

# Vulnerability Analysis of Power-Aware Schemes in Sensor Networks: LEACH and PEGASIS

Akshay Aggarwal - Guillermo Marro  
aggarwal , marro@cs.ucdavis.edu  
Department of Computer Science  
University of California at Davis  
ECS289I - Project Paper

## Abstract

Sensor networks are special types of ad-hoc wireless networks. They are generally used to collect some field data from a given region. The nodes have limited computing capabilities, reduced size and significant power constraints. Therefore extending the lifetime of these nodes is of primary importance. LEACH and PEGASIS are two well-known power-aware routing schemes for sensor networks. These protocols are susceptible to various forms of denial of service attacks due to vulnerabilities highlighted in the paper. The vulnerabilities in the protocols exist at various layers of the protocol stack. The paper presents an analysis of these vulnerabilities and proposes mitigation strategies to alleviate the consequences of such attacks. Additionally the complexity of implementing such countermeasures is discussed.

## 1 Introduction

Sensor networks are comprised of a conglomeration of frequently diminute, power-constrained, self-organized sensor nodes that can be loosely deployed in a variety of environments to collect field data and transmit it to a base station for further processing. Most frequently the goals of this type of networks are to be able to be deployed without perturbing the magnitude being measured (sometimes to avoid being detected by the enemy, in the case of military applications), to collect accurate information of the environment, to be resilient to individual node failures, and to have a long service lifetime. Among the most significant design constraints on the nodes are power limitations, reduced computing capabilities, restricted size and physical robustness. Some very smart schemes have been devised [1, 2] to address both goals and constraints, that concentrate on the energy efficiency of the design.

The purpose of this work is to explore the security aspects of the most promising power-conserving architectures for sensor networks, from a perspective of denial of service attacks (*DoS*, [5, 6]) aimed at accelerating the rate of battery exhaustion in the nodes (i.e. significantly shortening the longevity of the network, [4]).

In section 2 we provide a high level overview of both protocols LEACH and PEGASIS. In section 3 we propose some DoS attacks, and when possible we discuss some countermeasures and their feasibility. Finally in section 4 we present the conclusions.

## 2 Background

The security multidimensional space as approached by analysts and researchers is defined by orthogonal dimensions: *confidentiality*, *integrity* and *availability*<sup>1</sup>. The most relevant premises on sensor networks are to be able to collect meaningful information out of all the data exchanged by the nodes, even in harsh environmental conditions (military battlefield, natural disaster zones, industrial facilities with substantial *EMI*, etc), and to extend the service lifetime of the network as long as possible. Due to the very nature of these networks (untethered medium, power and computationally constrained nodes), *availability* plays a very important role in the security assessment of a certain architecture<sup>2</sup>. Some important research efforts have focused on some clever *DoS* attacks to this type of networks<sup>3</sup> [3].

Two of the most renowned power-aware architectures are: Low Energy Adaptive Clustering Hierarchy (LEACH, [1]) and Power Efficient Gathering in Sensor Information Systems (PEGASIS, [2]). In both schemes, nodes smartly handle communication with the base station with energy-conserving premises.

As a consequence of optimizing the energy dissipated by a node in communicating with others, some potential threats arise that might be exploited by malicious nodes to exhaust battery resources of the legitimate nodes with a faster rate than normal use would.

### 2.1 LEACH

LEACH (Low-Energy Adaptive Clustering Hierarchy) is a clustering-based protocol for sensor networks and is described in [1]. LEACH has a localized coordination and set up control for setting up clusters. The *cluster heads* are elected in a randomized manner. This process is completely distributed and does not require any control input from the base station. The nodes act in a localized manner and do not need any global information in order for them to operate. LEACH aims at substantially increasing the service lifetime of a given sensor network. The main energy saving comes from the aggregation of data at the cluster heads before forwarding to the base station. It is more energy-efficient than *direct communication* and *minimum transmission energy* (MTE) protocols. LEACH assumes that all the nodes are homogeneous and energy constrained. The base station is also fixed and located far from the nodes. Each node is within radio distance of the base station.

