IPHV7I3019X
International Journal Of Advanced Research and Innovation -Vol.7, Issue .III
ISSN Online: 2319 – 9253
Print: 2319 – 9245

MITIGATING EFFECT OF VAMPIRE ATTACKS FROM WIRELESS ADHOCK SENSOR NETWORKS

[#1]CHINTHALA VIJAYAKRANTHI - M.Tech Pursuing,
[#2]P.V.SARATH CHAND - Associate Professor,HOD,
Department of Computer Science & Engineering,
INDUR INSTITUTE OF ENGINEERING &TECHNOLOGY, SIDDIPET, MEDAK, TS, India.

ABSTRACT: Wireless Ad-hoc Sensor Network is an emerging platform in the field of remote sensing, data collection, analysis, rectification of the problem and research in various studies. The objective of this paper is to examine resource depletion attacks at the routing protocol layer, which attempts to permanently disable network nodes by quickly draining their battery power. This type of attack is called as vampire attack. These attacks are not specific to any protocol, but rather rely on the properties of many popular classes of routing protocols. In the worst case, a single Vampire can increase network-wide energy usage by a factor of O(N), where N is the number of network nodes. Methods to detect and secure data packets from vampires during the packet forwarding phase is discussed.

Keywords*: *Denial of service, routing, ad-hoc networks, sensor networks, wireless networks, routing.

I. INTRODUCTION

Ad-hoc wireless sensor networks (WSNs) provides continuous connectivity, and instantly deployable communication. Such networks are capable of monitoring environmental conditions, factory performance, and troop's deployment. As WSNs become more and more crucial to everyday functioning of individuals and organizations, high availability of these networks is a critical property and should function without failure even under malicious conditions. Since their communication network is ad hoc in nature, wireless ad hoc networks are particularly vulnerable to denial of service (DoS) attacks [2], and a great deal of research has been done to enhance survivability [3], [4], [5], [6], [7].
Vampire attacks are not protocol-specific, as they do not rely on design properties or implementation faults of particular routing protocols. These attacks do not rely on flooding the network with large amounts of data, but rather try to transmit as little data as possible to achieve the largest energy drain.

Contributions. This paper makes three primary contributions. First, a thorough evaluation of the existing routing protocols towards battery depletion attacks is done. We observe that existing secure routing protocols such as Ariadne [10], SAODV [8], and SEAD [9] do not protect against Vampire attacks. Existing work on secure routing attempts to ensure that adversaries cannot cause path discovery to return an invalid network path, but Vampires do not disrupt or alter discovered paths, instead use existing valid network paths to carry out the attack. Protocols that maximize power efficiency are also inappropriate, since they rely on cooperative node behavior and cannot optimize battery power usage. Second, simulation results quantifying the performance of several representative protocols in the presence of a single Vampire (insider adversary) is shown. Third, modification of an existing sensor network routing protocol is made to prevent the damage caused by Vampire attacks during packet forwarding phase.

1.1 Classification

Denial of service is an attack, where a victim can use 10 minutes of the CPU time to transmit a data packet, but whereas an honest node uses 1 minute of its CPU time to transmit the same data packet. In multihop routing network: a source composes the shortest path and transmits the data packet to the next hop, which transmits it further, until the destination is reached; consuming resources not only at the source node but also at every node the packet moves through.

Vampire attack can be defined as a voluntary action of composing and transmitting a malicious message that chooses the longest path which consumes more energy of the network than if an honest node transmits a message of identical size to the same destination. The strength of an attack can be measured by the ratio of network energy used in the honest case to the energy used in the malicious case.

