**Vampire Attacks: Draining Life from Wireless Ad Hoc Sensor Networks**

**Abstract**

Ad hoc low-power wireless networks are an exciting research direction in sensing and pervasive computing. Prior security work in this area has focused primarily on denial of communication at the routing or medium access control levels. This paper explores resource depletion attacks at the routing protocol layer, which permanently disable networks by quickly draining nodes’ battery power. These “Vampire” attacks are not specific to any specific protocol, but rather rely on the properties of many popular classes of routing protocols. We find that all examined protocols are susceptible to Vampire attacks, which are devastating, difficult to detect, and are easy to carry out using as few as one malicious insider sending only protocol-compliant messages. In the worst case, a single Vampire can increase network-wide energy usage by a factor of O (N), where N in the number of network nodes. We discuss methods to mitigate these types of attacks, including a new proof-of-concept protocol that provably bounds the damage caused by Vampires during the packet forwarding phase.

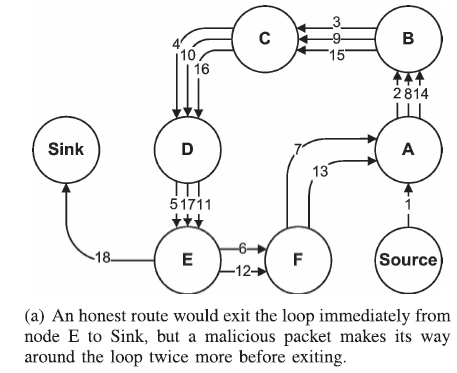
**Introduction**

AD hoc wireless sensor networks (WSNs) promise exciting new applications in the near future, such as ubiquitous on-demand computing power, continuous connectivity, and instantly deployable communication for military and first responders. Such networks already monitor environmental conditions, factory performance, and troop deployment, to name a few applications. As WSNs become more and more crucial to the everyday functioning of people and organizations, availability faults become less tolerable—lack of availability can make the difference between business as usual and lost productivity, power outages, environmental disasters, and even lost lives; thus high availability of these networks is a critical property, and should hold even under malicious conditions. Due to their ad hoc organization, wireless ad hoc networks are particularly vulnerable to denial of service (DoS) attacks, and a great deal of research has been done to enhance Survivability While these schemes can prevent attacks on the shortterm availability of a network, they do not address attacks that affect long-term availability—the most permanent denial of service attack is to entirely deplete nodes’

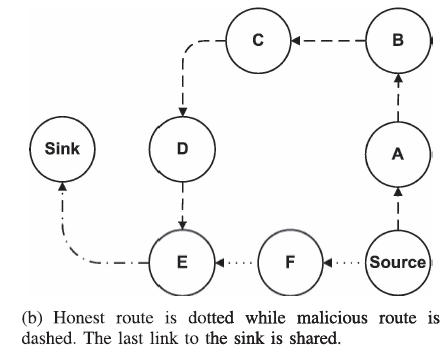
batteries. This is an instance of a resource depletion attack, with battery power as the resource of interest. In this paper, we consider how routing protocols, even those designed to be secure, lack protection from these attacks, which we call Vampire attacks, since they drain the life from networks nodes.

we present a series of increasingly damaging Vampire attacks, evaluate the vulnerability of several example protocols, and suggest how to improve resilience. In source routing protocols, we show how a malicious packet source can specify paths through the network which are far longer than optimal, wasting energy at intermediate nodes who forward the packet based on the included source route.

1. In our first attack, an adversary composes packets with purposely introduced routing loops. We call it the carousel attack



1. 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.



**Problem Definition**

The first challenge in addressing Vampire attacks is defining them—what actions in fact constitute an attack? DoS attacks in wired networks are frequently characterized by amplification an adversary can amplify the resources it spends on the attack, e.g., use 1 minute of its own CPU time to cause the victim to use 10 minutes. However, consider the process of routing a packet in any multihop network: a source composes and transmits it to the next hop toward the destination, which transmits it further, until the destination is reached, consuming resources not only at the source node but also at every node the message moves through. 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. 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. Brief mentions of this attack can be found in other literature, but no intuition for defense nor 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.

* Power draining attacks have not been rigorously defined, evaluated, or mitigated at the routing layer.

Existing research:

* “Sleep deprivation torture.”
* Resource exhaustion at the MAC and transport layers, but only offers rate limiting and elimination of insider adversaries
* Switching away from source routing as a solution
* SYN flood attack, wherein adversaries make multiple connection requests to a server, which will allocate resources for each connection request, eventually running out of resources. Solution: put more burdens on the Client.

