ISSN:0975-9646

Attack-Resistant Routing for Wireless Ad Hoc Networks

Susan Sharon George
Department of Computer Science & Engineering
T.John Institute of Technology, Bangalore

Suma.R
Assistant Professor,
Department of Computer Science & Engineering
T.John Institute of Technology, Bangalore

***Abstract*—Ad hoc low-power wireless networks are exposed to various types of attacks at different levels of the protocol stack. A number of security services: availability, confidentiality, authentication, integrity and non-repudiation are crucial to ensure a reliable data transfer over such networks and to secure the network resources. Prior security work in this area has concentrated primarily on the DoS attack at the routing layer. This paper focuses on a more devastating, difficult to prevent, and easy to carry out attack called Vampire attacks, which quickly drain nodes' battery power leading to the permanent disabling of nodes. Majority of the traditional routing protocols fail to provide security in this scenario. This paper discusses methods to mitigate these types of attacks, by introducing a new protocol that limits the damage caused by Vampire attacks.**

***Keywords*—Denial of Service, routing, security, wireless networks, ad hoc networks**

I. INTRODUCTION

An ad hoc wireless network is a decentralized, collection of wireless mobile nodes forming a temporary network without the aid of any established infrastructure. Such networks promise exciting new applications such as habitat monitoring, troop deployment, factory performance and so on. In Wireless ad hoc networks, every node acts as a router to relay every other node's packets to enhance performance and deployment. i.e., the traffic originating from a node is usually passed through other nodes to the destination.

As wireless networks are becoming more and more crucial to the day-to-day functioning of people and business organizations, the lack of availability becomes less tolerable. The ad hoc organization of wireless ad hoc networks makes them vulnerable to denial of service (DoS) attacks. A DoS is any event that diminishes the networks' capacity to perform its expected function. Hardware failures, software bugs, resource exhaustion are some factors that contribute to the denial of service attacks [2]. A great deal of research has been done to enhance the survivability of ad hoc networks. While these schemes prevent short-term availability of network, they do not address attacks that effect long-term availability. The most permanent denial-of service attack is the resource exhaustion attack, where the nodes are entirely depleted out of its battery, leading to the permanent disabling of the network. These attacks do not flood the network with large amounts of data; instead try to transmit as little as data to achieve the largest energy drain. We call such attacks Vampire attacks, since they drain the life from network nodes.

Extensive researches have been done on power draining and resource exhaustion scenarios [6], [8]. But these focused more on the other layers of protocol stack. Resource exhaustion attacks at the routing layer are left untouched. Vampire attacks differ from the previously studied DoS attacks in that they do not disrupt immediate availability. Rather it works over time, entirely draining out the nodes' battery power, leading to the permanent disabling of the network. Moreover, Vampire attacks *exploit general properties of protocol classes* such as link-state, distance vector, source routing and geographic routing. Vampire attacks are very difficult to detect and prevent, since they *use protocol-compliant messages*.

Consider the process of routing a packet in an ad hoc wireless network. A source composes and transmits the packet to the next hop node, which in turn relays the packet further, until the packet reaches its destination. However, this multihop relaying can consume the resources at each node. So, the process of routing a packet itself leads to resource exhaustion. Further, a malicious node within the path traced by the packet can cause an increase in the energy consumption while sending the same number of messages as an honest node. Hence, we define Vampire attacks as the composition and transmission of a message that causes an increase in the cumulative energy consumption by a network than if an honest node transmitted a message of identical size to the same destination.

The contributions of this work are highlighted as follows: First, we analyze the existing forms of Vampire attacks. Second, we modify an existing secure routing protocol to resist Vampire attacks in the packet forwarding phase.

II. RELATED WORK

A variety of power draining attacks exists, that has not been defined, analyzed and mitigated at the routing layer. One among the very early forms of power exhaustion attack is the 'denial-of-sleep' attacks. As the name says, these attacks prevent nodes from entering a low-power sleep cycle, targeting a battery-powered device's power supply in an effort to exhaust this constrained resource. D.R. Raymond et al. [6] discuss the denial –of-sleep attacks at the MAC layer. This classified sensor network denial-of-sleep attacks in terms of the attacker's knowledge of the MAC layer protocol and ability to bypass authentication and encryption protocols. Additional work mentions resource exhaustion at the MAC and transport layers [9]. A.D. Wood and J.A. Stankovic [2] define

Flooding attacks and Desynchronisation attacks as the DoS attacks in the transport layer, and proposes methods to counter them. Though [8], [10] discussed the problem of routing loops, no effective defenses were suggested.

