A Review of Wireless Sensor Networks: Attacks and Countermeasures

Srikanth Narayanaraju, Dr Syed Umar, Rajendra Kumar

Department of ECM,
KL University, A.P. INDIA.

***Abstract*— We consider routing security in wireless sensor networks. Many sensor network routing protocols have been proposed, but none of them have been designed with security as a goal. We propose security goals for routing in sensor networks, show how attacks against ad-hoc and peer-to-peer networks can be adapted into powerful attacks against sensor networks, introduce two classes of novel attacks against sensor networks — sinkholes and HELLO floods, and analyze the security of all the major sensor network routing protocols. We describe crippling attacks against all of them and suggest countermeasures and design considerations. This is the first such analysis of secure routing in sensor networks.**

I. INTRODUCTION

Our focus is on routing security in wireless sensor networks. Current proposals for routing protocols in sensor networks optimize for the limited capabilities of the nodes and the application specific nature of the networks, but do not consider security. Although these protocols have not been designed with security as a goal, we feel it is important to analyze their security properties. When the defender has the liabilities of insecure wireless communication, limited node capabilities, and possible insider threats, and the adversaries can use powerful laptops with high energy and long range communication to attack the network, designing a secure routing protocol is non-trivial. We present crippling attacks against all the major routing protocols for sensor networks. Because these protocols have not been designed with security as a goal, it is unsurprising they are all insecure. However, this is non-trivial to fix: it is unlikely a sensor network routing protocol can be made secure by incorporating security mechanisms after design has completed. Our assertion is that sensor network routing protocols must be designed with security in mind, and this is the only effective solution for secure routing in sensor networks. We make five main contributions.

- We propose threat models and security goals for secure routing in wireless sensor networks.
- We introduce two novel classes of previously undocumented attacks against sensor networks1 – sinkhole attacks and HELLO floods.
- We show, for the first time, how attacks against ad-hoc wireless networks and peer-to-peer networks [1], [2] can be adapted into powerful attacks against sensor networks.
- We present the first detailed security analysis of all the major routing protocols and energy conserving topology maintenance algorithms for sensor networks. We describe practical attacks against all of them that would defeat any reasonable security goals.
- We discuss countermeasures and design considerations for secure routing protocols in sensor networks.

II. BACKGROUND

We use the term *sensor network* to refer to a heterogeneous system combining tiny sensors and actuators with general purpose computing elements. Sensor networks may consist of hundreds or thousands of low-power, low-cost nodes, possibly mobile but more likely at fixed locations, deployed en masse to monitor and affect the environment. For the remainder of this paper we assume that all nodes' locations are fixed for the duration of their lifetime. For concreteness, we target the Berkeley TinyOS sensor platform in our work. Because this environment is so radically different from any we had previously encountered, we feel it is instructive to give some background on the capabilities of the Berkeley TinyOS platform. A representative example is the Mica *mote*2, a small (several cubic inch) sensor/actuator unit with a CPU, power source, radio, and several optional sensing elements. The processor is a 4 MHz 8-bit Atmel ATMEGA103 CPU with 128 KB of instruction memory, 4 KB of RAM for data, and 512 KB of flash memory. The CPU consumes 5.5 mA (at 3 volts) when active, and two orders of magnitude less power when sleeping. The radio is a 916 MHz low-power radio from RFM, delivering up to 40 Kbps bandwidth on a single shared channel and with a range of up to a few dozen meters or so. The RFM radio consumes 4.8 mA (at 3 volts) in receive mode, up to 12 mA in transmit mode, and 5 μA in sleep mode. An optional sensor board allows mounting of a temperature sensor, magnetometer, accelerometer, microphone, sounder, and other sensing elements. The whole device is powered by two AA batteries, which provide approximately 2850 mA hours at 3 volts. Sensor networks often have one or more points of centralized control called *base stations*. A base station is typically a gateway to another network, a powerful data processing or storage center, or an access point for human interface. They can be used as a nexus to disseminate control information into the network or extract data from it. In some previous work on sensor network routing protocols, base stations have also been referred to as *sinks*. Base stations are typically many orders of magnitude more powerful than sensor nodes. They might have workstation or laptop class processors, memory, and storage, AC power, and high bandwidth links for communication amongst themselves.

Protocol	Relevant attacks
TinyOS beaconing	Bogus routing information, selective forwarding, sinkholes, Sybil, wormholes, HELLO floods
Directed diffusion and its multipath variant	Bogus routing information, selective forwarding, sinkholes, Sybil, wormholes, HELLO floods
Geographic routing (GPSR, GEAR)	Bogus routing information, selective forwarding, Sybil
Minimum cost forwarding	Bogus routing information, selective forwarding, sinkholes, wormholes, HELLO floods
Clustering based protocols (LEACH, TEEN, PEGASIS)	Selective forwarding, HELLO floods
Rumor routing	Bogus routing information, selective forwarding, sinkholes, Sybil, wormholes
Energy conserving topology maintenance (SPAN, GAF, CEC, AFECA)	Bogus routing information, Sybil, HELLO floods

Fig. 1. Summary of attacks against proposed sensor networks routing protocols.