LEACH has a set-up phase, when clusters are formed and the steady-state phase, when data transmission occurs. In the set-up phase first comes the advertisement phase. In this phase every node chooses a random number. If this random number is less than a calculated threshold then the node becomes a cluster head, else it does not. Once a node becomes a cluster head it cannot become a cluster head for a certain number of rounds. The threshold value depends upon the percentage of nodes wanted as cluster heads and the number of rounds elapsed. The percentage of nodes as cluster heads is a very important parameter. There is an optimum percentage for which

---

<sup>1</sup>Often times other dimensions are considered in the analysis as well: *access control* and *non repudiation*.

<sup>2</sup>However depending on the type of application, other aspects might be treated as priorities. For instance in military battlefields, *confidentiality* is certainly a paramount.

<sup>3</sup>*DoS* attacks focus at disrupting services (or at very least at degrading *QoS*) by exhausting resources of the target node.

the energy dissipation over the network is minimum. The energy curve steeply rises for a lesser percentage. The energy dissipation curve's rise for a higher cluster-head percentage is relatively slower. Each non-cluster head node decides which cluster it will become part of on the basis of the signal strength of the cluster advertisement. The cluster heads use CSMA MAC for the cluster advertisement. Next comes the cluster set-up phase where each node informs its cluster head about its intention to join the cluster. The nodes use CSMA MAC for this. Then the cluster head creates and broadcasts a TDMA schedule which informs each node about the time slot it should transmit in.

In the steady-state phase the nodes transmit to the cluster head when their slot in the TDMA schedule arrives. The nodes can turn themselves off and sleep while waiting for their slot. This is one of the major energy-saving features of LEACH. Once the cluster head receives data from all the nodes it uses an aggregating algorithm to compress the data and then sends this compressed data stream to the base station. This is the most significant energy-saving feature of LEACH.

After a certain time frame the cluster set-up is repeated with new nodes becoming cluster-heads in a random manner. This randomisation helps in keeping the energy dissipation in the system low and also uniformly degrades the energy for all the nodes.

## 2.2 PEGASIS

In PEGASIS nodes communicate only with neighbors by forming a chain involving all the nodes in the topology, and only one of them per round (the node designated as *leader*) collects a digest of all exchanged data between nodes and relays it to the base station. Since forming an optimal chain is equivalent to solving the *traveling salesman* problem, which is known to be hard, a sub-optimal approach characterized by a *greedy algorithm* is used to encompass all nodes in the network. The leader node is chosen randomly once per round among all nodes in the chain. In this manner, energy consumption is equally distributed on average, since the most power consuming operation for these nodes is precisely communicate with the base station which is assumed to be at a significant distance from the rest of the nodes.

In sensor networks, a data compression technique [1] known as *data fusion* is used to reduce the amount of data transmitted between sensor nodes and the base station. Since the nodes use wireless communications to exchange information, transmission and reception of packets are expensive operations, particularly when the signal has to be strong enough to travel a significant distance. In PEGASIS, energy efficiency is guaranteed by two characteristics of the protocol: only one node communicates at a time with the base station, and the rest of the nodes communicate locally only with neighbors. Due to these design considerations, PEGASIS reportedly [2] achieves between 100 to 300% improvement in energy efficiency when compared to LEACH on a particular setup <sup>4</sup>, by eliminating the overhead of dynamic cluster formation. Another optimization present in PEGASIS is that a threshold is set to disallow nodes with only distant neighbors from becoming *leaders*, to compensate for the greater energy they normally spend each round as components of the chain.

All nodes are assumed to have global knowledge of the network <sup>5</sup>. The chain forming process starts with the farthest node from the base station, to ensure that all remote nodes are included in

---

<sup>4</sup>Most probably those percentages vary for different setups.

<sup>5</sup>In practice this might be achieved with GPS assistance.

