IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727
PP 59-64
www.iosrjournals.org

Power Draining Vampires in Wireless Ad hoc networks: Detection and Prevention

Anoopa S[1], Jayaprabha P[2]

[1] *(Assistant Professor, Department of Information Technology,College of engineering Perumon, India)*
[2] *(Associate Professor, Department of Information Technology,Cochin University College of Engineering Kuttanadu, India)*

Abstract: *Ad-hoc low-power wireless networks are the most promising research direction in sensing and pervasive computing. Prior security work in this area has focused primarily on denial of service at the routing or medium access control levels. Earlier, the resource depletion attacks are considered only as a routing problem, very recently these are classified in to a new group called vampire attacks. This thesis work explores the identification of resource depletion attacks at the routing protocol layer and in the application layer, which permanently disable networks by quickly draining nodes' battery power. These Vampire attacks are not specific to a particular protocol, but rather rely on the properties of many popular classes of routing protocols. It is clear 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. In this work a detection and control strategy is proposed for these vampire attacks, along with a secure packet forwarding mechanism, which will save Ad-hoc wireless nodes from power drainage due to vampire packets.*

Keywords: *Vampires in networks, Wireless Ad-Hoc network power drainage, PLGPa algorithm, vampire detection etc.*

I. Introduction

1.1 Background Ad-hoc mode is a method for wireless devices to directly communicate with each other. Operating in ad-hoc mode allows all wireless devices within range of each other to discover and communicate in peer-to-peer fashion without involving central access points. A wireless ad hoc network is a decentralized type of wireless network. The network is Ad-hoc because it does not rely on a pre existing infrastructure, such as routers in wired networks or access points in managed (infrastructure) wireless networks. Instead, each node participates in routing by forwarding data for other nodes, so the determination of which nodes forward data is made dynamically on the basis of network connectivity. An Ad-hoc network tends to feature a small group of devices all in very close proximity to each other. Performance suffers as the number of devices grows, and a large Ad-hoc network quickly becomes difficult to manage. Ad-hoc networks cannot bridge to wired LANs or to the Internet without installing a special-purpose gateway. In addition to the classic routing, Ad- hoc networks can use flooding for forwarding data.

A mobile ad-hoc network (MANET) is a self-configuring infrastructure less network of mobile devices connected by wireless. Ad hoc is Latin and means "for this purpose”. Each device in a MANET is free to move independently in any direction, and will therefore change its links to other devices frequently. Each must forward traffic unrelated to its own use, and therefore be a router. The primary challenge in building a MANET is equipping each device to continuously maintain the information required to properly route traffic. Such networks may operate by themselves or may be connected to the larger Internet.

A mobile ad hoc network has attracted a lot of attention due to the popularity of mobile devices and the advances in wireless communication technologies. A MANET is a peer-to-peer multihop mobile wireless network that has neither a fixed infrastructure nor a central server. Each node in a MANET acts as a router, and communicates with each other.A large variety of MANET applications have been developed. For example, a MANET can be used in special situations, where installing infrastructure may be difficult, or even infeasible, such as a battlefield or a disaster area. A mobile peer-to-peer file sharing system is another interesting MANET application.

MANETs are a kind of wireless ad hoc networks that usually has a routable networking environment on top of a Link Layer ad hoc network. The growth of laptops and 802.11/Wi-Fi wireless networking has made MANETs a popular research topic since the mid 1990s. Different protocols are then evaluated based on measure such as the packet drop rate, the overhead introduced by the routing protocol, end-to-end packet delays, network throughput etc.

