IJCSE International Journal of Computer Science and Engineering Open Access
Research Paper Volume-2, Issue-4 E-ISSN: 2347-2693

Defending Against Source Routing Attacks and Packet Forwarding in Ad-Hoc Networks

Sriharibabu.Kolla

Dept of Computer Science, Dhanekula Institute of Engg & Technology, India

www.ijcseonline.org

Received: 19/04/2014 Revised: 28 /04/2014 Accepted: 17/05/ 2014 Published: 31 /05/2014

Abstract- A wireless Ad-hoc network is the collection of wireless nodes that can co-operate by forwarding packets for each other to allow nodes to communicate directly. The prominence of the place of wireless ad-hoc networking is become more crucial to everyday functioning of people. This is due to obvious advantages of the wireless networks with ubiquity access and minimal hardware requirements in networks. The ad-hoc nature of sensor network means no structure can be statically defined. The network topology is always subject to change due to node failure, addition or mobility. Since nodes may fail or be replaced the network must support self-configuration. Routing and data forwarding is a crucial service for enabling communication in ad-hoc networks. Unfortunately, current routing protocols suffer from many security vulnerabilities. In this paper prior security work in this area focused primarily on source routing attacks. Low power wireless networks are an existing research direction in routing and packet forwarding.

Keywords: Security, Ad-Hoc Networks, Vulnerabilities, Wireless Networks, Sensor Networks, Resource Exhaustion Attacks, Route Disruption Attacks.

I. INTRODUCTION

A secure wireless Adhoc network is a challenging task. Adhoc network is a collection of nodes. In which individual nodes co-operate by forwarding packets. An Adhoc network assumes that every node is also a router that can forward packets. As consequence, when two nodes are communicating all nodes in the vicinity of them must remain silent for the duration of communication. Sensor networks provide economically viable solutions for a wide variety of applications. An Adhoc network assumes that every node is also a router that can forward messages. The deployment of wireless nodes where there is no infrastructure or the local infrastructure is not reliable can be difficult. Adhoc networks have been proposed in order to solve such problems. The main advantage of Adhoc networks are flexibility, low-cost and robustness. Adhoc networks can be easily setup, even in desert places and can endure to natural catastrophes and war.

The design of a wireless Adhoc network has to take into several interesting and difficult problems. Wireless Adhoc networks particularly vulnerable to DOS attacks [5], these makes secure routing difficult task, because a malicious node can easily join the network and modify of fabricate routing information and impersonating other networks. An unprotected Adhoc routing is vulnerable to these types of attacks. Attacks on Adhoc network routing protocols generally fall into two types. That are-

- Routing disruption attacks.
- Resource exhaustion attacks.

These attacks are distinct to all Dos attacks. Reduction of Quality (ROQ), routing infrastructure attacks they do not disturb immediate availability, but rather work overtime to entirely deplete network.

The contribution of this work is, to design a general purpose of secure routing protocol. Ultimate goal is to design a highly secure and minimize resource utilization in packet forwarding. Before going to implement new secure source routing protocol, we thoroughly evaluate the several existing secure routing protocols to prevent these attacks such as Ariadne [1], SAODV [2], and SEAD [3], do not defend against these attacks. Finally we will give solution to provably damage from these attacks during packet forwarding.

A. CLASSIFICATION

The first challenge in addressing source routing attacks is defining them – what actions in fact constitute an attack? Dos attacks in wired networks are frequently characterized by amplification [4, 5]. An adversary can amplify the resources it spends on the attack, e.g., use one minute of its own time to cause the victim to process to use ten minutes to process. However, consider the process of routing a packet in any multi-hop network. A source composes and transmits it to the next hop toward the destination, which transmits it to the next hop toward the destination, which transmits it further, until the destination is reached. It causes to consuming resources not only at the source node but also at every node the message moves through. If we consider the cumulative energy of an entire network, amplification attacks are always possible, given that an adversary can compose and send messages which are proposed by each node along the message path. So, the act of sending a message is in itself an

act of amplification, leading to resource exhaustion, as long as the aggregate cost of routing a message is lower than the cost to the source to compose and transmit it. So, we must drop amplification as our definition of maliciousness and instead focus on the cumulative energy consumption increase that a malicious node can cause while sending the same number of messages as an honest node.