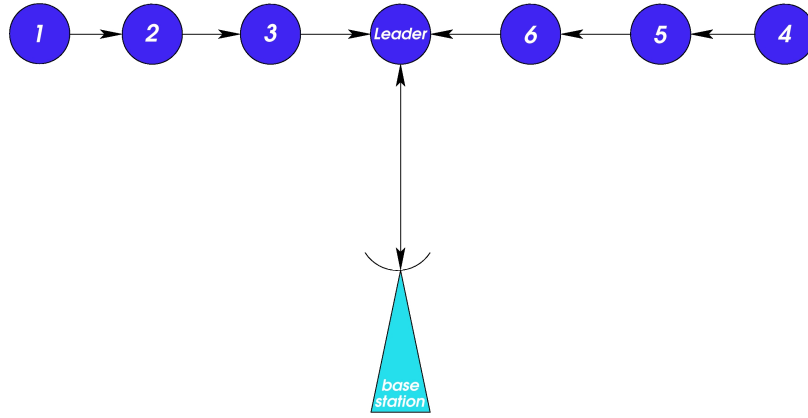


Figure 1: Token-passing communication in PEGASIS

the chain [2]. Whenever a node dies, neighbor nodes reconfigure the chain to avoid it. Given that random choice of the *leader* evenly distributes the energy consumed by all the nodes, the pattern of dying nodes will also be randomly distributed along the chain, and hence resiliency is achieved.

To coordinate data transmission, a simple token-passing mechanism is implemented by the *leader* (fig.1). Each node receives and transmits one packet (by virtue of data fusion)<sup>6</sup>. Therefore the data gathering process in PEGASIS is considerably more energy-efficient than in previously proposed schemes.

### 3 Attacks

#### 3.1 LEACH

We have found several design level flaws in LEACH which make it vulnerable to a number of attacks aiming towards Denial of Service of the entire network or certain portions of it. The vulnerabilities exist in different layers of the network architecture which makes it susceptible to different DoS attacks. We also propose solutions or mitigation strategies for some of these vulnerabilities.

The attacker needs to have varied capabilities to carry out these attacks. In our scenarios we will assume an intelligent attacker. He or she may be able to physically damage or subvert nodes as physical access control to sensor network nodes may not be feasible. The attacker may also be able to deploy nodes which are much more powerful and not subject to the same constraints as the sensor nodes. This may be true in a scenario like battlefield deployment behind enemy lines, where the enemy has an already established powerful wired network.

##### 3.1.1 Brute-force Jamming attack

Like all wireless networks, sensor networks are susceptible to jamming by an attacker. In large networks the attacker will have the ability to jam a portion of the network (jamming all the nodes

<sup>6</sup>With the notable exception of the end nodes that only transmit a packet, and the *leader* that receives two packets and transmit one to the base station per round

may not be feasible). Jamming can be detected by a constant energy signal on a certain frequency band and not by the absence of protocol response messages that cause failure of communication [4]. The jamming can be persistent, resulting in total lack of response and data exchange or sporadic in nature which can also lead to the same effect, if performed with some intelligence. The latter would allow an attacker to efficiently use his or her energy supplies. The normal mitigation strategy against jamming of the physical medium is the use of *spread-spectrum* technology. This will only make it more difficult for the attacker to carry out the attack, because he or she will need to allocate more resources to it. A powerful attacker can however jam the entire frequency range that the protocol utilizes. The use of out of band communication media is a probable solution but it may not be feasible for sensor networks.

### 3.1.2 Spoofed Cluster Head

During the formation of a cluster, each node can become a cluster head with some probability. The non-cluster heads then decide the cluster they belong to depending upon the received signal strength of the cluster head advertisement. An attacker can introduce a malicious node having high power which will always choose to be the cluster head and will also have a signal strength which is substantially higher than other cluster head candidates. This will let the malicious node become the cluster head for a large part of the network (if not all of it). As a cluster head the node may choose to drop all packets or be neglectful, by choosing to drop random packets. The cluster head would also have the capability to induce a TDMA schedule which will maliciously cause collisions between transmissions by different nodes. The attack would use a physical layer capability to subvert the routing capabilities of the network. The attack can be detected if the non-cluster head nodes remember the long-term frequency with which nodes become cluster heads and do not join a cluster of a node that advertises itself with a greater frequency than a specified threshold. The countermeasure itself may be defeated if the attacker has several such malicious nodes which randomly decide to become cluster heads and thus individually remain under the threshold. There is also an additional overhead of remembering the identity of past cluster heads at each node. Another possible strategy that the attacker might use is to have the spoofed cluster head to be mobile and identify itself differently each round to defeat the frequency threshold countermeasure proposed.