Characteristics- A wireless ad-hoc network is a collection of mobile devices equipped with a transmitter and receiver, connected in the absence of fixed infrastructure. Wireless ad-hoc network is defined with characteristics such as purpose-specific, autonomous and dynamic. In comparison with fixed wireless networks, there is no master slave relationship that exists in a wireless ad-hoc network. Nodes rely on each other to established communication, thus each node acts as a router. Therefore, in a wireless ad-hoc network, a packet can travel Each node is responsible to forward packet to other nodes in the networks. The nodes are also collaborate themselves to implement network routine functions such as security

The major characteristics of wireless adhoc networks are:

Power consumption constrains for nodes using batteries, Ability to cope with node failures ,Mobility of nodes, Dynamic network topology, Communication failures, Ability to withstand complex environmental conditions, Ease of use, Deeply distributed architecture.

1.2 ***Problem Objective***

The objective of this paper is to create a secure mechanism which detects the vampire packets and prevents the forwarding of vampire packets and the formation of such type of packet inside the node**.**

1.3. Problem Motivation

The life of the wireless adhoc network, especially that of sensor network depends on its node's battery power. In most of the applications, battery recharging or replacing is impossible. Power drainage will leads to the failure of the node and it will affect the network also. Data loss will also occur. Therefore an efficient energy utilization scheme is required, that is, data packets should be transmitted by using minimum units of energy. But some malicious packets called vampire packets may consume more energy for packet forwarding than that of honest packet forwarding .This will lead to power drainage of node and network failure. If we can find and avoid these type of vampire packets, then we can increase the life of the node and thereby the network. It will be very crucial in many of the situations and it will increase the wide acceptability of adhoc wireless networks in many crucial applications.

1.4 Literature Review

Eugene Y. Vasserman and Nicholas Hopper [2] introduced a definition for vampire attacks in february 2013._Vampire attacks are clearly defined in their study. The study makes three primary contributions. First evaluates the vulnerabilities of existing protocols to routing layer battery depletion attacks. The security measures to prevent Vampire attacks are orthogonal to those used to protect routing infrastructure, and so existing secure routing protocols such as Ariadne, SAODV , and SEAD 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 using existing valid network paths and protocol compliant messages. The authors proposed defenses against some of the forwarding-phase attacks and described PLGPa, the first sensor network routing protocol that provably bounds damage from Vampire attacks by verifying that packets consistently make progress toward their destinations. They have not offered a fully satisfactory solution for Vampire attacks during the topology discovery phase, but suggested some intuition about damage limitations possible with further modifications to PLGPa. They concentrated only in the network layer but not considered about the application layer where the malicious packet actually originates. Gergely Acs, Levente Buttyan, and Istvan Vajda [6] had introduced a new attack on Ariadne , a previously published "secure" routing protocol. These attacks clearly demonstrate that flaws can be very subtle, and therefore, hard to discover by informal reasoning. Hence, they 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.

Imad Aad, Jean-Pierre Hubaux, and Edward W. Knightly[5], mainly focuses on the design and study DoS attacks in order to assess the damage that difficult-to-detect attackers can cause. The authors presented a novel DoS attack perpetrated by JellyFish: relay nodes that stealthily disorder, delay, or periodically drop packets that they are expected to forward, in a way that leads astray end-to-end congestion control protocols. This attack is protocol- compliant and yet has a devastating impact on the throughput of closed-loop flows, such as TCP flows and congestion-controlled UDP flows. For completeness, they have also considered a well known attack, the Black Hole attack, as its impact on open-loop flows is similar to the effect of JellyFish on closed-loop flows.

Bryan Parno, Mark Luk, Evan Gaustad, and Adrian Perrig [3] introduce a secure routing protocol for Ad-hoc wireless networks. The deployment of sensor networks in security- and safety-critical environments requires secure communication primitives. In this study, the authors design, implement, and evaluate a new secure routing protocol for sensor networks. The protocol presented in this paper requires no special hardware and provides message delivery even in an environment with active adversaries. They adopt a clean-slate approach and design a new sensor network routing protocol with security and efficiency as central design parameters. The protocol is efficient yet highly resilient to active attacks.