However, sensors are constrained to use lower-power, lower-bandwidth, shorter-range radios, and so it is envisioned that the sensor nodes would form a multi-hop wireless network to allow sensors to communicate to the nearest base station. See Figure 3 for a picture illustrating a representative architecture for sensor networks. A base station might request a steady stream of data, such as a sensor reading every second, from nodes able to satisfy a query. We refer to such a stream as a *data flow* and to the nodes sending the data as *sources*. In order to reduce the total number of messages sent and thus save energy, sensor readings from multiple nodes may be processed at one of many possible *aggregation points*. An aggregation point collects sensor readings from surrounding nodes and forwards a single message representing an aggregate of the values. Aggregation points are typically regular sensor nodes, and their selection is not necessarily static. Aggregation points could be chosen dynamically for each query or event, for example. It is also possible that every node in the network functions as an aggregation point, delaying transmission of an outgoing message until a sufficient number of incoming messages have been received and aggregated. Power management in sensor networks is critical. At full power, the Berkeley Mica mote can run for only two weeks or so before exhausting its batteries. Consequently, if we want sensor networks to last for years, it is crucial that they run at around a 1% duty cycle (or less). Similarly, since the power consumption of the radio is three orders of magnitude higher when transmitting or listening than when in sleep mode, it is crucial to keep the radio in sleep mode the overwhelming majority of the time. It is clear that we must discard many preconceptions about network security: sensor networks differ from other distributed systems in important ways. The resource-starved nature of sensor networks poses great challenges for security. These devices have very little computational power: public-key cryptography
is so expensive as to be unusable, and even fast symmetric-key ciphers must be used sparingly. With only 4 KB of RAM, memory is a resource that must be husbanded carefully, so our security protocols cannot maintain much state. Also, communication bandwidth is extremely dear: each bit transmitted consumes about as much power as executing 800–
1000 instructions [3], and as a consequence, any message expansion caused by security mechanisms comes at significant cost. Power is the scarcest resource of all: each milliamp consumed is one milliamp closer to death, and as a result, nearly every aspect of sensor networks must be designed with power in mind. Lest the reader think that these barriers may disappear in the future, we point out that it seems unlikely that Moore's law will help in the foreseeable future. Because one of the most important factors determining the value of a sensor network comes from how many sensors can be deployed, it seems likely there will be strong pressure to develop ever-cheaper sensor nodes. In other words, we expect that users will want to ride the Moore's law curve down towards ever-cheaper systems at a fixed performance point, rather than holding price constant and improving performance over time. This leaves us with a very demanding environment. How can security possibly be provided under such tight constraints? Yet security is critical. With sensor networks being envisioned for use in critical applications such as building monitoring, burglar alarms, and emergency response, with the attendant lack of physical security for hundreds of exposed devices, and with the use of wireless links for communications, these networks are at risk.

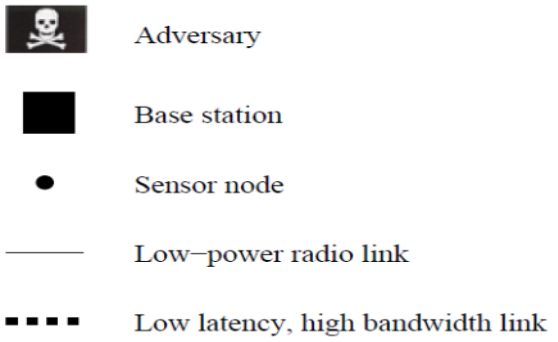

Fig. 2. **Sensor network legend.** All nodes may use low power radio links, but only laptop-class adversaries and base stations can use low latency, high
Bandwidth links.

III. SENSOR NETWORKS VS. AD-HOC WIRELESS NETWORKS

Wireless sensor networks share similarities with ad-hoc wireless networks. The dominant communication method in both is multi-hop networking, but several important distinctions can be drawn between the two. Ad-hoc networks typically support routing between any pair of nodes [4], [5], [6], [7], whereas sensor networks have a more specialized communication pattern. Most traffic in sensor networks can be classified into one of three categories:

1) Many-to-one: Multiple sensor nodes send sensor readings to a base station or aggregation point in the network.

2) One-to-many: A single node (typically a base station) multicasts or floods a query or control information to several sensor nodes.

3) Local communication: Neighboring nodes send localized messages to discover and coordinate with each other. A node may broadcast messages intended to be received by all neighboring nodes or unicast messages intended for a only single neighbor3.

