Defense against Black Hole Attacks in Wireless Sensor Network with Anomaly Report Cycling

Marcel A. Vieira
Information System Security Division
Naval Undersea Warfare Center
Newport, RI, USA
MVieira4@umassd.edu

Hong Liu
Department of Electrical and Computer Engineering
Univesity of Massachusetts Dartmouth
North Dartmouth, MA, USA
HLiu@UMassD.edu

***Abstract*—Wireless Sensor Network (WSN) becomes the dominate last-mile connection to cyber-physical systems and Internet-of-Things. However, WSN opens new attack surfaces such as black holes, where sensing information gets lost during relay towards base stations. Current defense mechanisms against black hole attacks require substantial energy consumption, reducing the system's lifetime. This paper proposes a novel approach to detect and recover from black hole attacks using an improved version of Low-Energy Adaptive Clustering Hierarchy (LEACH) protocol. LEACH is an energy-efficient routing protocol for groups of battery-operated sensor nodes in hierarchy. A round of selection for cluster heads is scheduled in a set time. We propose to improve LEACH with Anomaly Report Cycling (ARC-LEACH), tradeoff between security strength and energy cost. ARC-LEACH absorbs an attack when it occurs by rotating cluster heads to reestablish communication and then sending a message from the base station to coordinate all nodes against the malicious nodes. ARC-LEACH actively blocks malicious nodes while leveraging the resilience of LEACH for stronger resistance to blackhole attacks. ARC-LEACH can provide more defense capability when under attack from multiple malicious nodes that would otherwise be defenseless by LEACH, with only minor increase in energy consumption.**

Keywords—network security, wireless local area network (WLAN), wireless sensor network (WSN), cyber-physical system (CPS), Internet of Things (IoT)

I. INTRODUCTION

As the first hierarchical routing protocol for wireless sensor network (WSN), *Low-Energy Adaptive Clustering Hierarchy (LEACH)* protocol remains active in research since its birth [1]. WSN connects large number of sensor nodes with limited resources of computing and communication, most battery operated. Deployed randomly to cover a wide area for monitoring, distributed sensor nodes often form clusters where each selected Cluster Head (CH) aggregates data collected by Cluster Members (CMs) and transmits information to a Base Station (BS) for analysis at a data center. Due to power limitation and replacement cost of batteries, the main objective of LEACH is energy conservation. LEACH creators further improved energy efficiency by allocating all decisions such as CH selection to the BS, *LEACH-Centralized* [2]. Over two decades, LEACH variants have kept emerging to improve different factors such as even energy dissipation and equal cluster distribution [3].

WSN is inherently vulnerable to cyber-attacks due to their open nature and resource limit. However, LEACH and its variants have been focused on their performance for network lifetime and communication throughput [4]. A 2017 survey on successors of LEACH protocol revealed that the research community has placed security as the second lowest priority out of eight LEACH objectives [3]. As WSN becomes the dominate last-mile connection to cyber-physical systems (CPS) and Internet-of-Things (IoT), the stakes grow higher for adversaries to use WSN as a pathway for penetrating critical infrastructures. About half a dozen papers contribute to strengthening security for LEACH literature [3][5]. These LEACH-security protocols aim at meeting the security requirements similar to those of traditional wireless local area network (WLAN): *C.I.A. goals* standing for *Confidentiality, Integrity* (of both data and source as well as Freshness), and *Availability*. Hostile environments and resource constraints greatly challenge their efforts to secure WSN, leading to weak security with lightweight cryptography or diverting from the main objective of LEACH for energy efficiency [5].

Moreover, the hierarchical network structure that LEACH and its variants are based on makes WSN extremely vulnerable to denial-of-service (DoS) attacks by targeting at CHs or BS [6]. Particularly, WSN holds *the Holes Problem* including sensing coverage holes, network routing holes, transmission jamming holes, sink holes, black/gray holes, and worm holes [7]. This paper introduces a new energy-efficient routing protocol to defend WSN against black hole attacks, called *Anomaly Report Cycling Low-Energy Adaptive Clustering Hierarchy (ARC-LEACH)* protocol. ARC-LEACH leverages the fact that LEACH protocol can reduce the effects of a single black hole attack by rotating cluster heads. This work focuses on mitigating collaborative multiple black hole attacks against hierarchical routing protocols in homogeneous WSN with resource-constrained sensor nodes such as LEACH. The main contributions of our work are as follows:

- Devise an innovative scheme to detect and recover from black hole attacks in resource-constrained WSN.
- Leverage edge-computing technology to protect WSN for both security strength and energy efficiency.
- Add security in CPS/IoT design.

This work was supported in part by the National Science Foundation (NSF) Innovations in Graduate Education (IGE) program under grant #2105718 Graduate Education in Cyber-Physical Systems Engineering.