These both attacks can be categorized under source routing attacks. We define source routing attacks as the packet composition and transmitting that causes more resources can be consumed by the network than if an honest node transmitted the packet of identical size to the same destination, although using different packet headers. Let us measure the strength of the attack by the ratio of network energy used in the benign case to the energy used in the malicious case, i.e. the ratio of network-wide power utilization with malicious nodes present to energy usage with only honest nodes when the number and size of packets sent remains constant. Safety from source routing attacks implies that this ratio is 1. Energy used by malicious nodes is not considered, since they can always unilaterally drain their own batteries.

B. OVERVIEW

In the remainder of this paper, we resent a series of increasingly damaging source routing 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 that forward the packet based on the included source route. In routing protocols where forwarding decisions are made independently by each node, we suggest how directional antenna and wormhole attacks [6] 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 the disadvantages of existing protocol against source routing protocols. In the existing protocol 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.

C. ATTACKS ON SOURCE ROUTING PROTOCOLS.

In this section we present simple but previously omitted attacks on source routing protocols such as DSR [8]. In this protocol, the entire route to destination within the packet header, so intermediates do not make independent forwarding decisions, relying rather on a route specified by the intermediate node finds itself in the route and transmits the message to next hop. The burden is on the source to ensure that the route is valid at the time of sending, and that every node in the route is a physical neighbor of the previous route hop. This approach has advantage of requiring very little forwarding logic at intermediate nodes and allows for entire routes to be sender authenticated using digital signatures, as in Ariadne [1].

We evaluated both the route disruption and resource exhaustion attacks in randomly generated topology. Energy usage for the minimum number of packets required to deliver a single message, so sending more messages increases the strength of the attack linearly until bandwidth constraint. We independently computed resource utilization of honest and adversary nodes found that adversary nodes did not use a disproportionate amount of energy in carrying out the attack.

Route disruption attack [12]:
In this attack, an adversary sends a packet with a route composed as a series of loops (as shown in fig (1)), such that the same node appears in the route many times. These strategies 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.

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 to the success of this attack.

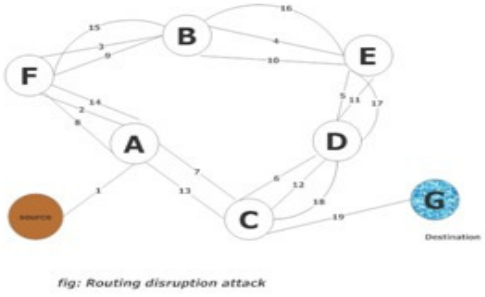

Fig (1): Routing disruption attacks

Resource exhaustion attack [12]: In this attack, an adversary node constructs artificially long source routes (as shown in fir (2)), causing packets to traverse a larger than optimal number of nodes. An honest source would select the route while the latter uses energy at the nodes that were already in the honest path, the former extends the consumed energy "equivalence lines" to a wider section of the network. Energy usage is less localized around the original path, but more total energy is consumed.

