi: 10.1109/SECURWARE.2009.36
32. Naik, S. Shekokar, N.: Conservation of energy in wireless sensor network by preventing denial of sleep attack. In: Procedia Computer Science, Elsevier (2015).
33. Hsueh, C. T, Wen, C. Y., Ouyang. Y. C.: A secure scheme against power exhausting attacks in hierarchical wireless sensor networks. In: IEEE Sens. J., vol. 15, no. 6, pp. 3590–3602, IEEE (2015).
34. Montoya, M., Bacles-Min, S., Molnos A, Fournier, J.J.: SWARD: A Secure Wakeup RaDio against Denial-of-Service on IoT devices. 11th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec'18), CCSD, Stockholm, Sweden (2018). doi: 10.1145/3212480.3212488.cea-01922847
35. Raymond D. R., Midkiff S. F., Denial-of-Service in Wireless Sensor Networks: Attacks and Defenses. In: IEEE Pervasive Computing, vol. 7, no. 1, pp. 74-81, Jan.-March IEEE (2008). doi: 10.1109/MPRV.2008.6.
36. Fotohi, R., Firoozi Bari, S. A novel countermeasure technique to protect WSN against denial-of-sleep attacks using firefly and Hopfield neural network (HNN) algorithms. J Supercomputing 76, 6860–6886 (2020). doi: 10.1007/s11227-019-03131-x
37. Desnitsky V.: Approach to Machine Learning based Attack Detection in Wireless Sensor Networks. 2020 International Russian Automation Conference (RusAutoCon), pp. 767-771, IEEE Sochi, Russia, (2020). doi: 10.1109/RusAutoCon49822.2020.9208085.