II. Related Work

The diversity of WSN, in technology characteristics and application domains, makes WSN fall out the scope of the well-established WLAN security standard such as IEEE 802.11i. The 2008 NIST guide to securing legacy IEEE 802.11 wireless networks withdrew in 2018 [8], still not superseded as of today. A small community has been developing security protocols for LEACH, listed below in chronical order.

SLEACH is the first protocol to add security in LEACH [9]. It builds on SPINS, security for general WSN [10], using lightweight cryptography against outsiders. The protocol fails on insider attacks and misses protection in several scopes such as cluster formation phase.

SecLEACH adds more security protections than SLEACH like sink hole attacks and selective forwarding attacks [11]. Notice the subtle differences between sink holes and black holes: the former as an outsider impersonates a base station to lure traffic while the latter as an insider compromises a cluster head to stop relay traffic towards a base station. Although stronger security, SecLEACH performs poorly in terms of network lifetime.

SC-LEACH enhances the basic LEACH in two aspects: one is security using pre-shared key pairs, and the other is control to produce the optimal number of CHs in every round [12]. Due to the lack of details, its security strength is hardly justified [5].

Armor-LEACH resolves the energy-efficiency problem of SecLEACH with a time-controlled clustering algorithm [13]. However, it wastes bandwidth on the large number of control messages.

MS-LEACH resolves the limitations of SLEACH to some extent [14]. It provides data confidentiality and source authentication in sensor nodes to CH. It also outperforms SLEACH in many aspects: security strength, system lifetime, and network throughput. However, pairwise key for scheduling consumes much more power at CHs.

Kodali et al demonstrate a multi-level secure LEACH protocol using RC-4. Their simulation result in NS-3 is promising [15].

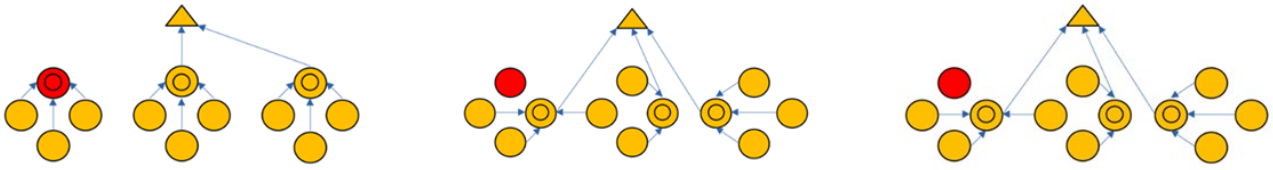
LEACH resistance to black/gray hole attacks, thanks to its dynamic CH selection at each round, is analyzed in [16]. However, its reactive defense is slow and is defenseless against collaborative multiple black hole attacks. Solutions for other types of wireless sensor networks [17][18][19][20] should be investigated to defend LEACH. Recently, a black hole detection scheme has been developed in Max-LEACH [21].

We also consider other network structures of resource-constrained WSN. Specifically, we explore the applicability of our security proposal to Hybrid Energy-Efficient Distributed clustering (HEED) [22] and Distributed Hierarchical Agglomerative Clustering (DHAC) [23].

III. Anomaly Report Cycling

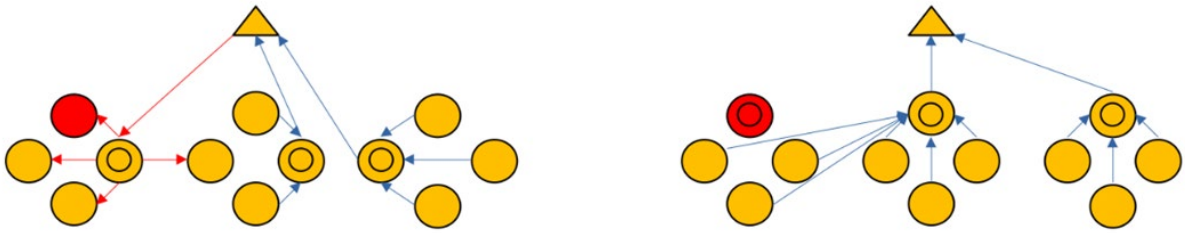
ARC-LEACH aims to take a non-conventional approach to network security. Rather than preventing an attack in its entirety ARC-LEACH aims to use the black hole attack to find the malicious node and neutralize it to prevent future attacks. Although this approach has some packet loss it eliminates the need to send extra packets to find the malicious node. The extra packet loss incurred is only during the initial attack making the effect negligible if there are many rounds. The minor performance hit combined with the major reduction of packets transmitted makes securing a network from a blackhole attack less of a tradeoff between security and energy cost.

ARC-LEACH builds on basic LEACH protocol behaving similarly as shown in Figure 1, except how it responds to black hole attacks illustrated in Figure 2. ARC-LEACH adds a few bytes to a standard LEACH packet that contains a sequence number. The sequence number is an indicator of the number of packets sent up until that point allowing the sink to tell how many packets have been sent between two points in time.