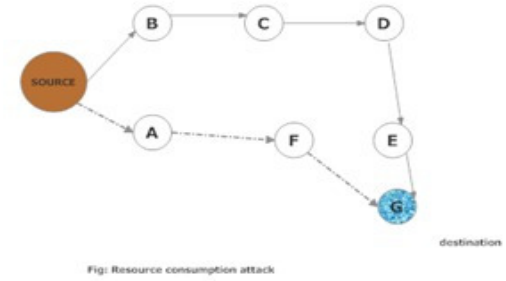

Fig (2): resource exhaustion attack

A. MITIGATION METHODS

In our first attack Route disruption attack, can be prevented entirely by having forwarding nodes check source routes for loops. While this adds extra forwarding logic and thus more overhead, we can except the gain to be worthwhile in malicious environments. When a loop is detected, the source route could be corrected and the packet sent on, but one of the attractive features of source routing is that the route can itself be signed by source [1]. Therefore, it is better to simply drop the packet, especially considering that the sending node is likely malicious (honest nodes should not introduce loops). An alternative solution is to alter how intermediate nodes process the source route. To forward a message, a node must determine the next hop by locating itself in the source route. If a node searches for itself from the destination backward instead from the source forward, any loop that includes the current node will be automatically truncated[1]. No extra processing is required for this defense, since a node must perform this check anyway; we only alter the way the check is done.

In our second attack, Resource exhaustion attack is more crucial and challenging to prevent. Its success rests on the forwarding node. If we call the no optimization case "strict" source routing, since the route is followed exactly as specified in the header, we can define loose source routing, where the intermediate nodes may replace part or the entire route in the packet header if they know of a better route to the destination. This makes it necessary for nodes to at least some fraction of other nodes, partially defeating the as-needed discovery advantage. Moreover, caching must be done carefully lest a maliciously suboptimal route be introduced.

B. OTHER POSSIBLE ATTACKS

Directional antenna attack: Source routing attacks have little control over packet progress when forwarding decisions are made independently by each node, but they can still waste energy by restarting a packet in various parts of the network. 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.

Malicious discovery attack: Another attack on all previously mentioned-routing protocols is spurious route discovery. in most protocols, every node will forward route discovery packets meaning it is possible to initiate a flood by sending a single message. Systems that perform as needed route discovery, such as AODV, PLGP and DSR, particularly vulnerable, since nodes may legitimately initiate discovery at any time, not just during a topology change.

II. CLEAN SLATE SENSOR METWORK ROUTING.

This protocol is introduced by Parno, Luk, Gaustad and Perrig. PLGP [7] can be modified to provably resist to several attacks like vampire attacks during the packet forwarding phase. The original version of the protocol, although designed for security, is vulnerable to attacks. PLGP consists a topology discovery phase and followed by a packet forwarding phase. At the end of discovery, each node should compute the same address tree as other nodes. All leaf nodes in the tree are physical nodes in the network, and their virtual addresses correspond to their position in the tree. All nodes learn each other's virtual addresses and cryptographic keys. The final address tree is verifiable after network convergence, and all forwarding decisions can be independently verified.

Topology discovery: discovery begins with a time-limited period during which every node must announce its presence by broadcasting a certificate of identity, including its public key, signed by a trusted offline authority. Each node starts as its own group of size one, with a virtual address is zero. Nodes who over hear presence broadcasts from groups with their neighbors. When two individual nodes (each with an initial address 0) form a group of size two, one of them takes the address 0, and the other becomes 1. Groups merge preferentially with the smallest neighboring group, which may be a single node. We may think of groups acting as individual nodes, with decisions made using secure multiparty computation. Like individual nodes, each group will initially choose a group address 0, and will choose 0 or 1 when merging with another group. Each group member pretends the group address to their own address, e.g. node 0 in group 0 becomes 0.0, and node 0 in group 1 becomes 1.0, and so on. Each time two groups merge, the address of each node is lengthened by one bit. Implicitly, this forms a binary tree of all addresses as leaved. Note that this tree is not a virtual coordinate system, as the only information coded by the tree is neighbor relationships among nodes.