IPHV7I3019X
International Journal Of Advanced Research and Innovation -Vol.7, Issue .III
ISSN Online: 2319 – 9253
Print: 2319 – 9245

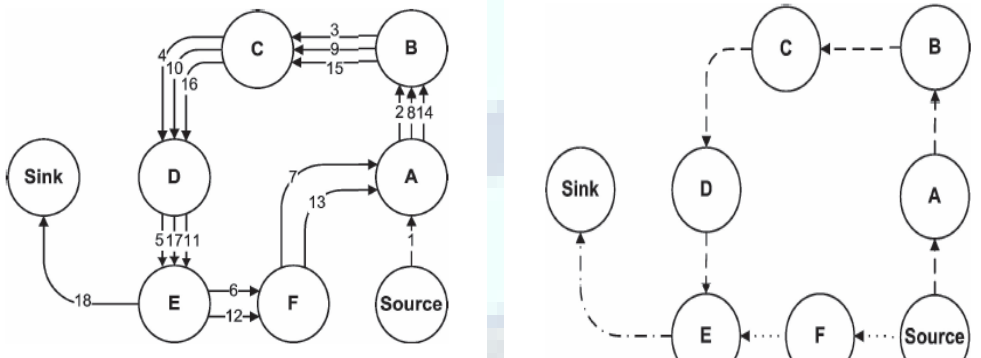
1.2 Protocols and Assumptions

In this paper, we consider the effect of Vampire attacks on Destination sequence distance vector routing protocols, as well as a logical ID-based sensor network routing protocol proposed by Parno et al. [11]. These protocols are likely to prevent Vampire attacks, so the covered protocols are an important subset of our routing solution space. We differentiate on-demand routing protocols, where topology discovery is done at transmission time, and static protocols, where topology is discovered during an initial phase, with periodic rediscovery to handle rare topology changes.

The adversaries are malicious insiders and have the same resources and level of network access as honest nodes. Sending malicious packet automatically allows few Vampires to attack many honest nodes. We will show later that a single Vampire may attack every network node simultaneously, meaning that vampires are to be isolated from the honest nodes. Vampire attacks may be weakened by using groups of nodes with staggered cycles: only active-duty nodes are vulnerable while the Vampire is active; nodes are safe while the Vampire sleeps

1.3 Overview

In the remainder of this paper, we present a series of increasingly damaging Vampire attacks, evaluate the vulnerability of several example protocols, and suggest how to improve flexibility. In source routing protocols, we show how a malicious packet source, can specify paths through the network, which are far longer than optimal, thus wasting energy at intermediate nodes that forward the packet as suggested by the source. In routing schemes, where forwarding decisions are made independently by each node (as opposed to specified by the source), we suggest how directional antenna and wormhole attacks [12] can be used to deliver packets to multiple remote network positions, forcing packet processing at nodes that would not normally receive that packet at all, and thus increasing network-wide energy expenditure. Lastly, we show how an adversary can target not only packet forwarding but also route and topology discovery phases—if discovery messages are flooded, an adversary can, for the cost of a single packet, consume energy at every node in the network



(a) An honest route would exit the loop immediately from node E to Sink, but a malicious packet makes its way around the loop twice more before exiting.

(b) Honest route is dotted while malicious route is dashed. The last link to the sink is shared.

Figure 1: Attacks on source routing (a) carousel attack (b) stretch attack

In our first attack, an adversary composes packets with purposely introduced routing loops. We call it the carousel attack, since it sends packets in circles as shown in Fig. 1a. It targets source routing protocols by exploiting the limited verification of message headers at forwarding nodes, allowing a single packet to repeatedly traverse the same set of nodes. Results show that in a randomly generated topology, a single attacker can use a carousel attack to increase energy consumption by as much as a factor of 4. Brief mentions of this attack can be found in other literature [11], [13], but no intuition for defence or any evaluation is provided. In our second attack, also targeting source routing, an adversary constructs artificially long routes, potentially traversing every node in the network.

We call this the stretch attack, since it increases packet path lengths, causing packets to be processed by a number of nodes that is independent of hop count along the shortest path between the adversary and packet destination. An example is illustrated in Fig. 1b. Stretch attacks increase energy usage by up to an order of magnitude, depending on the position of the malicious node. The impact of these attacks can be further increased by combining them, increasing the number of adversarial nodes in the network, or simply sending more packets. Although in networks that do not employ authentication or only use end-to-end authentication, adversaries are free to replace routes in any overheard packets.