An article of computer communications [4] , 29(2006),no 2, describes an INtrusion-tolerant routing protocol for wireless SEnsor NetworkS (INSENS). INSENS constructs forwarding tables at each node to facilitate communication between sensor nodes and a base station. INSENS does not rely on detecting intrusions, but rather tolerates intrusions by bypassing the malicious nodes. An important property of INSENS is that while a malicious node may be able to compromise a small number of nodes in its vicinity, it cannot cause widespread damage in the network.

Jae-Hwan Chang and Lindros Tassiulas[7] had extended the maximum lifetime routing problem to include the energy consumption at the receivers during reception._In wireless sensor networks where nodes operate on limited battery energy, the efficient utilization of the energy is very important. One of the main characteristics of these networks is that the transmission power consumption is closely coupled with the route selection. The energy efficiency has been considered in wireless adhoc network routing, but the conventional routing objective was to minimize the total consumed energy in reaching the destination.

1.5 Problem Definition

Wireless ad-hoc networks are particularly vulnerable to denial of service (DoS) attacks due to their ad-hoc organization, and a great deal of research has been done to enhance survivability. Vampire attack can be defined as the composition and transmission of a message that causes more energy to be consumed by the network than if an honest node transmitted a message of identical size to the same destination, although using different packet headers[2]. We can 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 Vampire attacks implies that this ratio is 1. Energy use by malicious nodes is not considered, since they can always unilaterally drain their own batteries[2].

There are two important types of vampire attacks. In our first attack, an adversary composes packets with purposely introduced routing loops. We call it the carousel attack, it targets source routing protocols by exploiting the limited verification of message headers forwarding nodes, allowing a single packet to repeatedly traverse the same set of nodes[1].

In our second type of 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 [1].

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 assume that

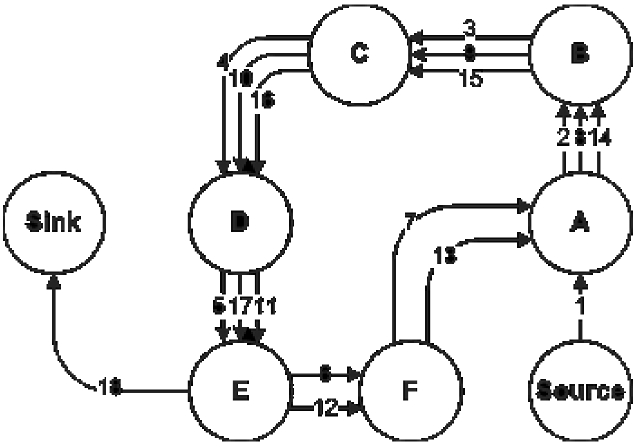

Fig. 1.1 An honest route would exit the loop immediately from node E to sink, but a malicious packet makes its way around the loop twice more before exiting.

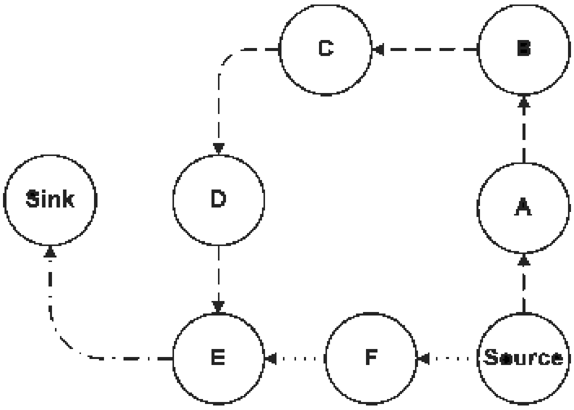

Fig 1 Honest route is dotted while malicious route is dashed. The last link to the sink is shared.

only messages originated by adversaries may have maliciously-composed routes. From the literature review it is clear that no mechanisms are available to detect and prevent vampire attacks in wireless adhoc networks. So, here I am introducing a mechanism which detects the vampire packets and prevents the forwarding of vampire packets and the formation of such type of packet inside the node.