**System Analysis**

**EXISTING SYSTEM:**

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 using existing valid network paths and protocol compliant messages. Protocols that maximize power efficiency are also inappropriate, since they rely on cooperative node behavior and cannot optimize out malicious action. While some of the individual attacks are simple, and power draining and resource exhaustion attacks have been discussed before prior work has been mostly confined to other levels of the protocol stack, e.g., medium access control (MAC) or application layers, and to our knowledge there is little discussion, and no thorough analysis or mitigation, of routing-layer resource exhaustion attacks. There are two types of attacks are occur carousel attack and 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. 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. Brief mentions of this attack can be found in other literature but neither intuition for defense nor 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.

**Disadvantages of Existing System:**

* Power outages
* Due to Environmental disasters, loss in the information
* Lost productivity
* Various DOS attacks
* Secure level is low
* They do not address attacks that affect long-term availability.

**PROPOSED SYSTEM:**

This paper makes three primary contributions. First, we thoroughly evaluate the vulnerabilities of existing protocols to routing layer battery depletion attacks. We observe that security measures to prevent Vampire attacks are orthogonal to those used to protect routing infrastructure, and so existing secure routing protocols. 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 using existing valid network paths and protocol-compliant messages. Protocols that maximize power efficiency are also inappropriate, since they rely on cooperative node behavior and cannot optimize out malicious action. Second, we show simulation results quantifying the performance of several representative protocols in the presence of a single Vampire (insider adversary). Third, we modify an existing sensor network routing protocol to provably bind the damage from Vampire attacks during packet forwarding.  In proposed system we show simulation results quantifying the performance of several representative protocols in the presence of a single Vampire. Then, we modify an existing sensor network routing protocol to provably bind the damage from Vampire attacks during packet forwarding.

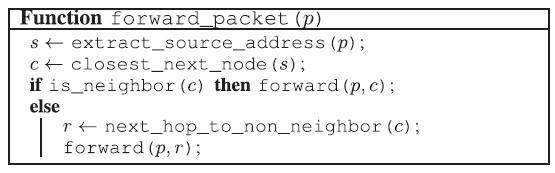
**Advantages of Proposed System:**

* Protect from the vampire attacks
* Secure level is high
* Boost up the Battery power

**Steps:**

1. **Packet Filtering**

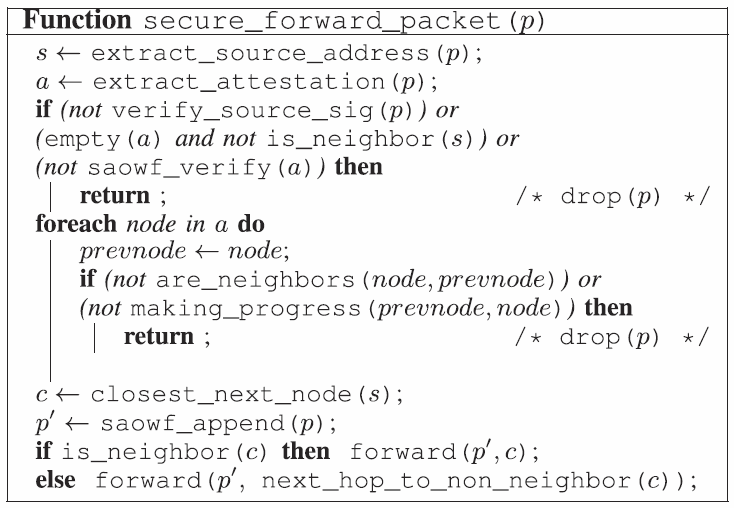
* Controlling access to a network by analyzing the incoming and outgoing [packets](http://www.webopedia.com/TERM/P/packet.html) and letting they pass or halting them based on the [IP addresses](http://www.webopedia.com/TERM/I/IP_address.html) of the source and destination. Packet filtering is one technique, among many, for implementing security [firewalls](http://www.webopedia.com/TERM/F/firewall.html).
* **Packet forwarding.** During the forwarding phase, all decisions are made independently by each node. When receiving a packet, a node determines the next hop by finding the most significant bit of its address that differs from the message originator’s address .
* Thus, every forwarding event (except when a packet is moving within a group in order to reach a gateway node to proceed to the next group) shortens the logical distance to the destination, since node addresses should be strictly closer to the destination



1. **Bootstrapping**

PLGP with attestations (**PLGPa**)**:**

* Add a verifiable path history to every PLGP packet
* The resulting protocol, PLGP with attestations (PLGPa) uses this packet history together with PLGP’s tree routing structure so every node can securely verify progress, preventing any significant adversarial influence on the path taken by any packet which traverses at least one honest node.
* These signatures form a chain attached to every packet, allowing any node receiving it to validate its path. Every forwarding node verifies the attestation chain to ensure that the packet has never traveled away from its destination in the logical address space.