### 3.1.3 Supported Cluster Head

An attacker can deploy radio repeaters in the region of the sensor network and use these to amplify cluster advertisement from nodes it wants elected as cluster-heads. The non-cluster heads would be deceived into joining clusters whose heads are not the closests to them. LEACH is unclear about how this would affect the protocol. Two situations arise from this scenario. In the first, the non-cluster head nodes would boost their transmitting power to reach the head. This would force them to consume more power. In the second case, the nodes would be unable to reach the head and would transmit regardless of not being heard. This would cause loss of data. It would also increase the size of a cluster and therefore the TDMA schedule would also be longer for a cluster, and thus the duty cycle of the nodes would be smaller. Both cases lead to a denial of service attack, one by power draining and the other by data loss. The attackers ability to deploy repeaters can also be used to disrupt the random behavior of electing cluster heads in LEACH. A mitigation strategy against this attack would be to remember a profile of signal strength of every node that wants to become a cluster head and look for anomalies. This would obviously impose more memory

storage space and computation on the nodes, but it would build a web of awareness that if correctly implemented would probably increase the robustness of the scheme.

#### 3.1.4 Ghost Nodes

The performance of LEACH is highly dependent on the suggested percentage of head nodes. Deviation from this percentage leads to more energy dissipation. In networks where the threshold level is not preset before deployment or networks which are periodically replenished a malicious node can advertise itself as  $k$  nodes and thus increase the total number of nodes in the system. This will cause the number of heads needed to satisfy the optimum percentage to increase and thus the energy dissipation of the system will increase. A malicious node can also assume the role of cluster head more often by assuming all ghost identities without appearing to be anomalous.

#### 3.1.5 Inhibiting Node Discovery

In some sensor networks, initially deployed nodes have to discover each other when the network is first activated. An attacker could inhibit the discovery of some nodes by jamming or by forcing them to remain silent<sup>7</sup>. This would lead the other nodes in the network to believe that there are a lesser number of nodes in the network. In other words, to depopulate the network by reducing sensor density. A fewer number of head nodes would be needed to satisfy the head node percentage. Simulation models [1] have shown that having fewer than the optimum number of nodes leads to more energy dissipation than by having a greater number of head nodes. The effect of this attack would be to increase the power consumption of the network greatly. A solution to this attack would be to have each node insure that at least  $n$  nodes have discovered it and acknowledged its presence. An additional overhead would have to be incurred at the time of network discovery.

#### 3.1.6 Forged Base Station

The attacker may cause a DoS attack by impersonating a base station. This would be possible if the attacker breaks the authentication mechanism for the base station or if no authentication mechanism is used. The attacker would place a malicious base station with a stronger signal than the original base station. This would cause the cluster heads to transmit at a lower power and the data would not reach the intended base station. The attack is only possible if the nodes are not preprogrammed with the position of the base station. The fake base station may choose to drop all data, a random part of it or just alter the data.

#### 3.1.7 Neighbor Interference

In LEACH, transmissions from different clusters will disturb each other. To minimize this interference each cluster uses different CDMA codes. A malicious cluster head may determine the CDMA code used by neighboring cluster heads and adopt the same spreading code or use a code that is invalid and will interfere with the other nodes. This would cause the network in that region

---

<sup>7</sup>This could be achieved by either physical damage, or electromagnetic shielding or logical hacking into them.

to be lossy and unreliable. This would be an attack that could be used in order to suppress information from a particular region of the network without the specific use of jamming signals.

### 3.1.8 Cluster Set-Up Channel Blocking

During cluster formation each node has to inform the cluster head about its intention of joining its cluster. This information is transmitted using a CDMA MAC protocol. A malicious node may block access to this channel by continuously sending on the channel. Other nodes will then start a back-off stage and will not be able to join a cluster. This would lead to a breakdown in the routing mechanism of LEACH and also drain the power of the nodes as they would have to sense the channel repeatedly.