Depletion of resources such as memory, CPU time and bandwidth can easily cause problems even in non-power-constrained systems. As in classic TCP SYN flood, an adversary sends many connection establishment requests to the victim. Each request causes the victim to allocate resources, eventually running out of resources. An effective defense requires the clients to demonstrate the commitment of their own resources to each connection by solving client puzzles. An adversary must therefore be able to commit far more computational resources per unit time to flood the server with valid connections. This solution is actually a form of rate limiting and place minimal load on legitimate clients, but prevent malicious nodes who will attempt for a large number of connections.

Significant researches have been done on attacks and defenses against quality of service (QoS) degradation, or RoQ attacks. RoQ attacks produced long term degradation in network performance. As M.G. Uirguis et al. [3] says, in RoQ attacks, attacker's ultimate goal is to maximize the damage at any cost. The paper focused on attacks whose perpetrators are not focused on denying access (i.e., targeting availability), but rather they are focused on bleeding the system of its capacity. The focuses of these works were on transport layers rather than routing protocol layer, so these defenses were not applicable.

Other work on denial of service in ad hoc wireless networks has dealt with the problem of incorporating security mechanisms into routing protocols for wireless ad hoc networks [5], [11]. The focuses were on adversaries who prevent route setup, disrupt communication, or preferentially establish routes through themselves to drop, manipulate, or monitor packets.

Another area of research was on minimal-energy routing, which aims to increase the lifetime of power-constrained networks by minimizing the energy spent in transmitting and receiving messages. V. Rudoplu and T.H.Meng [4] described a distributed network protocol optimized for achieving the minimum energy for randomly deployed ad hoc networks. This paper focused on cooperative nodes and not on malicious scenarios. However, Vampires will increase energy usage even in minimal-energy routing scenarios and when power conserving MAC protocols are used. Attackers will produce packets which traverse more hops than necessary, so even if nodes spend the minimum energy required to transmit the packets, each packet is still more expensive to transmit in the presence of Vampires.

In a path-based DoS attack, an adversary overwhelms sensor nodes a long distance away by flooding a multihop-end-to-end communication path with either replayed packets or injected spurious packets. Deng et al. discuss PDoS attacks and defenses [7], proposing a solution using one-way hash chains to protect end-to-end communications in WSNs against PDoS attacks, limiting the rate at which nodes can transmit packets. While this strategy may protect against traditional DoS, it does not protect against intelligent adversaries who use a small number of packets or do not originate packets at all.

Another form of a path based attack is the Wormhole attack, a severe attack in ad hoc networks that is particularly difficult to prevent against. First introduced in [12], it allows two non-neighboring malicious nodes to emulate a neighbor relationship, even in secure routing systems. These links are not made visible to other network members, but can be used by the colluding nodes to privately exchange messages. In the wormhole attack, an attacker record packets at one location in the network, tunnels them to another location, and retransmits them there into the network. Though the authors propose a new, general mechanism, called packet leashes, for detecting and thus defending against wormholes, their solution is not always acceptable.

III. VAMPIRE ATTACKS: AN OVERVIEW

This section analyses two of the increasingly damaging Vampire attacks: Carousel and Stretch attacks. We have several protocol classes such as source routing, distance vector, link-state, geographic routing and so on. In source routing, to send a packet to another host, the sender constructs a source route in the packet's header, giving the address of each host in the network through which the packet should be forwarded in order to reach the destination host. In this case, a malicious source can specify a source route through the network that traverses more hops than optimal, draining energy from the intermediate nodes who forward the packet based on the source route. In routing schemes, were forwarding decisions are made independently by each node, directional antenna and wormhole attacks can be used to deliver packets to remote locations. This forces packet processing at all nodes that would not normally receive packet, causing increased energy expenditure at each node.

Our first attack, called the carousel attack, targets source routing protocols, exploiting the limited verification of message headers at each forwarding node. Here, a malicious node composes and transmits packets with purposely introduced routing loops. It sends packets in circles, hence the name. Carousel attack causes a single packet to repeatedly traverse the same set of nodes, depleting the nodes' battery power. As Fig. 3a shows, a malicious packet introduces routing loops, makes its way twice around the loop before delivering it to the sink. This makes the packet repeatedly traverse the same set of nodes, while a honest loop passes the packet directly from E to sink.