Nodes in ad-hoc networks have generally been considered to have limited resources, but as we have seen in Section II, sensor nodes are even more constrained. Of all of the resource
Constraints, limited energy is the most pressing. After deployment, many sensor networks are designed to be unattended for long periods and battery recharging or replacement may be infeasible or impossible. Nodes in sensor networks often exhibit trust relationships beyond those that are typically found in ad-hoc networks. Neighboring nodes in sensor networks often witness the same or correlated environmental events. If each node sends a packet to the base station in response, precious energy and bandwidth are wasted. To prune these redundant messages to reduce traffic and save energy, sensor networks require in-network processing, aggregation, and duplicate elimination. This often

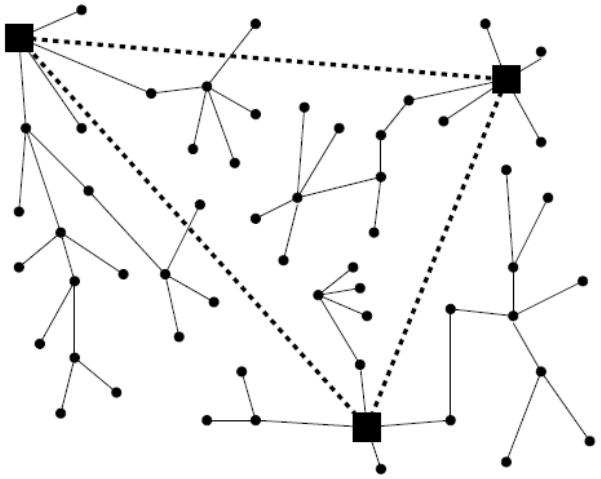

Fig. 3. A representative sensor network architecture.

Necessitates trust relationships between nodes that are not typically assumed in ad-hoc networks.

IV. RELATED WORK

Security issues in ad-hoc networks are similar to those in sensor networks and have been well enumerated in the literature [8], [9], but the defense mechanisms developed for ad-hoc networks are not directly applicable to sensor networks. There are several reasons for why this is so, but they all relate to the differences between sensor and ad-hoc networks enumerated in the previous section.

Some ad-hoc network security mechanisms for authentication and secure routing protocols are based on public key cryptography [8], [10], [11], [12], [13], [14], [15], [16]. Public key cryptography is too expensive for sensor nodes. Security protocols for sensors networks must rely exclusively on efficient symmetric key cryptography. Secure routing protocols for ad-hoc networks based on symmetric key cryptography have been proposed [17], [18], [19], [20]. These protocols are based on source routing or distance vector protocols and are unsuitable for sensor networks. They are too expensive in terms of node state and packet overhead and are designed to find and establish routes between *any* pair of nodes—a mode of communication not prevalent in sensor networks.

Marti et al. [21] and Buchegger and Boudec [22] consider the problem of minimizing the effect of misbehaving or selfish nodes on routing through punishment, reporting, and holding grudges. These application of these techniques to sensor networks is promising, but these protocols are vulnerable to blackmailers. Perrig et al. [23] present two building block security protocols optimized for use in sensor networks, SNEP and μ TESLA. SNEP provides confidentiality, authentication, and freshness between nodes and the sink, and μ TESLA provides authenticated broadcast.

V. ATTACKS ON SENSOR NETWORK ROUTING

Many sensor network routing protocols are quite simple, and for this reason are sometimes even more susceptible to attacks against general ad-hoc routing protocols. Most network layer attacks against sensor networks fall into one of the following categories:

- Spoofed, altered, or replayed routing information.
- Selective forwarding
- Sinkhole attacks
- Sybil attacks
- Wormholes
- HELLO flood attacks
- Acknowledgement spoofing

In the descriptions below, note the difference between attacks that try to manipulate user data directly and attacks that try to affect the underlying routing topology. We start with some general discussion of these types of attacks; in Section VII, we show how these attacks may be applied to compromise routing protocols that have been proposed in the literature.

A. Spoofed, altered, or replayed routing information

The most direct attack against a routing protocol is to target the routing information exchanged between nodes. By spoofing, altering, or replaying routing information, adversaries may be able to create routing loops, attract or repel network traffic, extend or shorten source routes, generate false error messages, partition the network, increase end-to-end latency, etc.

B. Selective forwarding

Multi-hop networks are often based on the assumption that participating nodes will faithfully forward received messages. In a selective forwarding attack, malicious nodes may refuse to forward certain messages and simply drop them, ensuring that they are not propagated any further. A simple form of this attack is when a malicious node behaves like a black hole and refuses to forward every packet she sees. However, such an attacker runs the risk that neighboring nodes will conclude that she has failed and decide to seek another route. A more subtle form of this attack is when an adversary selectively forwards packets. An adversary interested in suppressing or modifying packets originating from a select few nodes can reliably forward the remaining traffic and limit suspicion of her wrongdoing. Selective forwarding attacks are typically

most effective when the attacker is explicitly included on the path of a data flow. However, it is conceivable an adversary *overhearing* a flow passing through neighboring nodes might be able to emulate selective forwarding by jamming or causing a collision on each forwarded packet of interest. The mechanics of such an effort are tricky at best, and may border on impossible. Thus, we believe an adversary launching a selective forwarding attack will likely follow the path of least resistance and attempt to include herself on the actual path of the data flow. In the next two sections, we discuss sinkhole attacks and the Sybil attack, two mechanisms by which an adversary can efficiently include herself on the path of the targeted data flow.