## 3.2 Location aware modifications to LEACH

Variations to LEACH have been proposed and implemented in network simulators where the nodes are aware of the network topology and the location of each node in the system. This changes the assumptions that were introduced in the original paper [1]. Some of the attacks identified earlier are not possible in this scenario as they exploit the inability of nodes to determine the location of neighboring nodes to execute the attack.

The network is still vulnerable to brute-force jamming attacks as these are attacks against the very nature of the physical medium of communication, the radio waves, and are independent of the protocol. All wireless protocols are susceptible to these attacks. These attacks can be identified easily but no comprehensive solution is available.

The spoofed cluster head attack is also still possible because the nodes will still join the cluster head which has the maximum signal strength. The attacker will have to subvert a node in order to gain control over it. The use of an amplifier may not be detected by the other nodes. However, the nodes will be aware of the distance of the head nodes from them and as such will have the ability to detect an abnormally high signal strength from a node that is far away. The nodes therefore could choose to avoid joining such clusters.

Cluster heads could still be supported by the use of radio repeaters. This would not require the attacker to subvert any nodes. Anomaly detection in signal strength could again be used as a strategy for detecting the attack and response could be initiated.

Introduction of ghost nodes will only be possible if the nodes are not preprogrammed with the location of other nodes prior to deployment (an unrealistic assumption anyways). In case of the network using a node discovery algorithm to discover nodes then ghost nodes could be introduced by malicious nodes. Also in scenarios where there is replenishment of nodes, ghost nodes could be introduced in the replenishment phase. With the use of a node discovery algorithm the system would still remain vulnerable to the inhibition of the node discovery phase.

Once a malicious node becomes a cluster head it will still retain its ability to degrade the performance of the network by causing interference in the communication of its neighboring cluster by using duplicate or erroneous CDMA spreading codes. This attack will not require the adversary

to have the ability of introducing a radio jammer which might be easily detectable.

A malicious node will still be able to subvert the cluster set-up phase which uses CDMA MAC for communication. It can continuously broadcast on a channel and make it impossible for the cluster to be set up.

Forging of the base station will not be a probable attack in this scenario as the base station would remain static and its location preprogrammed into all nodes. The nodes can then determine the signal strength needed to reach the base station.

### **3.3 PEGASIS**

Many of the conceptual attacks described for LEACH are also effective with PEGASIS. The difference is that in PEGASIS attacks attempt to subvert the chaining mechanism and the relay of information from the leader to the base station, whereas in LEACH attacks try to break either cluster communication or cluster head to base station connectivity.

Given that only a high-level overview of PEGASIS is presented in the paper, attacks presented here are only described from a conceptual standpoint, without providing implementation details.

#### **3.3.1 Brute-force Jamming attack**

Once again, a jamming signal can be used to disrupt communication among sensor nodes. Schemes such as FHSS (Frequency Hopping Spread Spectrum) can only mitigate the problem and are still vulnerable to a broadband jamming signal generator. Sophisticated attacks can be devised in such a way that the jamming signal only affect the relaying of information from the leader to the base station. In this way, the energy necessary from all the nodes in the chain to relay information to the leader is wasted and yet the purpose of communicating meaningful data to the base station is defeated. Moreover, since the storage capacity in the nodes is limited, with only relatively a small amount of signal energy the attacker can successfully jam the leader.

#### **3.3.2 Attacks on the leader**

Although the paper describing PEGASIS does not provide details about the leader election other than highlighting its alternate random location along the chain, attacks concentrated on biasing the election of the designated node towards the ends of the chain (or farthest points from the base station) might be feasible. This in turn would presumably force the leader nodes to spend more energy on the average since the signal has to travel longer distances. A possible way to mitigate this attack would be to enforce a mechanism by means of which the probability of being chosen as leader is inversely proportional to the power reserves of the nodes. Clearly this mechanism has some trust issues not necessarily easy to solve, such as compromised nodes not revealing their true power reserve value, and the need to maintain a robust power-state polling mechanism. Unfortunately in sensor networks the amount of cryptographic primitives that might be used is truly limited (due to the computationally constrained nature of the nodes) and therefore distributed applications involving trust are a hard problem.