We explore numerous mitigation methods to bound the damage from Vampire attacks, and find that while the carousel attack is simple to prevent with negligible overhead, the stretch attack is far more challenging. The first cause for vampire attack is loose

source routing, where any forwarding node can reroute the packet if it knows a shorter path to the destination. Thus, we modify the protocol developed by Parno et al [11] to guarantee that it mitigates all mentioned Vampire attacks. The topology discovery mechanism and one way hashing technique [23] is used for initial security stages. A sketch of how to modify the protocol to detect Vampire nodes during the packet forwarding phase and thus to isolate the adversarial nodes from the network is proposed.
In this paper, section II provides a brief description about the existing resource depletion attacks, section III and IV provides the impact of statefull and stateless protocols against vampire attacks with simulation results, section V describes about the proposed routing protocol M-DSDV, section VI deals with the current status and future work of the design.

II. RELATED WORKS

A very early mention of power exhaustion can be found in [21], as "sleep deprivation torture." As per the name, the proposed attack prevents nodes from entering a sleep cycle, and thus depletes their batteries faster. Newer research on "denial-of-sleep" only considers attacks at the MAC layer [14]. Malicious cycles (routing loops) have been briefly mentioned [11], [13], but no effective defenses are discussed other than increasing efficiency of the underlying MAC and routing protocols or switching away from source routing. Vampires do not drop packets; the quality of the malicious path itself may remain high.
Other work on denial of service in ad hoc wireless networks has primarily dealt with adversaries who prevent route setup, disrupt communication, or preferentially establish routes through themselves to drop, manipulate, or monitor packets [6], [9], [10], [15], [8]. The effect of denial or degradation of service on battery life and other finite node resources has not generally been a security consideration. Protocols that define security in terms of path discovery success, ensuring that only valid network paths are found, cannot protect against Vampire attacks, since Vampires do not use or return illegal routes or prevent communication in the short term.
Current work in minimal-energy routing, which aims to increase the lifetime of power-constrained networks by using less energy to transmit and receive packets[16], [17], [18],[19]. However, Vampires will increase energy usage even in minimal-energy routing scenarios. Attackers will produce packets which traverse more hops than necessary, so even if nodes spend the minimum required energy to transmit packets, each packet is still more expensive to transmit in the presence of Vampires. Our work can be thought of as attack-resistant minimal-energy routing, where the adversary's goal is to reduce energy savings.

III. ATTACKS ON STATELESS PROTOCOLS

In these systems, the source node specifies the entire route to a destination within the packet header, so intermediate nodes rely on the route specified by the source. The burden is on the source to ensure that the route is valid at the time of sending the data packet, and that every node in the route is a physical neighbour of the previous route hop. This approach has the advantage that intermediate nodes have fewer burdens while forwarding the data packets towards the destination, and also allows for entire routes to be sender authenticated using digital signatures, as in Ariadne [10].

3.1 Simulation Results

We evaluated both the carousel and stretch attacks (Fig. 1a) in a randomly generated 30-node topology and a single randomly selected malicious DSR agent, using the ns-2 network simulator [1]. Energy usage is measured for the minimum number of packets required to deliver a single message. We independently computed resource utilization of honest and malicious nodes and found that malicious nodes did not use an equal amount of energy as the honest nodes while carrying out the attack.

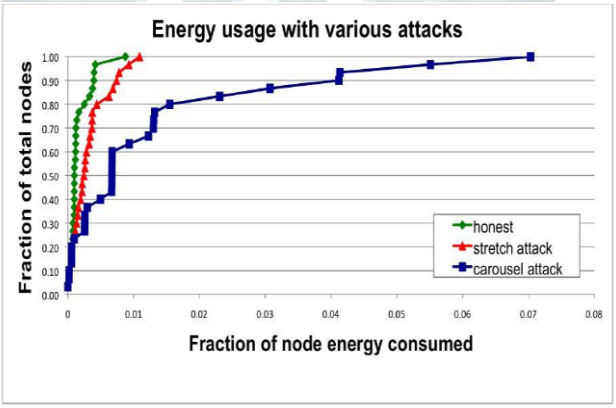


Figure 2. Results of a single malicious packet sent by the attacker is evaluated under both attacks is shown.