C. Sinkhole attacks

In a sinkhole attack, the adversary's goal is to lure nearly all the traffic from a particular area through a compromised node, creating a metaphorical sinkhole with the adversary at the center. Because nodes on, or near, the path that packets follow have many opportunities to tamper with application data, sinkhole attacks can enable many other attacks (selective forwarding, for example). Sinkhole attacks typically work by making a compromised node look especially attractive to surrounding nodes with respect to the routing algorithm. For instance, an adversary could spoof or replay an advertisement for an extremely high quality route to a base station. Some protocols might actually try to verify the quality of route with end-to-end acknowledgements containing reliability or latency information. In this case, a laptop-class adversary with a powerful transmitter can actually *provide* a high quality route by transmitting with enough power to reach the base station in a single hop, or by using a wormhole attack discussed in Section VI-E. Due to either the real or imagined high quality route through the compromised node, it is likely each neighboring node of the adversary will forward packets destined for a base station through the adversary, and also propagate the attractiveness of the route to its neighbors. Effectively, the adversary creates a large "sphere of influence", attracting all traffic destined for a base station from nodes several (or more) hops away from the compromised node. One motivation for mounting a sinkhole attack is that it makes selective forwarding trivial. By ensuring that all traffic in the targeted area flows through a compromised node, an adversary can selectively suppress or modify packets originating from any node in the area.

It should be noted that the reason sensor networks are particularly susceptible to sinkhole attacks is due to their specialized communication pattern. Since all packets share the same ultimate destination (in networks with only one base station), a compromised node needs only to provide a single high quality route to the base station in order to influence a potentially large number of nodes.

D. The Sybil attack

In a Sybil attack [2], a single node presents multiple identities to other nodes in the network. The Sybil attack can significantly reduce the effectiveness of fault-tolerant schemes such as distributed storage [24], dispersity [25] and multipath [26] routing, and topology maintenance [27], [28]. Replicas, storage partitions, or routes believed to be using disjoint nodes could in actuality be using a single adversary presenting multiple identities. Sybil attacks also pose a significant threat to geographic routing protocols. Location aware routing often requires nodes to exchange coordinate information with their neighbors to efficiently route geographically addressed packets. It is only reasonable to expect a node to accept but a single set of coordinates from each of its neighbors, but by using the Sybil attack an adversary can "be in more than one place at once".

E. Wormholes

In the wormhole attack [1], an adversary tunnels messages received in one part of the network over a low latency link and replays them in a different part5. The simplest instance of this attack is a single node situated between two other nodes forwarding messages between the two of them. However, wormhole attacks more commonly involve two distant malicious nodes colluding to understate their distance from each other by relaying packets along an out-of-bound channel available only to the attacker. An adversary situated close to a base station may be able to completely disrupt routing by creating a well-placed wormhole. An adversary could convince nodes who would normally be multiple hops from a base station that they are only one or two hops away via the wormhole. This can create a sinkhole: since the adversary on the other side of the wormhole can artificially provide a high-quality route to the base station, potentially all traffic in the surrounding area will be drawn through her if alternate routes are significantly less attractive. This will most likely always be the case when the endpoint of the wormhole is relatively far from a base station. Figure 6 shows an example of a wormhole being used to create a sinkhole. Wormholes can also be used simply to convince two distant nodes that they are neighbors by relaying packets between the two of them.

Wormhole attacks would likely be used in combination with selective forwarding or eavesdropping. Detection is potentially difficult when used in conjunction with the Sybil attack.

F. HELLO *flood attack*

We introduce a novel attack against sensor networks: the HELLO flood. Many protocols require nodes to broadcast HELLO packets to announce themselves to their neighbors, and a node receiving such a packet may assume that it is within (normal) radio range of the sender. This assumption may be false: a laptop-class attacker broadcasting routing or other information with large enough transmission power could convince every node in the network that the adversary is its neighbor. For example, an adversary advertising a very high quality route to the base station to every node in the network could cause a large number of nodes to attempt to use this route, but those nodes sufficiently far away from the adversary would be sending packets into oblivion. The network is left in a state of confusion. A node realizing the link to the adversary is false could be left with few options: all its neighbors might be attempting to forward packets to the adversary as well. Protocols which depend on localized information exchange between neighboring nodes for topology maintenance or flow control are also subject to this attack. An adversary does not necessarily need to be able to construct legitimate traffic in order to use the

HELLO flood attack. She can simply re-broadcast overhead packets with enough power to be received by every node in the network. HELLO floods can also be thought of as one-way, broadcast wormholes.

Note: "Flooding" is usually used to denote the epidemic like propagation of a message to every node in the network over a multi-hop topology. In contrast, despite its name, the HELLO flood attack uses a single hop broadcast to transmit a message to a large number of receivers.

G. Acknowledgement spoofing

Several sensor network routing algorithms rely on implicit or explicit link layer acknowledgements. Due to the inherent broadcast medium, an adversary can spoof link layer acknowledgments for "overheard" packets addressed to neighboring nodes. Goals include convincing the sender that a weak link is strong or that a dead or disabled node is alive. For example, a routing protocol may select the next hop in a path using link reliability. Artificially reinforcing a weak or dead link is a subtle way of manipulating such a scheme. Since packets sent along weak or dead links are lost, an adversary can effectively mount a selective forwarding attack using acknowledgement spoofing by encouraging the target node to transmit packets on those links.