Nodes will request to join with the smallest group in their vicinity, with ties broken by group IDs, which are computed cooperatively by the entire group as a deterministic function of individual number IDs. When larger groups merge, they both broadcast their group IDs to each other, and proceed with a merge protocol identical to the two- node case. Groups that have grown large enough that some members are not within radio range of other groups will communicate through "gateway nodes," which are within range of both groups. Each node stores the identity of one or more nodes through which it heard an announcement that another group exists. That node may have itself heard the information second-hand, so every node within a group will end up with a next-hop path to every other group, as in distance-vector. Topology discovery proceeds in this manner until all network nodes are members of a single group. By the end of topology discovery, each node learns every other node's virtual address, public key and certificate since every group members knows the identities of all other group members and the network converges to a single group.

[1] The last instance of the local node will be found in the source route rather than the first.

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 shortens the logical distance to the destination, since node addresses should be strictly closer to the destination.

The original version of PLGP does not satisfy no-backtracking. So this protocol is modified to preserve no backtracking, it add a verifiable path history to every PLGP packet, similar to route authentications in Ariadne [1] and path vector signatures in [9]. 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. Whenever node 'n' forwards a packet 'p'. It attaches a non-replay able attestation (signature). 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 travelled away from its destination in the logical address space.

PLGPa protocol is modified from PLGP. To show that modified protocol preserves the no-backtracking property, it define a network as a collection of nodes, a topology, connectivity properties, and node identities, borrowing the model used by Poturalski et al. in [10]. Honest nodes can broadcast and receive messages, while malicious nodes can also use directional antennas to transmit to any node in the network without being overhead by any other node.

Honest node can compose, forward, accept or drop messages, and malicious nodes can also arbitrarily transform them. Our adversary is assumed to control 'm' nodes in an 'n; node network and has perfect knowledge of the network topology. Finally, the adversary cannot affect connectivity between any two honest nodes.

Since all messages are signed by their originator, messages from honest nodes cannot be arbitrarily modified by malicious nodes wishing to remain undetected. Rather, the adversary can only alter packet fields that are changed en route (and so are not authenticated), so only the route attestation field can be altered, shortened, or removed entirely. To prevent truncation, which would allows attacks to hide the fact that they are moving a packet away from its destination, PLGPa protocol uses Sexena and Soh's one-way signature chain construction [11], which allow nodes to add links to an existing signature chain, but not remove links, making attestations append-only. For the purposes of source routing attacks, we are unconcerned about packets with arbitrary hop counts that are never received by honest nodes but rather are routed between adversaries only, so we define the hop count of a packet as follows.

When any node receives a message, it checks that every node in the path attestation 1) has a corresponding entry in the signature chain, and 2) is logically closer to the destination than the previous hop in the chain. This way, forwarding nodes can enforce the forward progress of a message, preserving no-backtracking. If no attestation is present, the node checks to see if the originator of the message is a physical neighbor. Since messages are signed with the originator's key, malicious nodes cannot falsely claim to be the origin of a message, and therefore do not benefit by removing attestations.

III. BACK DROPS IN EXISTING PROTOCOLS

Since no-backtracking guarantees packet progress, and PLGPa preserves no-backtracking, it is the only protocol discussed so far that provably bounds the ratio of energy used in the adversarial scenario to that used with only honest nodes to 1, and by the definition of no-backtracking PLGPa resists source routing attacks. This is achieved because packet progress is securely verifiable. Note that we cannot guarantee that a packet will reach its destination, since it can always be dropped.

PLGPa includes path attestations, increasing the size of every packet, incurring penalties in terms of bandwidth use, and thus radio power. Adding extra packet verification requirements for intermediate nodes also increases processor utilization, requiring time and additional power.

Energy expenditure for cryptographic operations at intermediate hops is, unfortunately, much greater than transmit or receive overhead, and much more dependent on the specific chipset used to construct the sensor in total, the overhead on the entire network of PLGPa (over PLGP) when using 32-bit processor or dedicated cryptographic accelerator is the energy equivalent 90 additional bytes per packet.