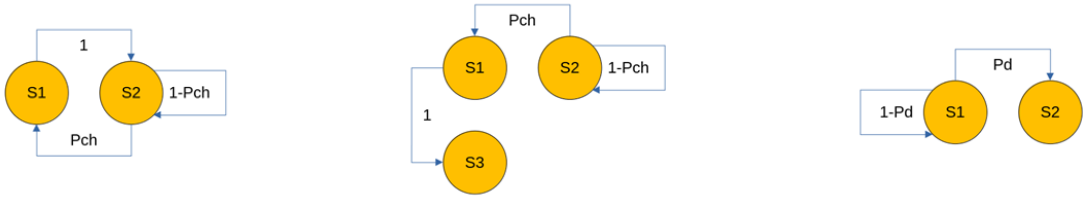
(a) Round 1 (b) Round 2 (c) Round 3

Figure 1: LEACH Operation under Black Hole Attack



(a) Blacklist Packet (b) Blacklist Node

Figure 2: ARC-LEACH Operation under Black Hole Attack



(a) LEACH (b) ARC-LEACH (c) Alternate Routing

Figure 3: Protocol State Diagrams

When an attack occurs ARC-LEACH absorbs the initial attack causing all packets in a round to potentially be lost. The following round ARC-LEACH uses the cluster head rotation from LEACH to reestablish communication with the sink [1]. With communication reestablished to the sink packets can continue to flow. Once the sink receives the next packet it will be able to see a gap in the packets sent allowing the sink to determine if the last cluster head was malicious.

The sink will not deem a previous cluster head malicious immediately if a gap is detected in data from a single node. The sink will wait until it detects a gap in data from a second node. Once two gaps are detected caused by the same cluster head, that cluster head is deemed malicious.

Upon realizing that the previous cluster head was malicious the sink sends out a blacklist packet telling all nodes to blacklist the malicious node. When a node is blacklisted, it will not be selected as the cluster head for a node even if it seems like the best option. The blacklist packet is sent to each of the cluster heads that are active during that round. When each cluster head receives the blacklist packet it will add the malicious node to its own blacklist then relay the blacklist packet to the rest of the nodes in the cluster. By disseminating the blacklist packet to all nodes in the network it ensures that the malicious node is unable to become cluster head again.

ARC-LEACH when it comes to power consumption has similar power consumption to LEACH. The small difference in power consumption is due to the packet size increase being negligible, about 8 bytes added to a 2kB packet. ARC-LEACH also only has extra transmissions after an attack, at all other times it has the same number of transmissions as LEACH.

ARC-LEACH works best when the cluster head is consistently rotated. If the cluster head is not rotated often such as in Max-LEACH [21], ARC-LEACH will not be as effective. The cluster head should be rotated often because that reduces the number of packets that are lost in the first round after the initial attack.

IV. Energy Consumption and Packet Loss

To estimate energy consumption in WSN, we take the calculations from the basic LEACH [1]. *Equations (1) and (2)* show that as packet size increases, so does energy consumption: transmission (Tx) increases power as distance length (d) exponentially while receiving (Rx) consumes energy as message size (k) linearly but to d independently.

$$E_{Tx}(k,d) = E_{Tx-elec}(k) + E_{Tx-amp}(k,d) \quad (1)$$
$$E_{Tx}(k,d) = E_{elec} \cdot k + \epsilon_{amp} \cdot k \cdot d^2$$

$$E_{Rx}(k) = E_{Rx-elec}(k) \quad (2)$$
$$E_{Rx}(k) = E_{elec} \cdot k$$

The basic LEACH uses *Equation (3)* to show that the energy required to send a packet directly to the sink is greater than the energy required to send the packet between multiple nodes before reaching the sink. This holds true if the distance between nodes is greater than one meter.

$$E_{Tx-amp}(k, d = d_{AB}) + E_{Tx-amp}(k, d = d_{BC}) \quad (3)$$
$$< E_{Tx-amp}(k, d = d_{AC})$$
$$d_{AB}^2 + d_{BC}^2 < d_{AC}^2$$

The equations provided in the LEACH proposal are for a single transmission. When finding the energy consumption over the life of a network multiple transmissions should be considered. This means the energy of a single transmission should be multiplied by the total number of transmissions in the network in order to find the total energy utilized by the network. Since it can be assumed that energy consumption when transmitting or receiving a packet is greater than zero it can be assumed that more transmissions also lead to more power consumption [16].

The distances between the nodes and the sink are not uniform, which poses a challenge in calculating energy consumption. To overcome this challenge, it is assumed that there are two distance values of relevance: the mean distance from a node to the sink and the mean distance between the nodes and other nodes. The distances will be referred to as long range transmission and short-range transmission respectively.