VI. ATTACKS ON SPECIFIC SENSOR NETWORK PROTOCOLS

All of the proposed sensor network routing protocols are highly susceptible to attack. Adversaries can attract or repel traffic flows, increase latency, or disable the entire network with sometimes as little effort as sending a single packet. In this section, we survey the proposed sensor network routing protocols and highlight the relevant attacks.

A. TinyOS beaconing

The TinyOS beaconing protocol constructs a breadth first spanning tree rooted at a base station. Periodically the base station broadcasts a route update. All nodes receiving the update mark the base station as its parent and rebroadcast the update. The algorithm continues recursively with each node marking its parent as the first node from which it hears a routing update during the current *time epoch*. All packets received or generated by a node are forwarded to its parent (until they reach the base station).

Attacks: The TinyOS beaconing protocol is highly susceptible

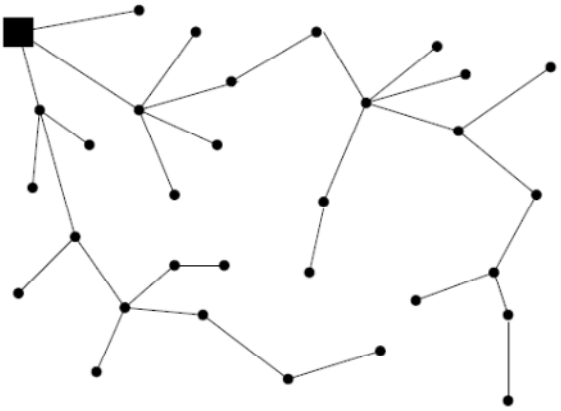

Fig. 4. A representative topology constructed using TinyOS beaconing with a single base station.

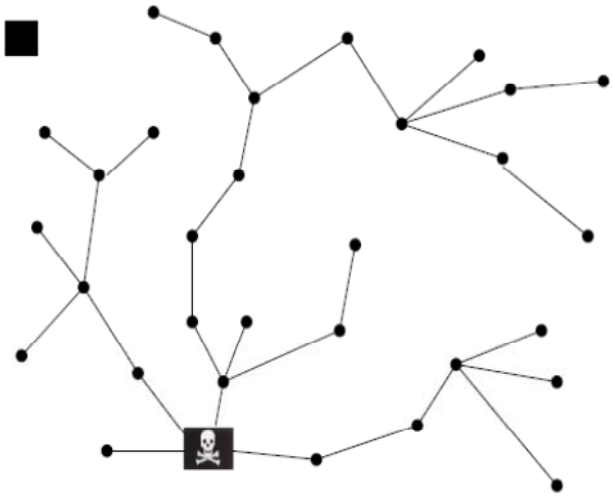

Fig. 5. An adversary spoofing a routing update from a base station in TinyOS beaconing.

to attack. Since routing updates are not authenticated, it is possible for any node to claim to be a base station and become the destination of all traffic in the network (see Figure 5). Authenticated routing updates will prevent an adversary from claiming to be a base station, but a powerful laptop class adversary can still easily wreak havoc. An adversary interested in eavesdropping on, modifying, or suppressing packets in a particular area can do so by mounting a combined wormhole/sinkhole attack. The adversary first creates a wormhole between two colluding laptop-class nodes, one near the base station and one near the targeted area. The first node forwards (authenticated) routing updates to the second through the wormhole, who participates normally in the protocol and rebroadcasts the routing update in the targeted area. Since the "wormholed" routing update will likely reach the targeted area considerably faster than it normally would have through multi-hop routing, the second node will create a large routing sub-tree in the targeted area with itself as the root. As seen in Figure 6, all traffic in the targeted area will be channeled through the wormhole, enabling a potent selective forwarding attack. If a laptop-class adversary has a powerful transmitter, it can use a HELLO flood attack to broadcast a routing update

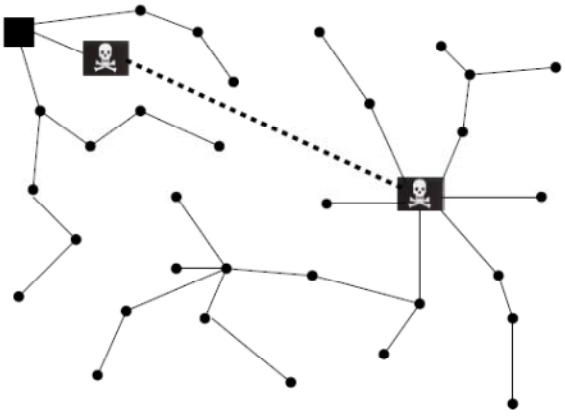

Fig. 6. A laptop-class adversary using a wormhole to create a sinkhole in TinyOS beaconing.