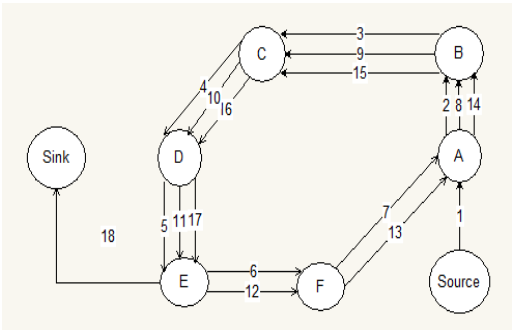

Fig. 3a. Carousel attack

The second attack, Stretch attack, the malicious packet traverses potentially all parts of the network. It also targets source routing. Here, an adversary constructs artificially long route, making the packet traverse all nodes of the network. We call this stretch attack, since it increases the packet length, causing the packets to be processed by a number of nodes, regardless of the hop count along the shortest path between the adversary and packet destination. Fig. 3b shows an example of stretch attack. Honest route is made solid. The last link to the sink is shared.

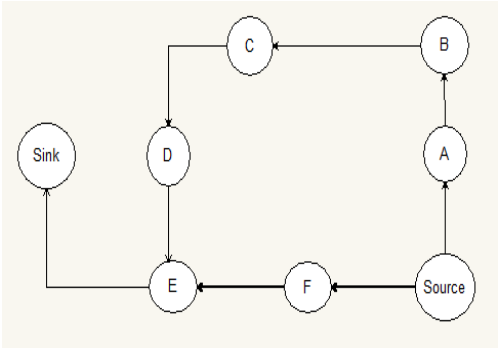

Fig. 3b. Stretch attack

Carousel attack can be easily defended with negligible overhead, while stretch attack is challenging. The first protection mechanism to consider is loose source routing, where any node can reroute the packet if it knows a shorter path to the sink. Unfortunately, this proves to be less efficient. The second protection mechanism is to modify the clean slate sensor network routing protocol [8], to assure that the packet consistently makes progress to the destination. This limits the adversarial success through a limited topology discovery period, followed by a long packet forwarding phase.

IV. CLEAN-SLATE SECURE NETWORK ROUTING

A clean-slate secure network routing protocol by Parno, Luk, Gaustad and Perrig (PLGP) [8] can be modified to limit the damage caused by Vampire attacks. All routing protocols employ at least one topology discovery phase, since ad hoc deployment implies no prior position knowledge. PLGP consists of a topology discovery phase followed by a packet forwarding phase. The discovery phase is repeated periodically to ensure that the topology information stays current. Discovery deterministically organizes nodes into a tree that will be later used as an addressing scheme.

Topology discovery. Initially every node comprises its own group, i.e., the node knows only itself. Then, the discovery repeatedly merges group of nodes into larger size. A group G merges with the smallest neighboring group G'. After each merge, the nodes in group G adds a bit to their network addresses to differentiate themselves from the nodes in group G'. Discovery begins with every node announcing its presence broadcasts that include its ID and the accompanying certificate of identity (assigned before network deployment) to its neighbors. Nodes, who overhear this presence broadcast, verify this certificate and add that node to their neighbor lists. Each node starts with its own group of size one, with a virtual address 0. When two individual nodes form a group of size two, one of them takes the address 0, and the other becomes 1. Similarly, when groups merge, each group will choose either 0 or 1. Each group member prepends the group address to their own address. In this way, each time two groups merge, the address of each node is lengthened by 1 bit. The resulting network addresses forms a binary tree of all addresses in the network stopping when the entire network is a single group.

At the end of discovery, each node should compute the same address tree as the other nodes. All leaf nodes in the tree correspond to physical nodes in the network. Each node stores the ID of one or more nodes through which it heard a broadcast that another group exists. So, every node within a group has a next-hop path to every other group. When discovery terminates, all nodes learn each others' virtual address and cryptographic keys.

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 bits of its address that differs from the message originator's address. When a node with network address *N* receives a message destined for address *D*, it finds the most significant digit between *D* and *N* that differs and sends the message towards the other group at the corresponding level. For example, if *D* = 0.1.0 and *N* = 0.0.1, then the fact that *D* and *N* share the same identifier (0) in the most significant bit of their addresses implies that they are in the same level 3 group. However, the next bit reveals that within that group, *D* resides in the level 2 group with a 1 identifier, while *N* resides in the level 2 group with a 0 identifier. Thus, *N* will forward the packet towards the level 2 group with a 1 identifier.