Since PLGP offers the change to detect active source routing attacks once the network converges, successive rediscovery periods become safer. This is more than can be said of other protocols, where malicious behavior during discovery may go undetected, or at least unpunished. However, the bound we can place on malicious discovery damage in PLGPa is still unknown. Moreover, if we can conclude that a single malicious node causes a factor of 'k' energy increase during discovery; it is not clear how that value scales under collusion among multiple malicious nodes. Without fully solving the problem of malicious topology discovery, we can still omit it by forcing synchronous discovery and ignoring discovery messages during the intervening periods.

IV. *CONCLUSION*

In this paper I mentioned new attacks in source routing mechanisms. These attacks do not depend on particular protocols or implementations, but rather expose

vulnerabilities in a number of popular protocol classes. Existing protocol defenses against forwarding-phase attacks and described PLGPa, it is the first sensor network routing protocol that provably bounds damage from source routing attacks by verifying that packets and appends links to every PLGP packets. This protocol is not offered a fully satisfactory solution for source routing attacks during the topology discovery phase. Securing ad-hoc networks is still an open issue. In future I would like to implement a new source routing protocol to defense all source routing attacks and ultimate goal is to implement secure protocol as SRPF (Secure source-Routing and Packet Forwarding). This paper will hopefully motivate future researchers to come up with smarter security and make network safer.

ACKNOWLEDGEMENT

Most of all, I shall give glory, honor and thank to my family for their pray. I am very pleased of Professor Mr. B B K PRASAD, who spread no effort to ensure that I have everything I needed.

REFERENCES

[1]. Yih-Chun Hu, Adrian Perrig, and David B. Johnson, Ariadne: A secure on-demand routing protocol for ad hoc networks, MobiCom, 2002.

[2]. Yih-Chun Hu, David B. Johnson, and Adrian Perrig, SEAD: secure efficient distance vector routing for mobile wireless ad hoc networks, IEEE workshop on mobile computing systems and applications, 2002.

[3]. Manel Guerrero Zapata and N. Asokan, Securing ad hoc routing protocols, WiSE, 2002.

[4]. Kihong Park and Heejo Lee, On the effectiveness of probabilistic packet marking for IP traceback under denial of service attack, INFOCOM, 2001.

[5]. Vern Paxson, An analysis of using reflectors for distributed denial-of service attacks, SIGCOMM Comput. Commun. Rev. 31 (2001), no. 3.

[6]. Packet leashes: A defense against wormhole attacks in wireless ad hoc networks, INFOCOM, 2003.

[7]. Bryan Parno, Mark Luk, Evan Gaustad, and Adrian Perrig, Secure sensor network routing: A clean-slate approach, CoNEXT, 2006.

[8]. David B. Johnson, David A. Maltz, and Josh Broch, DSR: the dynamic source routing protocol for multihop wireless ad hoc networks, Ad hoc networking, 2001.

[9]. Lakshminarayanan Subramanian, Randy H. Katz, Volker Roth, Scott Shenker, and Ion Stoica, Reliable broadcast in unknown fixed-identity networks, Annual ACM SIGACT-SIGOPS symposium on principles of distributed computing, 2005.

[10]. Marcin Poturalski, Panagiotis Papadimitratos, and Jean-Pierre Hubaux, Secure neighbor discovery in wireless networks: Formal investigation of possibility, ACM ASIACCS, 2008.

[11]. Amitabh Saxena and Ben Soh, One-way signature chaining: a new paradigm for group cryptosystems, International Journal of Information and Computer Security 2 (2008), no. 3.

[12]. A Survey of Source Routing Protocols, Vulnerabilities and Security in Wireless Ad-hoc Networks, International journal of Computer Sciences and Engineering, 2014.

AUTHORS PROFILE

SRIHARI BABU.KOLLA is perusing Masters' degree in computer science and engineering, JNTU KAKINADA. He published several theses on various publications. His research interested in network security, privacy and anonymity, low-power, security for sensor networks and mobile applications.