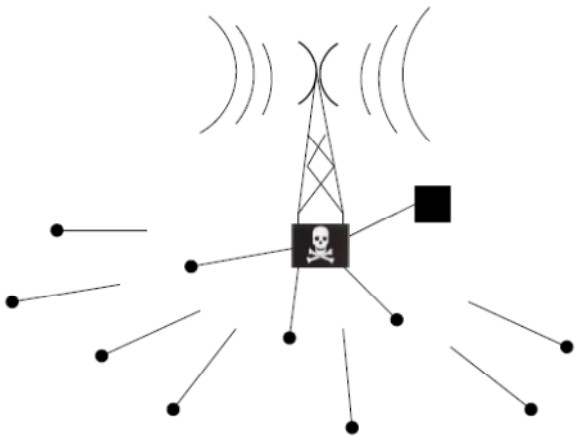

Fig. 7. HELLO **flood attack against TinyOS beaconing.** A laptop-class adversary that can retransmit a routing update with enough power to be received by the entire network leaves many nodes stranded. They are out of normal radio range from the adversary but have chosen her as their parent.

loud enough to reach the entire network, causing every node to mark the adversary as its parent. Most nodes will be likely out of normal radio range of both a true base station and the adversary. As shown in Figure 7, the network is crippled: the majority of nodes are stranded, sending packets into oblivion. Due to the simplicity of this protocol, it is unlikely there exists a simple extension to recover from this attack. A node that realizes its parent is not actually in range (say by using link layer acknowledgements) has few options short of flooding every packet. Each of its neighbors will likely have the adversary marked as its parent as well. Routing loops can easily be created by mote-class adversaries spoofing routing updates. Suppose an adversary can determine that node A and node B are within radio range of each other. An adversary can send a forged routing update to node B with a spoofed source address indicating it came from node A. Node B will then mark node A as its parent and rebroadcast the routing update. Node A will then hear the routing update from node B and mark B as it is parent. Messages sent to either A or B will be forever forwarded in a loop between the two of them.

B. Directed diffusion

Directed diffusion [29] is a data-centric routing algorithm for drawing information out of a sensor network. Base stations flood interests for named data, setting up gradients within the network designed to draw events (i.e., data matching the interest). Nodes able to satisfy the interest disseminate information along the reverse path of interest propagation. Nodes receiving the same interest from multiple neighboring nodes may propagate events along the corresponding multiple links. Interests initially specify a low rate of data flow, but once a base station starts receiving events it will reinforce one (or more) neighbor in order to request higher data rate events. This process proceeds recursively until it reaches the nodes generating the events, causing them to generate events at a higher data rate. Alternatively, paths may be negatively reinforced as well. There is a multipath variant of directed diffusion [30] as well. After the primary dataflow is established using positive reinforcements, alternate routes are recursively established with maximal disjointedness by attempting to reinforce neighbors not on the primary path.

Attacks: Due to the robust nature of flooding, it may be difficult for an adversary to prevent interests from reaching targets able to satisfy them. However, once sources begin to generate data events, an adversary attacking a data flow might have one of four goals:

Suppression: Flow suppression is an instance of denial-of-service. The easiest way to suppress a flow is to spoof negative reinforcements.

Cloning: Cloning a flow enables eavesdropping. After an adversary receives an interest flooded from a legitimate base station, it can simply replay that interest with herself listed as a base station. All events satisfying the interest will now be sent to both the adversary and the legitimate base station.

Path influence: An adversary can influence the path taken by a data flow by spoofing positive and negative reinforcements and bogus data events. For example, after receiving and rebroadcasting an interest, an adversary interested in directing the resulting flow of events through herself would strongly reinforce the nodes to which the interest was sent while spoofing high rate, low latency events to the nodes from which the interest was received. Three actions result:

(1) Data events generated upstream by legitimate sources will be drawn through the adversary because of her artificially strong positive reinforcements,

(2) Alternate event flows will be negatively reinforced by downstream nodes because the adversary provides (or spoofs) events with the lowest latency or highest frequency, and

(3) The adversary's node will be positively reinforced due the high quality spoofed and real data events she is able to provide. With this attack, an adversary is able to ensure any flow of events propagates through herself on the way to the base station that originally advertised the associated interest.

VII. COUNTERMEASURES

A. Outsider attacks and link layer security

The majority of outsider attacks against sensor network routing protocols can be prevented by simple link layer encryption and authentication using a globally shared key. The Sybil attack is no longer relevant because nodes are unwilling to accept even a single identity of the adversary. The majority of selective forwarding and sinkhole attacks are not possible because the adversary is prevented from joining the topology. Link layer acknowledgements can now be authenticated. Major classes of attacks not countered by link layer encryption and authentication mechanisms are wormhole attacks and HELLO flood attacks. Although an adversary is prevented from joining the network, nothing prevents her from using a wormhole to tunnel packets sent by legitimate nodes in one part of the network to legitimate nodes in another part to convince them they are neighbors or by amplifying an overheard broadcast packet with sufficient power to be received by every node in the network. The attacks against TinyOS beaconing described in Section VII-A illustrate these techniques, and link layer security mechanisms can do

nothing to prevent them. If a wormhole has been established, encryption may make some selective forwarding attacks against packets using the wormhole more difficult, but clearly can do nothing to prevent "black hole" selective forwarding. Link layer security mechanisms using a globally shared key are completely ineffective in presence of insider attacks or compromised nodes. Insiders can attack the network by spoofing or injecting bogus routing information, creating sinkholes, selectively forwarding packets, using the Sybil attack, and broadcasting HELLO floods. More sophisticated defense mechanisms are needed to provide reasonable protection against wormholes and insider attacks. We focus on countermeasures against these attacks in the remaining sections.