Yet another possible attack is to compromise the leader on each round in such a way that the data fused from the chain does not get relayed to the base station. Luckily the practical implementation of this sort of attack is far from being trivial, since it implies guessing the pseudo-randomness of the leader election mechanism. A robust implementation of the pseudo-random algorithm (strong seed generation and software-based solutions) makes the practical implementation of this attack almost impossible to achieve.

### 3.3.3 Nodes Inhibition

By intermittently inhibiting nodes<sup>8</sup>, the chain reconstruction mechanism by means of which PEGASIS bypass non-responding nodes<sup>9</sup> is forced to kick in periodically. Since the instability in the chain topology causes the system to go into neighbor discovery phase which is an energy-consuming operation, the total power reserve of the system is seriously affected. Just as mentioned in the previous paragraph, avoiding this attack implies to have a protocol that reliably sense power reserves in the nodes, which is certainly challenging to implement.

### 3.3.4 Forged Base Station

Just as in LEACH, the leader might be tricked into trusting the forged base station as the legitimate one, provided that the forged base station can negotiate its role with the leader before the other. It is not clear from the paper what the negotiation involves, but whatever the handshake for mutual pseudo-authentication is it can be forged as well (given that as mentioned before, cryptographic primitives are most probably not used) with a stateful implementation of the base station role.

### 3.3.5 Information Corruption

Compromised nodes can tamper with information transmitted along the chain without any node being able to detect it. From the perspective of application data, the chain still constitutes a single point of failure sort of mechanism, because a single malicious node in the chain can compromise the integrity of the information collected by the base station. A decent solution to this problem involves cryptographic primitives and certification authorities for key management, which is an unrealistic set of requirements in sensor networks for reasons previously discussed.

## 4 Conclusions

Sensor networks have proven essential in data collection scenarios where manual deployment of traditional wired sensors is unfeasible. Often times these networks have to operate in harsh environmental conditions. Although these networks are designed to be resilient to such conditions, to self-organize, to be energy-efficient and infrastructureless and to overcome node failures, current routing schemes are vulnerable to a variety of denial of service attacks aimed at exhausting power reserves and hence shortening service lifetime. Routing algorithm designers have striven to extend

---

<sup>8</sup>As mentioned before this might be achieved with electromagnetic shielding or logical hacks into the nodes.

<sup>9</sup>Nodes incrementally increase the signal strength until they receive an acknowledge from the closest neighbors.

battery autonomy of nodes but have not kept an eye on security. There is an urgent need to review the design of these protocols to make them withstand potentially intelligent adversaries. A comprehensive, layered scheme of countermeasures seems to be the best approach to cope with these threats. At the physical layer the most robust technology seems to be FHSS. In upper layers, introducing lightweight authentication among nodes together with some distributed mechanism to maintain global state through a redundant web of trust (nodes monitoring their neighbors) might help mitigate the impact of the attacks presented in this paper, even though as a trade-off these measures might adversely affect the optimal network lifetime achieved in the original papers.

## References

- [1] W. Heinzelman, A. Chandrakasan and H. Balakrishnan. **Energy-Efficient Communication Protocol for Wireless Microsensor Networks**. Proceedings of the 33rd Hawaii International Conference on System Sciences (HICSS '00), January 2000.
- [2] S. Lindsey, C. S. Raghavendra. **PEGASIS: Power Efficient GATHERing in Sensor Information Systems**. IEEE Aerospace Conference, March 2002.
- [3] F. Stajano, R. Anderson. **The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks**. 7th International Workshop on Security Protocols, Cambridge, UK, April 1999.
- [4] A Wood, J Stankovic. **Denial of Service in Sensor Networks**. IEEE Computer magazine, October 2002.
- [5] Needham R. **Denial of Service: an example**. Communications of the ACM - Volume 37 , Issue 11 - November 1994.
- [6] Schuba C., Krsul I., Kuhn M., Spafford E., Sundaram A., and Zamboni D.. **Analysis of a Denial of Service Attack on TCP**. Proceedings of the 1997 IEEE Symposium on Security and Privacy - pp. 208-223 - May 1997.