As expected, the carousel attack causes excessive energy usage for a few nodes, since only nodes along a shorter path are affected. In contrast, the stretch attack shows more uniform energy consumption for all nodes in the network, since it lengthens the route, causing more nodes to process the packet. While both attacks significantly use network energy unnecessarily, individual nodes are affected,

IPHV7I3019X
International Journal Of Advanced Research and Innovation -Vol.7, Issue .III
ISSN Online: 2319 – 9253
Print: 2319 – 9245

by losing almost 10 percent of their total energy per message. Fig. 3a diagrams the energy usage when node 0 sends a single packet to node 19 in an example network topology with only honest nodes. Black arrows denote the path of the packet.

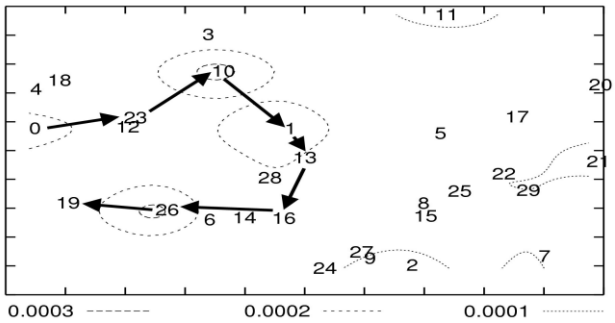


Figure 3a. Honest scenario: node 0 sends a single message to node 19

Carousel attack. In this attack, an adversary sends a packet with a malicious route composed as a series of loops, such that the same node appears in the route many times. This strategy can be used to increase the route length beyond the number of nodes in the network, only limited by the number of allowed entries in the source route. An example of this type of route is in Fig. 1a. In Fig. 3b, malicious node 0 carries out a carousel attack, sending a single message to node 19. The drastic increase in energy usage along the original path is shown. The theoretical limit of this attack is the energy usage increase by a factor of O(N), where N is the maximum route length.

Overall energy consumption increases by up to a factor of 3.96 per message. On average, a randomly located carousel attacker in our example topology can increase network energy consumption by a factor of 1.48 ± 0.99. The reason for this large standard deviation is that the attack does not always increase energy usage—the length of the adversarial path is a multiple of the honest path, which is in turn, affected by the position of the adversary in relation to the destination, so the adversary's position is important to the success of this attack.

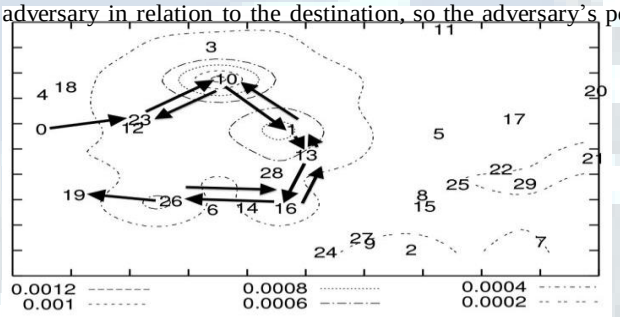


Figure 3b. Carousel attack (malicious node 0) the nodes traversed by the packet are the same as in (3a), but the loop overall forwarding nodes roughly triples the route length (the packet traverses the loop more than once).

Stretch attack. Another attack in the same vein is the stretch attack, where a malicious node constructs artificially long source routes, causing packets to traverse a larger number of nodes than choosing an optimal path. An honest source would select the route Source→F→E→Sink, affecting four nodes including itself, but the malicious node selects a longer route, affecting all nodes in the network. These routes force nodes that do not lie along the honest route to expel energy by forwarding malicious packets. An example of this type of route is in Fig. 1b. The outcome becomes clearer when we examine Fig. 3c and compare to the carousel attack. While the carousel attack uses energy at the nodes that were already in the honest path, but the stretch attack extends the routing path to a wider section of the network, and consumes energy from larger number of nodes.

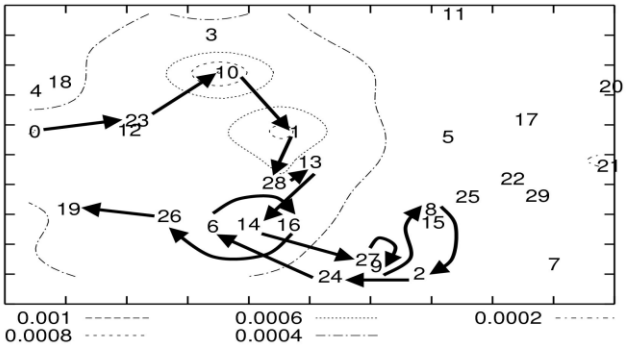


Figure 3c. Stretch attack (malicious node 0) the route diverts from the optimal path between source and destination, roughly doubling the length.