B. The Sybil attack

An insider cannot be prevented from participating in the network, but she should only be able to do so using the identities of the nodes she has compromised. Using a globally shared key allows an insider to masquerade as *any* (possibly even nonexistent) node. Identities must be verified. In the traditional setting, this might be done using public key cryptography, but generating and verifying digital signatures is beyond the capabilities of sensor nodes. One solution is to have every node share a unique symmetric key with a trusted base station. Two nodes can then use a Needham-Schroeder like protocol to verify each other's identity and establish a shared key. A pair of neighboring nodes can use the resulting key to implement an authenticated, encrypted link between them. In order to prevent an insider from wandering around a stationary network and establishing shared keys with every node in the network, the base station can reasonably limit the number of neighbors a node is allowed to have and send an error message when a node exceeds it. Thus, when a node is compromised, it is restricted to (meaningfully) communicating only with its verified neighbors. This is not to say that nodes are forbidden from sending messages to base stations or aggregation points multiple hops away, but they are restricted from using any node except their verified neighbors to do so. In addition, an adversary can still use a wormhole to create an artificial link between two nodes to convince them they are neighbors, but the adversary will not be able to eavesdrop on or modify any future communications between them.

C. HELLO *flood attacks*

The simplest defense against HELLO flood attacks is to verify the bi-directionality of a link before taking meaningful action based on a message received over that link. The identity verification protocol described in Section VIII-B is sufficient to prevent HELLO flood attacks. Not only does it verify the bi-directionality of the link between two nodes, but even if a well-funded adversary had a highly sensitive receiver or had wormholes to a multiple locations in the network, a trusted base station that limits the number of verified neighbors for each node will still prevent HELLO flood attacks on large segments of the network when a small number of nodes have been compromised.

D. Wormhole and sinkhole attacks

Wormhole and sinkhole attacks are very difficult to defend against, especially when the two are used in combination. Wormholes are hard to detect because they use a private, out-of-band channel invisible to the underlying sensor network. Sinkholes are difficult to defend against in protocols that use advertised information such as remaining energy or an estimate of end-to-end reliability to construct a routing topology because this information is hard to verify. Routes that minimize the hop-count to a base station are easier to verify, however hop-count can be completely misrepresented through a wormhole. When routes are established simply based on the reception of a packet as in TinyOS beaconing or directed diffusion, sinkholes are easy to create because there is no information for a defender to verify. A technique for detecting wormhole attacks is presented in [1], but it requires extremely tight time synchronization and is thus infeasible for most sensor networks. Because it is extremely difficult to retrofit existing protocols with defenses against these attacks, the best solution is to carefully design routing protocols in which wormholes and sinkholes are meaningless.

For example, one class of protocols resistant to these attacks is geographic routing protocols. Protocols that construct a topology initiated by a base station are most susceptible to wormhole and sinkhole attacks. Geographic protocols construct a topology on demand using only localized interactions and information and without initiation from the base station. Because traffic is naturally routed towards the physical location of a base station, it is difficult to attract it elsewhere to create a sinkhole. A wormhole is most effective when used to create sinkholes or artificial links that attract traffic. Artificial links are easily detected in geographic routing protocols because the "neighboring" nodes will notice the distance between them is well beyond normal radio range.

E. Leveraging global knowledge

A significant challenge in securing large sensor networks is their inherent self-organizing, decentralized nature. When the network size is limited or the topology is well-structured or controlled, global knowledge can be leveraged in security mechanisms. Consider a relatively small network of around 100 nodes or less. If it can be assumed that no nodes are compromised during deployment, then after the initial topology is formed, each node could send information such as neighboring nodes and its geographic location (if known) back to a base station. Using this information, the base station(s) can map the topology of the entire network. To account for topology changes due to radio interference or node failure, nodes would periodically update a base station with the appropriate information. Drastic or suspicious changes to the topology might indicate a node compromise, and the appropriate action can be taken. We have discussed why geographic routing can be relatively secure against wormhole, sinkhole, and Sybil attacks, but the main remaining problem is that location information advertised from neighboring nodes must be trusted. A compromised node advertising its location on a line between the targeted node and a base station will guarantee it is the destination for all forwarded packets from that node. Probabilistic selection of a next hop from several acceptable destinations or multipath routing to multiple base stations can help with this problem, but it is not perfect. When a node must route around a "hole", an

adversary can "help" by appearing to be the only reasonable node to forward packets to. Sufficiently restricting the structure of the topology can eliminate the requirement for nodes to advertise their locations if all nodes' locations are well known. For example, nodes can be arranged in a grid with square, triangular, or hex shaped cells. Every node can easily derive its neighbors' locations from its own, and nodes can be addressed by location rather than by an identifier.