**Literature Survey**

1. **Denial of Service Resilience in Ad Hoc Networks**

Significant progress has been made towards making ad hoc networks secure and DoS resilient. However, little attention has been focused on quantifying DoS resilience: Do ad hoc networks have sufficiently redundant paths and counter-DoS mechanisms to make DoS attacks largely ineffective? Or are there attack and system factors that can lead to devastating effects? In this paper, we design and study DoS attacks in order to assess the damage that difficultto- detect attackers can cause. The first attack we study, called the JellyFish attack, is targeted against closed-loop flows such as TCP; although protocol compliant, it has devastating effects. The second is the Black Hole attack, which has effects similar to the JellyFish, but on open-loop flows. We quantify via simulations and analytical modeling the scalability of DoS attacks as a function of key performance parameters such as mobility, system size, node density, and counter-DoS strategy. One perhaps surprising result is that such DoS attacks can increase the capacity of ad hoc networks, as they starve multi-hop flows and only allow one-hop communication, a capacity-maximizing, yet clearly undesirable situation.

Significant progress has been made in securing ad hoc networks via the development of secure routing protocols. Moreover, ensuring resilience to misbehavior and denial-ofservice attacks has also been the focus of significant research efforts as such resilience is a critical component of a secure system: examples include “watch-dog” mechanisms designed to detect and circumvent misbehaving nodes; rate-limiting of route-request messages to prevent route query-flood attacks; and “rushing attack prevention” that seeks to inhibit malicious nodes from attracting an excessive number of routes, which would increase their ability to inflict damage.

Our methodology is to study DoS resilience via a new and general class of protocol compliant denial-of-service attacks, which we refer to as JellyFish (JF). Although previously studied attackers disobey protocol rules, JellyFish conform to all routing and forwarding protocol specifications, and moreover, as implied by the name, are passive and difficult to detect until after the “sting.” JellyFish target closed-loop flows that are responsive to network conditions such as delay and loss. Examples include TCP flows and congestioncontrolled UDP flows employing a TFRC-like algorithm. The goal of JF nodes is to reduce the goodput of all traversing flows to near-zero while dropping zero or a small fraction of packets. In particular, JF nodes employ one of three mechanisms. The first JF variant is a packet misordering attack. TCP has a wellknown vulnerability to misordered packets due to factors such as route changes or the use of multi-path routing, and a number of TCP modifications have been proposed to improve robustness to misordering. However, no TCP variant is robust to malicious and persistent reordering as employed by the JF misordering attack. The second JF mechanism is periodic dropping according to a maliciously chosen period. This attack is inspired by the Shrew attack in which an endpoint sends maliciously spaced periodic pulses in order to force flows into repeated timeout phases. The JF periodic dropping attack utilizes the same principles but realizes the attack via periodic dropping at relay nodes. In particular, suppose that congestion losses force a node to drop x% of packets. if these losses occur periodically at the retransmission-time-out timescale (approximately 1 second), TCP throughput is reduced to near zero even for small values of x. Thus, a JF periodic-dropping node can drop no more packets than neighboring congested nodes, but inflict near-zero throughput on all TCP flows traversing it. Third, we consider a delay-variance attack in which the attacker randomly delays packets (preserving order) in order to thwart TCP’s timers and congestion inferences. This attack not only thwarts widely deployed TCP variants, but also can disrupt rate-based congestion control algorithms such as. Notice that JF nodes are protocol compliant in that IP’s datagram service does not mandate loss-free service, in-order delivery, or bounded delay jitter. Finally, in addition to the JF attack, we also study the “black hole” attack as described. This attack is relevant for openloop flows that do not respond to congestion, loss, or delay information, and hence cannot be thwarted by JellyFish. Black Hole nodes participate in the routing protocol to establish routes through themselves, yet drop all packets after correctly receiving them at the MAC layer.