To provide context in the calculations for the percentage of packet loss, state diagrams are used to describe protocols. LEACH, as shown in Figure 3(a), can either be under attack or not. Because LEACH has no detection capability, it continuously fluctuates between the two states. Black hole attacks drop all packets that they receive so for the fraction of rounds that the malicious node is cluster head a whole cluster worth of packets is lost. During the rounds the malicious node is not cluster head only the packet from the malicious node is dropped.

For ARC-LEACH shown in Figure 3(b), there are three possible states that the network could be in: not under attack, under attack but the malicious node was not detected, and under attack but the malicious node was detected. Like with LEACH when the node is not under attack only the packet from the malicious node is lost, and when under attack and not detected all packets are lost from the cluster. When the network is under

attack and the malicious node is detected, the malicious node is blacklisted and can only drop its own packets.

Alternate routing, in Figure 3(c), has two stages that are like LEACH: under attack and not. The key difference between the states for LEACH and alternate routing is that when alternate routing is under attack it detects the attack shortly after and recovers. Since alternate routing detects the attack and can adapt only the packets from the malicious node are lost for both states.

Figure 4 depicts an example topology where a WSN has four clusters, each with four sensor nodes. The distances in intra clusters are uniformed while the distances across inter clusters (i.e., each CH with BS) are the four times longer. One malicious node is conducting a black hole attack on the network. *Equation 4* specifies the parameters' values.

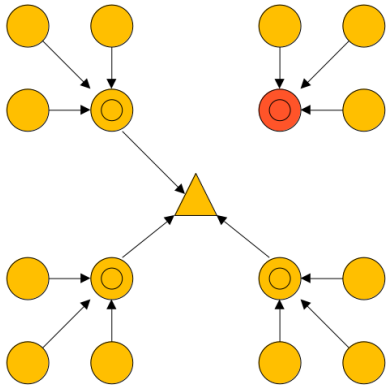


Figure 4: WSN Example Topology

$$t_l = long\ transmission \quad (4)$$
$$t_s = short\ transmission$$
$$c = average\ number\ of\ clusters$$
$$n = average\ number\ of\ nodes\ per\ cluster$$
$$m = number\ of\ malicious\ nodes$$
$$h = average\ number\ of\ hops\ to\ leave\ cluster$$
$$e_s = short\ transmission\ energy$$
$$e_l = long\ transmission\ energy$$
$$e_T = total\ transmission\ energy$$
$$c = 4\ clusters$$
$$n = 4\ nodes\ per\ cluster$$
$$m = 1\ malicious\ node$$
$$h = 1\ hop$$
$$e_l = 4e_s$$
$$e_s = 10kj$$

Using the state diagrams and an understanding of the topologies the percentage of rounds the malicious node is cluster head could be derived. For LEACH the cluster head is malicious a quarter of the time since LEACH does not detect the attack. LEACH not being able to detect the attack allows it to continue to drop packets making the packet loss worse.

The fraction of rounds there is a malicious cluster head for alternate routing and ARC-LEACH is drastically reduced due to both protocols detecting and preventing attacks. For alternate routing it is assumed that the malicious node loses control over the cluster immediately because they detect the attack and find a new node to relay packets too. ARC-LEACH absorbs the first attack to detect the malicious node so the round where an undetected node attacks, so if there is one malicious node it is only affected for one round.

The number of transmissions can be found by counting how many nodes the packet is relayed to before reaching the sink. For LEACH the number of short-range transmissions will be the number of non-malicious nodes in the WSN minus the cluster heads. The number of long-range transmissions is the number of cluster heads in the network excluding any malicious cluster heads.

The number of transmissions for ARC-LEACH is like the number of transmissions for LEACH. The key difference is that when under a blackhole attacks the sink sends an extra blacklist packet to each cluster and then the cluster head relays this blacklist packet to all the nodes in the cluster. This means that you would add another short-range transmission for each node in the network not including the cluster heads. No additional long-range transmissions are added because the transmission is sent from the sink which is considered to have infinite power in comparison to the individual nodes in the cluster and hence should be ignored.

Alternate routing presents additional complexity since the number of transmissions is dependent on the probability of the node finding a new route. This means that each node under attack may have to iterate through many nodes before finding a new path. For the sake of the investigation all calculations were done with the best case in mind. The best case for alternate routing is finding a new route after only a single try. In this case alternate routing has the same number of transmissions as LEACH plus an additional short-range transmission for all non-malicious nodes excluding cluster heads when not under attack due to acknowledgement. When under attack it adds an extra two short range transmissions: one from the acknowledgement when finding new path and one for each attempt to find a new route which is one in this case. If best case was not considered it the number of packets added when under attack could be significantly higher.