The theoretical limit of the stretch attack is a packet that traverses every network node, causing an energy usage increase by a factor of O (min(N,)), where N is the number of nodes in the network and is the maximum path length allowed. This attack is potentially less damaging per packet than the carousel attack, as the number of hops per packet is bounded by the number of network nodes. However, adversaries can combine carousel and stretch attacks to keep the packets within the network longer period of time. Therefore, stretch attack and routing loop problems should be detected and removed to prevent the combined attack. In our example topology, we see an increase in energy usage by as much as a factor of 10.5 per message over the honest scenario, with an average increase in energy consumption of 2.67 ± 2.49. As with the carousel attack, the reason for the large standard deviation is that the position of the adversarial node affects the strength of the attack. But, the stretch attack can achieve the same effectiveness and does not depend on the attacker's network position relative to the destination.

3.2 Mitigation Methods

The carousel attack can be prevented entirely by having, forwarding nodes to check the source routes for loops. When a loop is detected, it is better to simply drop the packet, especially considering that the sending node is likely malicious (honest nodes should not introduce loops).

The stretch attack is more challenging to prevent. Its success rests on the forwarding node not checking for optimality of the route, but simply follows the route exactly as specified in the header, this attack can be prevented by using loose source routing, where intermediate nodes may replace part or all of the route in the packet header if they know of a better route to the destination.

IV. ATTACKS ON STATEFUL PROTOCOLS

Two important classes of stateful protocols are link-state and distance-vector routing protools. In link state and distance vector network nodes are aware of the network topology, state and make independent forwarding decisions, so adversaries have limited power to affect packet forwarding, making these protocols immune to carousel and stretch attacks. But these protocols consume excess energy power as compared with the stateless protocols, since every node in the network frequently updates its routing table to keep track of the network nodes.

Directional antenna attack. Using directional antenna adversaries can deposit a packet in arbitrary parts of the network, while also forwarding the packet locally. This consumes the energy of nodes that would not have had to process the original packet, with the expected additional honest energy expenditure of O (d), where d is the network diameter. This attack can be considered a half-wormhole attack [12], since a directional antenna constitutes a private communication channel, but the node on the other end is not necessarily malicious. It can be performed more than once, depositing the packet at various distant points in the network, at the additional cost to the adversary for each use of the directional antenna.

Malicious discovery attack. Another attack on all previously mentioned routing protocols (including stateful and stateless) is spurious route discovery. A malicious node has a number of ways to induce a perceived topology change: it may simply falsely claim that a link is down, or claim a new link to a nonexistent node. Two cooperating malicious nodes may claim the link between them is down. However, nearby nodes might be able to monitor communication to detect link failure. a single node can emulate multiple nodes in neighbour relationships [20], or falsely claim nodes as neighbours countermeasure is to use authentication. To do this, two cooperating adversaries communicating through a wormhole could repeatedly announce and withdraw routes that use this wormhole, causing a theoretical energy usage increase of a factor of O (N) per packet.

Coordinate and beacon based protocols. These protocols also fall victim to directional antenna attacks in the same way as link-state and distance-vector protocols.

V. M-DSDV NETWORK ROUTING

In this section, we show that destination sequence distant vector a proactive network routing protocol [24] can be modified to provably resist Vampire attacks during the packet forwarding phase. Even though the existing DSDV is designed to overcome routing loop problems, it is still not a feasible method for efficient packet transmission, as the protocol is proactive which utilizes more battery power and bandwidth. M-DSDV consists of a topology discovery phase, followed by a topology maintenance phase. Legitimate network node has a unique certificate of membership, which includes its public key and code word (assigned by a trusted offline authority before network deployment). Topology discovery. Discovery of the neighbouring nodes begins, when there is a need to transmit the data packet. Each node has a limited view of the network—the node knows only itself. Nodes use the local broadcasting scheme to discover their neighbours, where the certificate identity verification is done to isolate the external unauthorized nodes from the network. Thus, each honest node learns its active neighbour node's address and public key.