V. SECURITY AGAINST VAMPIRE ATTACKS

PLGP in its original is vulnerable to Vampires. In PLGP, the forwarding nodes do not know what path a packet took, allowing adversaries to divert packets to any part of the network. Hence, we modify the forwarding phase of PLGP to avoid Vampires. For this, we introduce a property, *no-backtracking property*, which is satisfied for a given packet if and only if it consistently makes progress towards its destination in the logical network address space. Formally, we can state that*: No-backtracking is satisfied if every packet p traverses the same number of hops whether or not an adversary is present in the network.* i.e., no-backtracking implies that the number of honest nodes traced by a packet sent from source to sink is independent of the actions of malicious nodes.

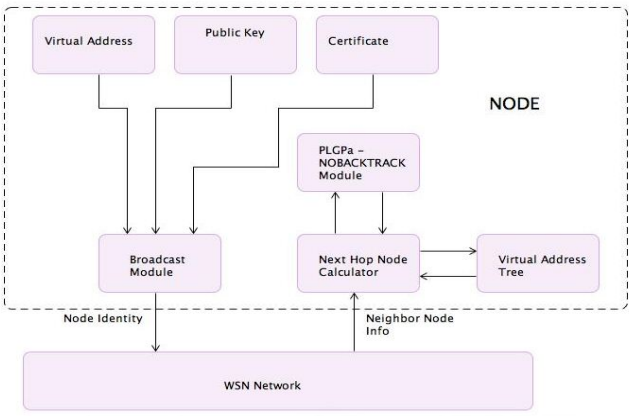

Fig. 5a. System Architecture of PLGPa

In source routing protocols, intermediate nodes in the source route cannot specify whether the source-defined path is optimal. When the forwarding decisions are made independently by each node, packets cannot contain maliciously composed routes. However, a clever adversary can still influence packet progress. We can prevent this by independently verifying packet progress. Hence, to preserve no-backtracking, we add a verifiable path history to every 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. Whenever a node forwards a packet, it does so by attaching its nonreplayable signature. Any node receiving this packet, verifies the chain of attestation to verify the path traced by the packet and to ensure that the packet is making progress towards its destination. The function secure_forward_packet gives the modified protocol.

Further more, to detect and drop duplicate packets, each node is assigned a local storage similar to a buffer that stores a hash of various packets received by that node. This helps to minimize the energy spent in verifying the attestation chain by each node. The buffer holds a hash of messages that has been through that node before, and having matched any of the incoming packets to the one already in the buffer, will drop it. This helps to figure out duplicate packets in a second and drop them.

When a node receives a message, it first compares the hash of that message to the ones already in the buffer. Having matched to any one already in the buffer, the node will immediately drop it. Else, it stores that hash value in its buffer. Then, it verifies the path attestation to see that every node in the attestation 1) has a corresponding entry in the signature chain 2) is logically closer to the destination. Since buffers makes sure that no duplicate packets exist in the network, and no-backtracking guarantees packet progress, and PLGPa preserves no-backtracking, it is the only protocol that bounds the damage caused by vampire attacks.

VI. EXPECTED RESULTS

PLGPa includes path attestations, increasing the size of every packet, incurring penalties in terms of bandwidth use. Adding extra packet verification requirements for intermediate nodes also increases processor utilization, requiring time and additional power. In the presence of even a small number of malicious nodes, the increased overhead becomes worthwhile when considering the potential damage of Vampire attacks. The additional bandwidth of our attestation scheme is minimal, as chain signatures are compact. The incorporated local storage increases performance from 10- 20 percent as the energy expenditure for cryptographic operations can be avoided for duplicate packets, detected by buffers.

VII. CONCLUSION

This paper discusses a more devastating form of DoS attacks called Vampire attacks. Vampire attacks targets on depleting a nodes' battery power, leading to the permanent disabling of the node, and gradually the network. We analyzed two of such attacks: carousel and stretch attacks. We also modified an existing sensor network routing protocol to bound the damage caused by Vampires in the forwarding phase.

REFERENCES

[1] Eugene Y. Vasserman and Nicholas Hopper, "Vampire Attacks: Draining Life from Wireless Ad Hoc Sensor Networks" , *IEEE Mobile Computing*, vol 12, February 2013.

[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] M. Guirguis, A. Bestavros, I. Matta, and Y. Zhang, "Reduction of Quality (RoQ) Attacks on Internet End-Systems," Proc. IEEE INFOCOM, 2005.

[4] 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.

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

[6] 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.

[7] 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.

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

[9] D.R. Raymond and S.F. Midkiff, "Denial-of-Service in Wireless Sensor Networks: Attacks and Defenses," IEEE Pervasive Computing, vol. 7, no. 1, pp. 74-81, Jan.-Mar. 2008.

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

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

[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.