II. Proposed System

In the proposed system, vampire attacks are clearly defined. A layered approach is used to solve the problem with the vampire attacks. Vampire packet monitoring is performed both in network layer and application layer. The network layer checking helps to point out the vampire packets from the network and the application layer checking helps to find out the vampires inside the running processes (ie, inside the node). Whenever an incoming packet is detected that is a vampire then the packet will not be forwarded and it will be discarded. Whenever a vampire is detected inside the node simply we can eliminate it.

A clean-slate secure sensor network routing protocol[3] by Parno, Luk, Gaustad, and Perrig "PLGP" can be modified to provably resist Vampire attacks during the packet forwarding phase. The original version of the protocol, although designed for security, is vulnerable to Vampire attacks. PLGP consists of a topology discovery phase, followed by a packet forwarding phase, with the former optionally repeated on a fixed schedule to ensure that topology information stays current. Here a modification in the forwarding phase of PLGP to provably avoid the above-mentioned attacks. First check the no backtracking property, satisfied for a given packet if and only if it consistently makes progress toward its destination in the logical network address space. More formally: No-backtracking is satisfied if every packet p traverses the same number of hops whether or not an adversary is present in the network. To preserve no-backtracking, need to 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. Whenever a node n forwards packet p, this by attaching a non-repayable 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[1].

Honest nodes can compose, forward, accept, or drop messages, and malicious nodes can also arbitrarily transform them. The 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 [1]. The hop count of packet p, received or forwarded by an honest node, is no greater than the number of entries in p's route attestation field, plus 1. 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 neighbour. 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. Since no-backtracking guarantees packet progress, and PLGPa preserves no-backtracking, it is the only protocol 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 Vampire attacks. This is achieved because packet progress is securely verifiable [1]. The proposed system will also give options to display the port scanning details and entropy variation.

2.1 Methodology

The proposed methodology is represented schematically in fig.2.1. The entire work is can be divided in to three phases.

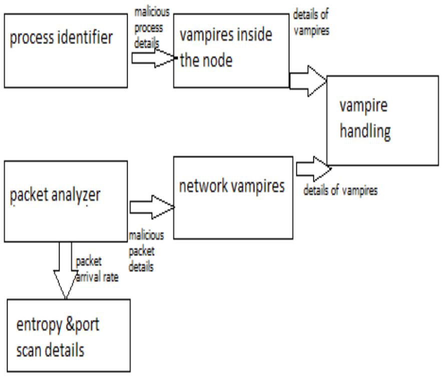

Fig 2 Architecture of the proposed system

2.1.1 Detection of anomalies from the network

In order to perform this work, an ad-hoc network is needed to create. The vampire detection system can be installed in a node as an administrative tool. The IP addresses of all the nodes in the network are needed to retrieve...All the incoming packets are monitored. The packets will go through the anomaly detection system. IP Header analysis includes each field such as IP Header Length (IHL), Type of Service (ToS), Identification (ID), Flags and etc. The IP packet header consists of 20 bytes of data and if the length is below than 20 bytes, that packet is assume as abnormal packet and go for analysis before report to administrator. An option exists within the header that allows further optional bytes to be added, but this is not normally used.

The TCP header is analyzed for the next step in this process. TCP header is built on top of IP header, which is unreliable and connectionless.TCP header occupies 20 bytes and has some limitations in header length. As mentioned, normal TCP header is 20 bytes but TCP can have another 40 bytes for option. So the header size is limited to 60 bytes. TCP Flags have six flags bits namely URG, ACK, PSH, RST, SYN and FIN, each of them has a special use in the connection establishment, connection termination or control purposes. Only few combinations of the six TCP flags can be carried in a TCP packet. URG and PSH flags can be used only when a packet carries data, for instance a combination of SYN and PSH becomes invalid. Since TCP SYN Flooding attack will flood the network with SYN packets, the three-way handshake application is checked in every packet. At this stage, packets are divided into two groups whether infected packets or normal packets. If the packet is infected, the system will distinguish the packet and go for analysis again to confirm whether the packet is truly comes from attackers. Otherwise, the normal packet will go through the network sending the data to the destination.