When a source node S, wants to send a data packet to destination D, first constructs and broadcasts a route request packet consisting of (source address, destination address, sequence number, next hop, metric, index number and time to live) fields. The source address and destination address are the internet protocol addresses, the sequence number is used to differentiate new routes from stale routes, the next hop and metric is a local counter maintained separately by each node and incremented each time a RReq is

broadcasted, the index number is initialized to zero, is used to keep track of the loops the packet has made and the final time to live field is used as a clock which increments whenever a RReq packet is sent.

On receipt of RReq, intermediate nodes inspect it to see if it is a duplicate, in which case it is rejected. If not the (source address, next hop, metric) pair is entered into the local history table. The destination address is looked up in the routing table, if a fresh route to it is known an RRep a route reply packet is sent back to S. If not, it increments the index number and rebroadcasts the RReq. This also creates a backward route towards S and exists has an optimization technique.
When destination receives RReq, it sends back an RRep packet to the node from which it got the first RReq packet. The format of the route reply packet includes (source address, destination address, destination sequence, index number, life time). Here, the source address, destination address and index number are copied from the incoming RReq packet, but the destination sequence number is taken from its counter in memory. The life time field indicates how long the route is valid.
On receipt of RRep, intermediate nodes on the way back, inspect the packet and create a backward route towards destination.
Intermediate nodes that got the original RReq packet but were not on the reverse path discard the reverse route table entry when the associated timer expires.
Topology maintenance phase. When the next hop link in the routing table entry breaks, all active neighbours are informed by means of RERR packets which updates the sequence number. RERR packets are also generated when a node X is unable to forward packet P from node S to node D on link (X, Y). The incremented sequence number N is included in the RERR. When node S receives the RERR, it initiates a new route discovery for D using the sequence number that is at least as large as N.

5.1 M-DSDV in the presence of vampires

In the presence of vampires, carousel attack and stretch attack can be prevented by using the index number. In case of, carousel attack, where a packet which traversed through the shortest path of the network, returns back again to the same node, that could be eliminated by checking the index number stored on the packet header and the index number stored in the local routing table of the node.
We can prevent the stretch attack by independently checking on the packet progress: the nodes keep track of route "metric" and, when acknowledgement returns back, the route metric value and the index number, which indicates the hop count can be verified. If the index value is greater than the metric value the source concludes that the stretch attack as occurred.
Moreover, to prevent truncation of the routing path, which would allow Vampires to hide the fact that they are moving a packet away from its destination, we use Saxena and Soh's one-way signature chain construction [22], which allow nodes to add links to an existing signature chain, but not remove links, making attestations append only. Thus if malicious intervention has been suspected the packet is dropped from further forwarding strategy. Thus, the damage from an attacker is bounded as a function of network size.

VI. CONCLUSION & FUTURE WORK

Thus, Vampire attack, a new class of resource consumption attacks that use routing protocols to permanently disable ad hoc wireless sensor networks by depleting nodes' battery power has been briefly defined. These attacks do not depend on particular protocols or implementations, but rather expose vulnerabilities in a number of popular protocol classes. Survey on existing routing protocols has been done. Using a small number of adversaries, vampire attacks success on a randomly generated topology of 30 nodes is shown using simulation, this suggests, that depending on the location of the adversary, network energy expenditure during the forwarding phase increases rapidly. M-DSDV, routing protocol that provably bounds damage from Vampire attacks by verifying that packets consistently is proposed in this paper. Prevention of data packets from entering into a malicious node is left for future work.

REFERENCES

[1] "The Network Simulator - ns-2," http://www.isi.edu/nsnam/ns,2012.

[2] A.D. Wood and J.A. Stankovic, "Denial of Service in Sensor Networks," Computer, vol. 35, no. 10, pp. 54-62, Oct. 2002.

[3] I. Aad, J.-P. Hubaux, and E.W. Knightly, "Denial of Service Resilience in Ad Hoc Networks," Proc. ACM MobiCom, 2004.

[4] J. Bellardo and S. Savage, "802.11 Denial-of-Service Attacks: Real Vulnerabilities and Practical Solutions," Proc. 12th Conf. USENIX Security, 2003.