F. Selective forwarding

Even in protocols completely resistant to sinkholes, wormholes, and the Sybil attack, a compromised node has a significant probability of including itself on a data flow to launch a selective forwarding attack if it is strategically located near the source or a base station. Multipath routing can be used to counter these types of selective forwarding attacks. Messages routed over η paths whose nodes are completely disjoint are completely protected against selective forwarding attacks involving at most η compromised nodes and still offer some probabilistic protection when over η nodes are compromised. However, completely disjoint paths may be difficult to create. Braided paths [30] may have nodes in common, but have no links in common. The use of multiple braided paths may provide probabilistic protection against selective forwarding and use only localized information. Allowing nodes to dynamically choose a packet's next hop probabilistically from a set of possible candidates can further reduce the chances of an adversary gaining complete control of a data flow.

VIII. CONCLUSION

Secure routing is vital to the acceptance and use of sensor networks for many applications, but we have demonstrated that currently proposed routing protocols for these networks are insecure. We leave it as an open problem to design a sensor network routing protocol that satisfies our proposed security goals. Link layer encryption and authentication mechanisms may be a reasonable first approximation for defense against mote-class outsiders, but cryptography is not enough to defend against laptop-class adversaries and insiders: careful protocol design is needed as well.

REFERENCES

[1] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Wormhole detection in wireless ad hoc networks," Department of Computer Science, Rice University, Tech. Rep. TR01-384, June 2002.

[2] J. R. Douceur, "The Sybil Attack," in *1st International Workshop on Peer-to-Peer Systems (IPTPS '02)*, March 2002.

[3] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, "System architecture directions for networked sensors," in *Proceedings of ACM SPLOS IX*, November 2000.

[4] V. D. Park and M. S. Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks," in *IEEE INFOCOM '97*, 1997, pp. 1405–1413.

[5] C. Perkins and E. Royer, "Ad-hoc on-demand distance vector routing," in *MILCOM '97 panel on Ad Hoc Networks*, 1997.

[6] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*, Imielinski and Korth, Eds. Kluwer Academic Publishers, 1996, vol. 353.

[7] C. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *ACM/SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications*, 1994, pp. 234–244.

[8] L. Zhou and Z. Haas, "Securing ad hoc networks," *IEEE Network Magazine*, vol. 13, no. 6, November/December 1999.

[9] F. Stajano and R. J. Anderson, "The resurrecting duckling: Security issues for ad-hoc wireless networks," in *Seventh International Security Protocols Workshop*, 1999, pp. 172–194.

[10] J. Hubaux, L. Buttyan, and S. Capkun, "The quest for security in mobile ad hoc networks," in *Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC 2001)*, 2001.

[11] J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang, "Providing robust and ubiquitous security support for mobile ad-hoc networks," in *ICNP*, 2001, pp. 251–260.

[12] M. G. Zapata, "Secure ad-hoc on-demand distance vector (SAODV) routing," IETF MANET Mailing List, Message-ID: 3BC17B40.BBF52E09@nokia.com, Available at ftp://manet.itd.nrl.navy.mil/pub/manet/2001-10.mail, October 8, 2001.

[13] H. Luo, P. Zefros, J. Kong, S. Lu, and L. Zhang, "Self-securing ad hoc wireless networks," in *Seventh IEEE Symposium on Computers and Communications (ISCC '02)*, 2002.

[14] J. Binkley and W. Trost, "Authenticated ad hoc routing at the link layer for mobile systems," *Wireless Networks*, vol. 7, no. 2, pp. 139–145,2001.

[15] B. Dahill, B. N. Levine, E. Royer, and C. Shields, "A secure routing protocol for ad-hoc networks," Electrical Engineering and Computer Science, University of Michigan, Tech. Rep. UM-CS-2001-037, August 2001.

[16] J. Kong, H. Luo, K. Xu, D. L. Gu, M. Gerla, and S. Lu, "Adaptive security for multi-layer ad-hoc networks," *Special Issue of Wireless Communications and Mobile Computing, Wiley Interscience Press*, 2002.

[17] Y.-C. Hu, D. B. Johnson, and A. Perrig, "SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks," in *Proceedings of the 4th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 2002)*, June 2002, pp. 3–13.

[18] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Ariadne: A secure on-demand routing protocol for ad hoc networks," Department of Computer Science, Rice University, Tech. Rep. TR01-383, December 2001.

[19] S. Basagni, K. Herrin, E. Rosti, and D. Bruschi, "Secure pebblenets," in *ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2001)*, October 2001, pp. 156–163.

[20] P. Papadimitratos and Z. Haas, "Secure routing for mobile ad hoc networks," in *SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS 2002)*, January 2002.