The first calculations were used to get the total energy of each protocol. The total energy consumption is calculated in terms of the energy consumption of a short-range transmission as seen in *Equations (5), (6), (8), (9), (11), and (12).* The energy consumption was calculated using the number of short- and long-range transmissions when there is malicious a cluster head and when there is not. In *Equations (7), (10), and (13)* weighted averages over fifty rounds are taken using the percentage of rounds that the malicious node cluster head, when it is not, and the energy consumption during both situations. The result is the average energy consumption over the entire network for fifty rounds.

Energy Consumed with Bad Cluster Head:

$$t_{sb} = (n-1)c = (4-1)4 = 12 \quad (5)$$
$$t_{lb} = c - m = 4 - 1 = 3$$
$$e_b = e_l t_{lb} + e_s t_{sb} = 4e_s t_{lb} + e_s t_{sb} = e_s(4t_{lb} + t_{sb}) = e_s(4(3) + 12) = 24e_s$$

Energy Consumed with Good Cluster Head:

$$t_{sg} = (n-1)c - m = (4-1)4 - 1 = 11 \quad (6)$$
$$t_{lg} = c = 4$$

$$e_g = e_s(4t_{lg} + t_{sg}) = e_s(4(4) + 11) = 27e_s$$

Energy Consumed on Average Over 50 Rounds:

$$e_T = \frac{1}{4}(24e_s) + \frac{3}{4}(27e_s) = 26.25e_s = 262.5kj \quad (7)$$

Energy Consumed with Bad Cluster Head:

$$t_{sb} = (n-1)c + (n-1)c = (n-1)2c \quad (8)$$
$$= (4-1)\big(2(4)\big) = 24$$
$$t_{lb} = c - m = 4 - 1 = 3$$
$$e_b = e_s(4t_{lb} + t_{sb}) = e_s(4(3) + 24) = 36e_s$$

Energy Consumed with Good Cluster Head:

$$t_{sg} = (n-1)c - m = (4-1)4 - 1 = 11 \quad (9)$$
$$t_{lg} = c = 4$$
$$e_g = e_s(4t_{lg} + t_{sg}) = e_s(4(4) + 11) = 27e_s$$

Energy Consumed on Average Over 50 Rounds:

$$e_T = \frac{1}{50}(e_b) + \frac{49}{50}(e_g) = \frac{1}{50}(36e_s) + \frac{49}{50}(27e_s) \quad (10)$$
$$= 27.18e_s = 271.8kj$$

Energy Consumed with Bad Cluster Head:

$$t_{sb} = (n-1)c + (n-1)hm + (n-1)c \quad (11)$$
$$= (n-1)(2c+m) = (4-1)(2(4)+1) = 27$$
$$t_{lb} = c = 4$$
$$e_b = e_s(4t_{lb} + t_{sb}) = e_s(4(4) + 27) = 43e_s$$

Energy Consumed with Good Cluster Head:

$$t_{sg} = (n-1)c - m + (n-1)c = (n-1)(2c) - m \quad (12)$$
$$= (4-1)\big(2(4)\big) - 1 = 23$$
$$t_{lg} = c = 4$$
$$e_g = e_s(4t_{lg} + t_{sg}) = e_s(4(4) + 23) = 39e_s$$

Energy Consumed on Average Over 50 Rounds:

$$e_T = \frac{1}{50}(e_b) + \frac{49}{50}(e_g) = \frac{1}{50}(43e_s) + \frac{49}{50}(39e_s) \quad (13)$$
$$= 39.08e_s = 390.8kj$$

The calculations in *Equations (14)-(16)* are used to find the percentage of packet lost for each protocol on average. These equations use the topology to derive the fraction of time spent in each given state. The percentage of time spent in each state is combined with information on the behavior of each protocol to determine what portion of the network would lose packet during each of the given states. A weighted average is then taken of the percentage of packets lost in each of the states for a given protocol. The result can then be used to compare the expected packet loss of one protocol vs another protocol.

In LEACH protocol if the cluster is being attacked by a given malicious node for the first time, then the entire cluster is lost. That means that the packet loss is the fraction clusters that have a malicious cluster head. When the network is not under attack only the malicious nodes are losing packets, so the packet loss is a fraction of nodes that are malicious. This can all be seen in *Equation (14)*.

Equation (15) calculates the percentage of packets loss in ARC-LEACH. The calculations for ARC-LEACH show similarities to the calculations for LEACH possessing the same packet loss when in each state. The primary difference between ARC-LEACH and LEACH when it comes to packet loss is the time spent in each state. ARC-LEACH spends significantly less time affected by the black hole drastically lowering the packet loss. Due to ARC-LEACH only being affected by the attack when an unknow malicious node attacks the network, that means over the course of all the rounds the number of rounds with a higher packet loss is equal to the number of malicious nodes. As the number of rounds increase and then number of malicious nodes decrease the fraction of time spent in the affected state is reduced improving packet loss.