1. **Provably Secure On-demand Source Routing in Mobile Ad Hoc Networks**

Routing is one of the most basic networking functions in mobile ad hoc networks. Hence, an adversary can easily paralyze the operation of the network by attacking the routing protocol. This has been realized by many researchers, and several “secure” routing protocols have been proposed for ad hoc networks. However, the security of those protocols have mainly been analyzed by informal means only. In this paper, we argue that flaws in ad hoc routing protocols can be very subtle, and we advocate a more systematic way of analysis. We propose a mathematical framework in which security can be precisely defined, and routing protocols for mobile ad hoc networks can be analyzed rigorously. Our framework is tailored for on-demand source routing protocols, but the general principles are applicable to other types of protocols too. Our approach is based on the simulation paradigm, which has already been used extensively for the analysis of key establishment protocols, but to the best of our knowledge, it has not been applied in the context of ad hoc routing so far. We also propose a new on-demand source routing protocol, called endairA, and we demonstrate the usage of our framework by proving that it is secure in our model.

Routing is one of the most basic networking functions in mobile ad hoc networks. Hence, an adversary can easily paralyze the operation of the network by attacking the routing protocol. This has been realized by many researchers, and several “secure” routing protocols have been proposed for ad hoc networks. However, the security of those protocols have been analyzed either by informal means only, or with formal methods that have never been intended for the analysis of this kind of protocols. In this paper, we present a new attack on Ariadne, a previously published “secure” routing protocol. Other attacks can be found. These attacks clearly demonstrate that flaws can be very subtle, and therefore, hard to discover by informal reasoning. Hence, we advocate a more systematic approach to analyzing ad hoc routing protocols, which is based on a rigorous mathematical model, in which precise definitions of security can be given, and sound proof techniques can be developed.

Routing has two main functions: route discovery and packet forwarding. The former is concerned with discovering routes between nodes, whereas the latter is about sending data packets through the previously discovered routes. There are different types of ad hoc routing protocols. One can distinguish proactive (e.g., OLSR) and reactive (e.g., AODV and DSR) protocols. Protocols of the latter category are also called on-demand protocols. Another type of classification distinguishes routing table based protocols (e.g., AODV) and source routing protocols (e.g., DSR). In this paper, we focus on the route discovery part of on-demand source routing protocols, but we believe that the general principles of our approach are applicable to the route discovery part of other types of protocols too.

At a very informal level, security of a routing protocol means that it can perform its functions even in the presence of an adversary. Obviously, the objective of the adversary is to prevent the correct functioning of the routing protocol. Since we are focusing on the route discovery part of on-demand source routing protocols, in our case, attacks are aiming at achieving that honest nodes receive “incorrect” routes as a result of the route discovery procedure. We will make it more precise later what we mean by an “incorrect” route. Regarding the capabilities of the adversary, we assume that it can mount active attacks (i.e., it can eavesdrop, modify, delete, insert, and replay messages) from corrupted nodes that have the same communication capabilities as the nodes of the honest participants in the network. This means that the adversary is not all powerful, and it cannot fully control the communication of the honest participants; it can receive only those messages that were transmitted by one of its neighbors, and its transmissions can be heard only by its neighbors. We further assume that the adversary has compromised some identifiers by which we mean that it has compromised the cryptographic keys that are used to authenticate those identifiers. Thus, the adversary can appear as an honest participant under the compromised identities. Using the notation introduced, our adversary is an Active-y-x adversary, which means that it controls x corrupted nodes in the network, and it can use y compromised identifiers. The mathematical framework that we introduce in this paper is based on the so called simulation paradigm. This has been successfully used in the analysis of some cryptographic algorithms and some cryptographic protocols. However, it has never been applied in the context of ad hoc routing protocols. One of the main contributions of this work is the application of this approach in a new context. Another contribution of this work is the discovery of as yet unknown attacks against previously published ad hoc routing protocols. Finally, yet another contribution is a new on-demand source routing protocol for mobile ad hoc networks, called endairA, which is provably secure in our model, and which may be of independent interest for practitioners.

1. **Denial-of-Service Attacks : Real Vulnerabilities and Practical Solutions**

The convenience of 802.11-based wireless access networks has led to widespread deployment in the consumer, industrial and military sectors. However, this use is predicated on an implicit assumption of confidentiality and availability. While the security flaws in 802.11’s basic confidentially mechanisms have been widely publicized, the threats to network availability are far less widely appreciated. In fact, it has been suggested that 802.11 is highly susceptible to malicious denial-of-service (DoS) attacks targeting its management and media access protocols.

This paper provides an experimental analysis of such 802.11-specific attacks – their practicality, their efficacy and potential low-overhead implementation changes to mitigate the underlying vulnerabilities.