[5] J. Deng, R. Han, and S. Mishra, "Defending against Path-Based DoS Attacks in Wireless Sensor Networks," Proc. ACM Workshop Security of Ad Hoc and Sensor Networks, 2005.

[6] J. Deng, R. Han, and S. Mishra, "INSENS: Intrusion-Tolerant Routing for Wireless Sensor Networks," Computer Comm., vol. 29,no. 2, pp. 216-230, 2006.

[7] A. Nasipuri and S.R. Das, "On-Demand Multipath Routing for Mobile Ad Hoc Networks," Proc. Int'l Conf. Computer Comm. And Networks, 1999.

[8] M.G. Zapata and N. Asokan, "Securing Ad Hoc Routing Protocols," Proc. First ACM Workshop Wireless Security (WiSE),2002.

[9]Y.-C. Hu, D.B. Johnson, and A. Perrig, "SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks," Proc. IEEE Workshop Mobile Computing Systems and Applications, 2002.

[10] Y.-C. Hu, D.B. Johnson, and A. Perrig, "Ariadne: A Secure On- Demand Routing Protocol for Ad Hoc Networks," Proc. MobiCom, 2002.

[11] B. Parno, M. Luk, E. Gaustad, and A. Perrig, "Secure Sensor Network Routing: A Clean-Slate Approach," CoNEXT: Proc. ACM CoNEXT Conf., 2006.

[12] Y.-C. Hu, D.B. Johnson, and A. Perrig, "Packet Leashes: A Defense against Wormhole Attacks in Wireless Ad Hoc Networks," Proc.IEEE INFOCOM, 2003.

[13] H. Chan and A. Perrig, "Security and Privacy in Sensor Networks," Computer, vol. 36, no. 10, pp. 103-105, Oct. 2003.

[14] D.R. Raymond, R.C. Marchany, M.I. Brownfield, and S.F. Midkiff, "Effects of Denial-of-Sleep Attacks on Wireless Sensor Network MAC Protocols," IEEE Trans. Vehicular Technology, vol. 58, no. 1, pp. 367-380, Jan. 2009.
[15] C. Karlof and D. Wagner, "Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures," Proc. IEEE Int'l Workshop Sensor Network Protocols and Applications, 2003.

[16] J.-H. Chang and L. Tassiulas, "Maximum Lifetime Routing in Wireless Sensor Networks," IEEE/ACM Trans. Networking, vol. 12,no. 4, pp. 609-619, Aug. 2004.
[17] S. Doshi, S. Bhandare, and T.X. Brown, "An On-Demand minimum Energy Routing Protocol for a Wireless Ad Hoc Network," ACM SIGMOBILE Mobile Computing and Comm. Rev.,vol. 6, no. 3, pp. 50-66, 2002.
[18] L.M. Feeney, "An Energy Consumption Model for Performance Analysis of Routing Protocols for Mobile Ad Hoc Networks,"Mobile Networks and Applications, vol. 6, no. 3, pp. 239-249, 2001.

[19] V. Rodoplu and T.H. Meng, "Minimum Energy Mobile Wireless Networks," IEEE J. Selected Areas in Comm., vol. 17, no. 8, pp. 1333-1344, Aug. 1999.

[20] J.R. Douceur, "The Sybil Attack," Proc. Int'l Workshop Peer-to-Peer Systems, 2002.

[21] F. Stajano and R. Anderson, "The Resurrecting Duckling: Security Issues for Ad-Hoc Wireless Networks," Proc. Int'l Workshop Security Protocols, 1999.

[22] A. Saxena and B. Soh, "One-Way Signature Chaining: A New Paradigm for Group Cryptosystems," Int'l J. Information and Computer Security, vol. 2, no. 3, pp. 268-296, 2008.

[23] Eugene Y. Vasserman and Nicholas Hopper, "Vampire Attacks: Draining Life from Wireless Ad Hoc Sensor Networks," IEEE transactions on mobile computing, vol. 12, no. 2, pp. 318-332, february 2013
[24] "DSDV routing protocol," http://www.nsnam.org/docs/models/html/dsdv.html