Alternate routing as shown by *Equation (16)* has the same rate of packet loss no matter the state of the network. This is because alternate routes can recover during the round retransmitting the date to a new safe route as in Max-LEACH [21]. When using alternate routing the only node that loses packets ends up being the malicious node. Since the malicious node is the only node that is losing packets, the packet loss is the percentage of nodes.

LEACH Packet Loss Percentage:

$$l_b = \frac{1}{c} = \frac{1}{4} = 0.25 = 25\% \quad (14)$$
$$l_g = \frac{1}{cn} = \frac{1}{16} = 0.0625 = 6.25\%$$
$$l = \frac{1}{4}\left(\frac{1}{4}\right) + \frac{3}{4}\left(\frac{1}{16}\right) = 0.1094 = 10.94\%$$

ARC-LEACH Packet Loss Percentage:

$$l_b = \frac{1}{c} = \frac{1}{4} = 0.25 = 25\% \quad (15)$$
$$l_g = \frac{1}{cn} = \frac{1}{16} = 0.0625 = 6.25\%$$
$$l = \frac{1}{50}\left(\frac{1}{4}\right) + \frac{49}{50}\left(\frac{1}{16}\right) = 0.06625 = 6.625\%$$

Alternate Routing Packet Loss Percentage:

$$l = \frac{1}{ct} = \frac{1}{16} = 0.0625 = 6.25\% \quad (16)$$

V. Evaluation with Simulation

We use Network Simulator 3 (NS3) [24] to evaluate our proposed ARC-LEACH, compared with LEACH and Alternate Routing. The simulates individual packet transmission through antennas which drains a battery in the node. The packets transmitted contain a size property as well as metadata that represents the information that would be found inside of the packet. The metadata of the packets sent and received are used to manipulate the behavior of each node in the network. The information and timing of packet transmission of individual nodes are then used to build up the topology of each protocol. This allows for a granular simulation that simulates accurate behavior. The structure of the simulation is observed in Figure 5, where the blue indicates NS3 created structures and the yellow indicates custom structures created for the simulation. Our simulation package is posted on GitHub for references:

https://github.com/mvieira4/arc-leach-sim

Figure 5: Simulation Structural Diagram

The simulation was run with 200 rounds and 100 events per round. There was a total of 12 nodes for the simulation each protocol with a 25% chance of a node being cluster head meaning on average there were 4 clusters of 3 at a time. The configuration was run on each protocol with a single malicious node as well as with 3 malicious nodes. This allowed for a comparison between each protocol in power consumption and percentage of packet arrival, as well as how much these values change with more malicious nodes.

The number of malicious nodes was set to 1, the events were set to 50, and the rounds were set to 100. The simulations were then run for each protocol switching between 12 and 24 nodes. This was done to check to see how the performance changed for each protocol if the number of nodes increased. Then, the clusters were made larger to see the impact of larger clusters.

VI. Results Jutification

The results of the simulation with a single malicious node are shown in Figure 6 and Table 1, agreeing with the finding by A.P. Renold et al [16]. When tested against a single malicious node LEACH had resilience against again packets being dropped by a malicious node. As seen on the graph there is an initial decrease caused by the black hole attack. This is followed by a minor increase in the packet arrival percentage stabilizing at 91%. This resilience is due to the rotation of the cluster head limiting the time that the malicious node can cause damage but because it is never mitigated there are continued dips in packet arrival.

Alternate routing performs the best with 94% arrival rate due to the node switching cluster head upon not receiving. Alternate routing starts at 94% and stays at the same value because it can detect and recover from the attack in the same round. This protocol resends the data if it is lost to the blackhole which allows for information to be more reliably transmitted.

ARC-LEACH has a similar performance to alternate routing with the percentage of packets lost stabilizing at 93%. During the simulation ARC-LEACH experienced the attack earlier than LEACH indicated by the arrival rate starting relatively low for the first couple rounds. As was expected, after the initial attack ARC-LEACH recovers then remains at a consistent rate.

Something else that is important to note is the number of rounds that the protocols last. The simulation was. Run for 150 rounds but LEACH died at approximately 135 rounds making it protocol with the least energy consumption tested. Alternate routing dies at around 120 rounds indicating a notable increase in power consumption making the worse protocol tested in this regard. The power consumption is reduced by the extra transmission when transmitting acknowledgment packets as well as the extra packet sent to the malicious node before it is detected. ARC-LEACH, although providing results close to alternate routing was able to achieve power consumption like LEACH. ARC-LEACH has low energy consumption because it can avoid sending packets when it is not under attack and permanently tracks blocked nodes.