The combination of free spectrum, efficient channel coding and cheap interface hardware have made 802.11-based access networks extremely popular. For a couple hundred dollars a user can buy an 802.11 access point that seamlessly extends their existing network connectivity for almost 100 meters. As a result, the market for 802.11-based LANs exceeded $1 Billion in 2001 and includes widespread use in the home, enterprise and government/military sectors, as well as an emerging market in public area wireless networks. However, this same widespread deployment makes 802.11-based networks an attractive target for potential attackers. Indeed, recent research has demonstrated basic flaws in 802.11’s encryption mechanisms [FMS01, BGW01] and authentication protocols [ASJZ01] – ultimately leading to the creation of a series of protocol extensions and replacements (e.g., WPA, 802.11i, 802.1X) to address these problems. However, most of this work has focused primarily on the requirements of access control and confidentiality, rather than availability.

In contrast, this paper focuses on the threats posed by denial-of-service (DoS) attacks against 802.11’s MAC protocol. Such attacks, which prevent legitimate users from accessing the network, are a vexing problem in all networks, but they are particularly threatening in the wireless context. Without a physical infrastructure, an attacker is afforded considerable flexibility in deciding where and when to attack, as well as enhanced anonymity due to the difficulty in locating the source of individual wireless transmissions. Moreover, the relative immaturity of 802.11-based network management tools makes it unlikely that a well-planned attack will be quickly diagnosed. Finally, as we will show, vulnerabilities in the 802.11 MAC protocol allow an attacker to selectively or completely disrupt service to the network using relatively few packets and low power consumption.

This paper makes four principal contributions. First, we provide a description of vulnerabilities in the 802.11 management and media access services that are vulnerable to attack. Second, we demonstrate that all such attacks are practical to implement by circumventing the normal operation of the firmware in commodity 802.11 devices. Third, we implement two important classes of denial-ofservice attacks and investigate the range of their practical effectiveness. Finally, we describe, implement and evaluate non-cryptographic countermeasures that can be implemented in the firmware of existing MAC hardware. The rest of this paper is structured as follows: Section 2 describes related security research conducted by others in academia, as well as unpublished, but contemporaneous, work from the “blackhat” security community.

1. **Security and Privacy in Sensor Networks**

Sensor networks offer economically viable solutions for a variety of applications. For example, current implementations monitor factory instrumentation, pollution levels, freeway traffic, and the structural integrity of buildings. Other applications include climate sensing and control in office buildings and home environmental sensing systems for temperature, light, moisture, and motion. Sensor networks are key to the creation of smart spaces, which embed information technology in everyday home and work environments. The miniature wireless sensor nodes, or

motes, developed from low-cost offthe- shelf components at University of California, Berkeley, as part of its smart dust projects, establish a selforganizing sensor network when dispersed into an environment.

The privacy and security issues posed by sensor networks represent a rich field of research problems. Improving network hardware and software may address many of the issues, but others will require new supporting technologies.

**SENSOR NODE COMPROMISE**

We expect future sensor networks to consist of hundreds or thousands of sensor nodes. Each node represents a potential point of attack, making it impractical to monitor and protect each individual sensor from either physical or logical attack. The networks may be dispersed over a large area, further exposing them to attackers who capture and reprogram individual sensor nodes. Attackers can also obtain their own commodity sensor nodes and induce the network to accept them as legitimate nodes, or they can claim multiple identities for an altered node. Once in control of a few nodes inside the network, the adversary can then mount a variety of attacks—for example, falsification of sensor data, extraction of private sensed information from sensor network readings, and denial of service.

**EAVESDROPPING**

In wireless sensor network communications, an adversary can gain access to private information by monitoring transmissions between nodes. For example, a few wireless receivers placed outside a house might be able to monitor the light and temperature readings of sensor networks inside the house, thus revealing detailed information about the occupants’ personal daily activities. Encrypting sensor node communications partly solves eavesdropping problems but requires a robust key exchange and distribution scheme. The scheme must be simple for the network owner to execute and feasible for the limited sensor node hardware to implement. It must also maintain secrecy in the rest of the network when an adversary compromises a few sensor nodes and exposes their secret keys. Ideally, these schemes would also allow revocation

of known exposed keys and rekeying of sensor nodes. The large number of communicating nodes makes end-to-end encryptionusually impractical since sensor node hardware can rarely store a large number of unique encryption keys. Instead, sensor network designers may opt for

hop-by-hop encryption, where each sensor node stores only encryption keys shared with its immediate neighbors. In this case, adversary control of a communication node eliminates encryption’s effectiveness for any communications directed through the compromised node. This situation could be exacerbated if an adversary manipulates the routing infrastructure to send many communications through a malicious node. More robust routing protocols are one solution to this problem. Another solution is multipath routing, which routes parts of a message over multiple disjoint paths and reassembles them at the destination. Efficient discovery of the best disjoint paths to use for such an operation is another research challenge.