Like TCP header checking, header checking has to be performed for all other application layer protocols.

Packet Filtering

In order to do the packet filtering, many factors will be considered. There are three main factors in this paper:

1. The traffic filtered each packet to each protocol such as TCP, UDP and ICMP.
2. TCP flags SYN, ACK, RST, FIN, are divided to each group to check the three-way handshake is complete or not.
3. IP address is valid and not a spoofed address.

2.1.2 *Application Layer anomaly detection*

All the activities such as creation, editing and deletion of the files are monitored to find application layer vampires. Normally if an anomaly is present, then the normal rate of these processes will be altered. The file editing and deleting rate will increase drastically. And file creation rate will decrease .Consumption of memory will also increase in an abnormal fashion

2.1.3 Anomaly handling

Anomalies are handled based on the category in which the anomaly belongs to. If the vampire is from the network, then that should be prevented from entering in to the node and from forwarding to another node. We cannot delete that packet because the packet is created by some other node in the network .If any vampire is found inside the node that should be deleted immediately and should prevent from forwarding. For avoiding the entry of anomalies from the network to any packet, all the packets should satisfy no backtracking property [2]. The algorithm is explained below

Algorithm (PLGPa)

```
Function secure_forward_packet (p)
s ← extract_source_address(p);
a ← extract_attestation(p);
if (not verify_source_sig(p)) or
(empty(a) and not is_neighbor(s)) or
(not saowf_verify(a)) then
return ; /* drop(p) */
foreach node in a do
prevnode ← node;
if (not are_neighbors(node, prevnode)) or
(not making_progress(prevnode, node)) then
return ; /* drop(p) */
c ← closest_next_node(s);
p′ ← saowf_append(p);
if is_neighbor(c) then forward(p′, c);
   else
   forward (p′,next_hop_to_non_neighbor(c));
```

III. Conclusion

In this paper a detection and control method is introduced for the vampire attacks, a new class of resource consumption attacks that use routing protocols to permanently disable ad-hoc wireless networks by depleting nodes' battery power.The proposed methodology can be implemented as four modules, network layer vampire detection, Application layer vampire detection, Vampire handling and entropy details display. By using all these concepts the system is made more secure against the vampire attacks .Methods are there to detect the vampires from the network and inside the node. Once a vampire is detected then, they are handled according to their type. Also by using PLGPa algorithm the packets can be safely forward in a network. This scheme provides high level of security against the vampire attacks.

References

[1]. Tao Shu and Marwan Krunz, Fellow IEEE, Privacy –preserving and Truthful Detection of Packet Dropping Attacks in Wireless Ad hoc Networks, *IEEE Transactions on Mobile Computing,volume 14, No.4* , published on April 2015(pages 813-827)

[2]. E Y Vasserman , N Hopper ,Vampire Attacks: Draining life from wireless Ad hoc sensor networks, *IEEE Transactions on Mobile Computing,volume 12, issue 2* , published on feb 2013(pages 318-332)

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

[4]. INSENS: Intrusion-tolerant routing for wireless sensor networks, *Computer Communications 29 (2006), no. 2.*

[5]. Imad Aad, Jean-Pierre Hubaux, and Edward W. Knightly, Denial of service resilience in ad hoc networks, *MobiCom*, 2004.

[6]. Gergely Acs, Levente Buttyan, and Istvan Vajda, Provably secure on demand source routing in mobile ad hoc networks, *IEEE Transactions on Mobile Computing 05 (2006), no. 11*.

[7]. Jae-Hwan Chang and Leandros Tassiulas, Maximum lifetime routing in wireless sensor networks, *IEEE/ACM Transactions on Networking 12 (2004), no. 4.*

[8]. Anoopa S,Sudha S K ,Detection and Control of Vampire Attacks in Ad hoc Wireless Networks *,Int. Journal of Engineering Research and Applications, volume4 ,issue4(,version 6),* April 2014(pages 01-07)