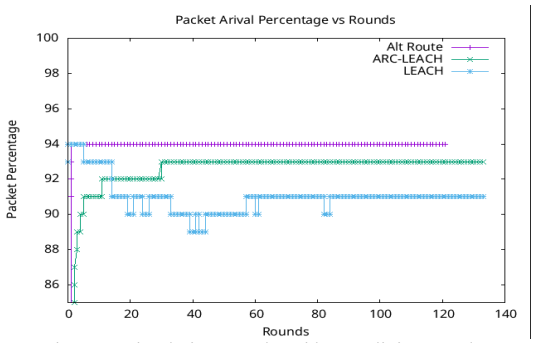


Figure 6: Simulation Results with 1 Malicious Nodes

When running the simulation with three malicious nodes shown in Figure 7 and Table 1, LEACH loses its resilience completely breaking down with packet arrival stabilizing at 67%. This is most likely because the probability of a malicious node causing packet loss of an entire node goes up reducing the effect of just rotating the cluster head. The increased number of attacks can be seen by the increased number of dips when looking at the graph for three malicious nodes vs the graph for one malicious node.

What was surprising was the decline in performance by alternate routing. The decline in performance may be due to multiple malicious nodes becoming cluster head at the same time. This would not only cause a significant decrease in performance sitting at 73% but may also significantly reduce the number of alternative routes that could be taken. If a node is only in the range of two potential new cluster heads and both are malicious it will not be able to avoid both. Multiple malicious cluster heads at the same time would explain how the reduction in performance is so substantial when the attacks are effective.

ARC-LEACH was able to hold up well against the three malicious nodes. The packet arrival was able to reach 80% most likely to ARC-LEACH not depending on finding routes but rather black listing the malicious nodes. Since the gaps still appear when the cluster head rotates the blacklist packet is still able to be sent. If the next node is malicious the blacklist message may not be received but whenever there is a non-malicious cluster head the previous one will be blacklisted. This allows the malicious nodes to be eliminated one by one until no threat remains. In the case of the simulation, it looks like the network was hit hard with an initial attack that kept the packet loss high for an extended period, but the malicious nodes were blacklisted, and the network was able to recover.

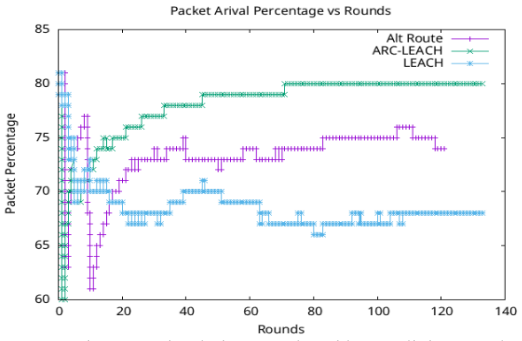


Figure 7: Simulation Results with 3 Malicious Nodes

Table 11: Simulation Results

	LEACH	ARC-LEACH	Alt Route
Last Round	135	134	121
Recv Percentage (1 Mal Node)	91%	93%	94%
Recv Percentage (3 Mal Nodes)	67%	80%	73%
Energy Consumption (J)	6M	6M	6M
Number of Nodes	16	16	16

VII. CINCLUSION AND FUTURE WORK

This investigation shows that the basic LEACH protocol resists a single black hole attack but fails at multiple black hole attacks. ARC-LEACH and Alternate Routing provide stronger protection. Alternate Routing yields the lowest packet loss rate with two costs: higher energy consumption and less resilience against collaborative multiple black hole attacks. Our ARC-LEACH offers the best balance between energy efficiency and security strength, closing the gap in LEACH-security protocols including S-LEACH [9] and MS-LEACH [14], against black hole attacks.

ARC-LEACH needs to be tested for its potential against gray hole attacks. The randomization effort of transmitting and obfuscating meta data by encryption should provide sufficient security, but it has yet to be tested. It may also be worth investing in some variation of ARC-LEACH on retransmitting packet lost after blacklisting malicious nodes and building a trusted WSN [25]. It would allow ARC-LEACH to circumvent packets lost in the first attack at a reasonable cost of energy consumption.

REFERENCES

[1] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," in *33rd Hawaii International Conference on System Sciences*, IEEE, 2000, pp. 1–10.

[2] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans Wirel Commun*, vol. 1, no. 4, pp. 660–670, 2002.

[3] S. K. Singh, P. Kumar, and J. P. Singh, "A Survey on Successors of LEACH Protocol," *IEEE Access*, vol. 5, pp. 4298–4328, 2017, doi: 10.1109/ACCESS.2017.2666082.

[4] Y. Li, N. Yu, W. Zhang, W. Zhao, X. You, and M. Daneshmand, "Enhancing the Performance of LEACH Protocol in Wireless Sensor Networks," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2011, pp. 223–228.

[5] T. M. Rahayu, S.-G. Lee, and H.-J. Lee, "Survey on LEACH-based Security Protocols," in *16th International Conference on Advanced Communication Technology (ICACT)*, IEEE, 2014, pp. 304–309.

[6] A. D. Wood and J. A. Stankovic, "Denial of Service in Sensor Networks," *IEEE Computer*, vol. 35, no. 10, pp. 54–62, 2002.