**PRIVACY OF SENSED DATA**

Sensor networks are tools for collecting information, and an adversary can gain access to sensitive information either by accessing stored sensor data or by querying or eavesdropping on the network. Adversaries can use even seemingly innocuous data to derive sensitive information if they know how to correlate multiple sensor inputs. For example, an adversary that gains access to both the indoor and outdoor sensors of a home may be able to isolate internal noise from external noise and thus extract details about the inhabitants’ private activities. The main privacy problem, however, is not that sensor networks enable the collection of information that would otherwise be impossible. In fact, much information from sensor networks could probably be collected through direct site surveillance. Rather, sensor networks aggravate the privacy problem because they make large volumes of information easily available through remote access. Hence, adversaries need not be physically present to maintain surveillance. They can gather information in a low-risk, anonymous manner. Remote access also allows a single adversary to monitor multiple sites simultaneously. Ensuring that sensed information stays within the sensor network and is accessible only to trusted parties is an essential step toward achieving privacy. Data encryption and access control is one approach. Another is to restrict the network’s ability to gather data at a detail level that could compromise privacy. For example, a sensor network might anonymize data by reporting only aggregate temperatures over a wide area or approximate locations of sensed individuals. A system stores the sensed data in an anonymized database, removing the details that an adversary might find useful. Another approach is to process queries in the sensor network in a distributed manner so that no single node can observe the query results in their entirety. This approach guards against potential system abuse by compromised malicious nodes.

**DENIAL-OF-SERVICE ATTACKS**

As safety-critical applications use more sensor networks, the potential damage of operational disruptions becomes significant. Defending against denial-of-service attacks, which aim to destroy network functionality rather than subverting it or using the sensed information, is extremely difficult.DoS attacks can occur at the physical layer—for example, via radio jamming.

They can also involve malicious transmissions into the network to interfere with sensor network protocols or physically destroy central network nodes. Attackers can induce battery exhaustion in sensor nodes— for example, by sending a sustained series of useless communications that the targeted nodes will expend energy processing and may also forward to other nodes. More insidious attacks can occur from inside the sensor network if attackers can compromise the sensor nodes. For example, they could create routing loops that will eventually exhaust all nodes in the loop. Potential defenses against denial-ofservice attacks are as varied as the attacks themselves. Techniques such as spread-spectrum communication or frequency hopping can counteract jamming attacks. Proper authentication can prevent injected messages from being accepted by the network. However, the protocols involved must be efficient so that they themselves do not become targets for an energyexhaustion attack. For example, using signatures based on asymmetric cryptography can provide message authentication. However, the creation and verification of asymmetric signatures are highly computationally intensive, and attackers that can induce a large number of these operations can mount an effective energy-exhaustion attack.

**CONCLUSION**

we defined Vampire attacks, a new class of resource consumption attacks that use routing protocols to permanently disable ad hoc wireless sensor networks by depleting nodes’ battery power. These attacks do not depend on particular protocols or implementations, but rather expose vulnerabilities in a number of popular protocol classes. We showed a number of proof-of-concept attacks against representative examples of existing routing protocols using a small number of weak adversaries, and measured their attack success on a randomly generated topology.

**Future Enhancement**

Without fully solving the problem of malicious topology discovery, we can still mitigate it by forcing synchronous discovery and ignoring discovery messages during the intervening periods. This can lead to some nodes being separated from the network for a period of time, and is essentially a form of rate limiting. Although we rejected rate limiting before, it is acceptable here since discovery should consume a small fraction of running time compared to packet forwarding.

**System Requirement Specification**

**Hardware requirements:**

* Processor : Any Processor above 500 MHz
* Ram : 128Mb.
* Hard Disk : 10 GB.
* Compact Disk : 650 Mb.
* Input device : Standard Keyboard and Mouse.
* Output device : VGA and High Resolution Monitor

**Software requirements:**

* Operating System : Windows Family.
* Language : JDK 1.5
* Data Bases : Microsoft SQL Server
* Front End : Java Swing