[7] N. Ahmed, S. S. Kanhere, and S. Jha, "The Holes Problem in Wireless Sensor Networks: A Survey," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 9, no. 2, pp. 4–18, 2005.

[8] K. A. Scarfone, D. Dicoi, M. Sexton, and C. Tibbs, "Guide to securing legacy IEEE 802.11 wireless networks," *NIST SP 800-48r1*. Jul. 2008. doi: 10.6028/NIST.SP.800-48r1.

[9] A. C. Ferreira, M. Aurélio Vilaça, L. B. Oliveira, E. Habib, H. C. Wong, and A. A. Loureiro, "On the Security of Cluster-based Communication Protocols for Wireless Sensor Networks," in *4th IEEE International Conference on Networking (ICN)*, 2005, pp. 449–458.

[10] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, "SPINS: Security Protocols for Sensor Networks," in *7th Annual International Conference on Mobile Computing and Networking (MobiCom)*, ACM, Jul. 2001, pp. 189–199. doi: 10.1145/381677.381696.

[11] L. B. Oliveira, H. C. Wong, M. Bern, R. Dahab UNICAMP, and A. A. F Loureiro UFMG, "SecLEACH - A Random Key Distribution Solution for Securing Clustered Sensor Networks," in *5th IEEE International Symposium on Network Computing and Applications*, 2006, pp. 145–154.

[12] J. Wang, G. Yang, S. Chen, and Y. Sun, "Secure LEACH Routing Protocol based on Low-Power Cluster-Head Selection Algorithm for Wireless Sensor Networks," in *International Symposium on Intelligent Signal Processing and Communication Systems*, IEEE, 2007.

[13] M. A. Abuhelaleh, T. M. Mismar, and A. A. Abuzneid, "Armor-LEACH – Energy Efficient, Secure Wireless Networks Communication," in *17th International Conference on Computer Communications and Networks (ICCCN)*, IEEE, 2008. doi: 10.1109/ICCCN.2008.ECP.142.

[14] M. El_Saadawy and E. Shaaban, "Enhancing S-LEACH Security for Wireless Sensor Networks," in *IEEE International Conference on Electro/Information Technology (EIT)*, 2012.

[15] R. K. Kodali, S. K. Gundabathula, and L. Boppana, "Multi level secure LEACH protocol model using NS-3," in *First International Conference on Networks & Soft Computing (ICNSC)*, IEEE, 2014. doi: 10.1109/CNSC.2014.6906715.

[16] A. P. Renold, R. Poongothai, and R. Parthasarathy, "Performance Analysis of LEACH with Gray Hole Attack in Wireless Sensor Networks," in *International Conference on Computer Communication and Informatics (ICCCI)*, IEEE, 2012.

[17] S. P. Dongare and R. S. Mangrulkar, "Implementing Energy Efficient Technique for Defense against Gray-Hole and Black-Hole Attacks in Wireless Sensor Networks," in *International Conference on Advances in Computer Engineering and Applications (ICACEA)*, 2015, pp. 167–173.

[18] D. Zala, D. Thummar, and B. R. Chandavarkar, "Mitigating Blackhole attack of Underwater Sensor Networks," in *12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, IEEE, 2021. doi: 10.1109/ICCCNT51525.2021.9579473.

[19] C. N. Pushpatode and S. V. Sankpal, "Black Hole Attack Prevention in AODV Routing Protocol," in *Asian Conference on Innovation in Technology (ASIANCON)*, IEEE, Aug. 2021.

[20] N. Arya, U. Singh, and S. Singh, "Detecting and Avoiding of Worm Hole Attack and Collaborative Blackhole attack on MANET using Trusted AODV Routing Algorithm," in *International Conference on Computer, Communication and Control (IC4)*, IEEE, 2015.

[21] S. Sabeti and K. Shyamala, "Detection of Black Hole Attack on Max LEACH Protocol," in *International Mobile and Embedded Technology Conference (MECON)*, IEEE, 2022, pp. 199–204.

[22] O. Younis and S. Fahmy, "Distributed Clustering in Ad-hoc Sensor Networks: A Hybrid, Energy-Efficient Approach," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2004.

[23] C.-H. Lung and C. Zhou, "Using Hierarchical Agglomerative Clustering in Wireless Sensor Networks: An Energy-Efficient and Flexible Approach," in *IEEE GlobeCom*, 2008, pp. 1–5.

[24] NSnam.org, "NS-3 Documentation," *a discrete-event network simulator for internet systems*. https://www.nsnam.org/docs/release/3.36/doxygen/index.html (accessed Apr. 14, 2023).

[25] H. Hu, Y. Han, M. Yao, and X. Song, "Trust Based Secure and Energy Efficient Routing Protocol for Wireless Sensor Networks," *IEEE Access*, vol. 10, pp. 10585–10596, 2022, doi: 10.1109/ACCESS.2021.3075959.