INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

DETECTION AND ELIMINATION OF VAMPIRE ATTACK IN WSN USING ID-BASED SENSOR NETWORK PROTOCOL

MISS. APURVA KALE[1], PROF. G.D. GULHANE[2], DR. H. R. DESHMUKH[3]

1. PG Research Scholar and Department of CSE, IBSS College of Engineering, Amravati.
2. Asst. Professor, Department of CSE, IBSS College of Engineering, Amravati.
3. Professor and Head, Department of CSE, IBSS College of Engineering, Amravati.

Accepted Date: 15/03/2016; Published Date: 01/05/2016

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. 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 factor of O (N), where N in the number of network nodes. discuss methods to mitigate increase network-wide energy usage by a these types of attacks, including a new proof-of-concept protocol that provably bounds the damage caused by Vampires during the packet forwarding phase.

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

Corresponding Author: MISS. APURVA KALE

Access Online On:

www.ijpret.com

How to Cite This Article:

Apurva Kale, IJPRET, 2016; Volume 4 (9): 159-167

PAPER-QR CODE

INTRODUCTION

A wireless ad hoc sensor network has a wide range of application in the communication environment. It is mostly used in the remote areas, in the military communication, for finding environmental disasters etc. 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; high availability of these networks is a critical property, and should hold even under malicious conditions. The vampire attack consumes more energy from network than the original node during the transmission of message between nodes. 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 In this schemes can prevent attacks on the short-term 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 paper 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. routing infrastructure attacks as they do not disrupt immediate availability, but rather work over time to entirely disable a network.

Vampire attacks are not protocol-specific, in that they do not rely on design properties or implementation faults of particular routing protocols, but rather exploit general properties of protocol classes such as link-state, distance vector,source routing, and geographic and beacon routing. Neither do these attacks rely on flooding the network amounts of data, but rather try to transmit as little data as possible to achieve the largest energy drain, a rate limiting solution. Since Vampires use protocol-compliant messages, these attacks are very difficult to detect and prevent.

Literature Review & Related Work

Eugene Y. Vasserman and Nicholas Hopper[1] Define 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. 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 of 30 nodes. show that depending on the location of the adversary, network energy expenditure during the forwarding phase

increases from between 50 to 1,000 percent. Theoretical worst case energy usage can increase by as much as a factor of O(N) per adversary per packet, where N is the network size. The 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

G. Acs L. Buttyan, and I. Vajda [2] this paper shows attacks against ad hoc routing protocols can be subtle and difficult to discover by informal reasoning about the properties of the protocol. Demonstrated this by presenting novel attacks on Ariadne. Another message is that it is possible to adopt rigorous techniques developed for the security analysis of cryptographic algorithms and protocols, and apply them in the context of ad hoc routing protocols in order to gain more assurances about their security

J.-H. Chang and L. Tassiulas [3] In this paper authors had formulated the routing problem as maximizing the network lifetime. The new problem formulation has revealed that the minimum total energy (MTE) routing is not suitable for network-wise optimum utilization of transmission energy.

Vampire attack analysis in WSN:-

Here, present simple but previously neglected attacks on source routing protocols, such as DSR . In these systems, the source node specifies the entire route to a destination within the packet header, so intermediaries do not make independent forwarding decisions, relying rather on a route specified by the source. To forward a message, the intermediate node finds itself in the route (specified in the packet header) and transmits the message to the 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 the 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.

evaluated both the carousel and stretch attacks in a randomly generated 30-node topology and a single randomly selected malicious DSR agent, using the ns-2 network simulator. Energy usage is measured 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 saturation. independently computed resource utilization of honest and

malicious nodes and found that malicious nodes did not use a disproportionate amount of energy in carrying out the attack.

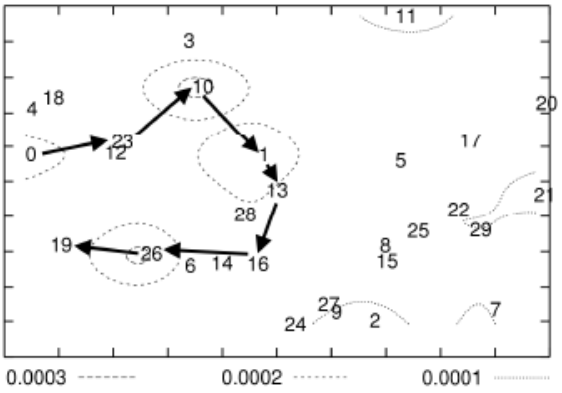
As expected, the carousel attack causes excessive energy usage for a few nodes, since only nodes along a shorter path are affected. In contrast, the stretch attack shows more uniform energy consumption for all nodes in the network, since it lengthens the route, causing more nodes to process the packet. While both attacks significantly network-wide energy usage, individual nodes are also noticeably affected, with some losing almost 10 percent of their total energy reserve per message. Fig. 1a diagrams the energy usage when node 0 sends a single packet to node 19 in an example network topology with only honest nodes. Black arrows denote the path of the packet.

In this attack, an adversary sends a packet

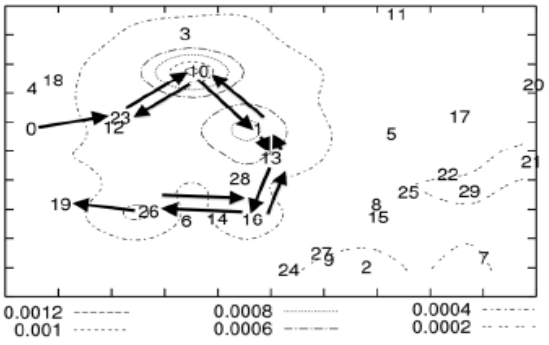
With a route composed as a series of loops, such that the same node appears in the route many times. This strategy can be used to increase the route length beyond the number of nodes in the network, only limited by the number of allowed entries in the source route.2 An example of this type of route is in Fig. 1a. In Fig. 1b, malicious node 0 carries

out a carousel attack, sending a single message to node 19 (which does not have to be malicious). Note the drastic increase in energy usage along the original path.3 assuming the adversary limits the transmission rate to avoid saturating the network, the theoretical limit of this attack is an energy usage increase factor of $O(\lambda)$, where λ is the maximum route length.

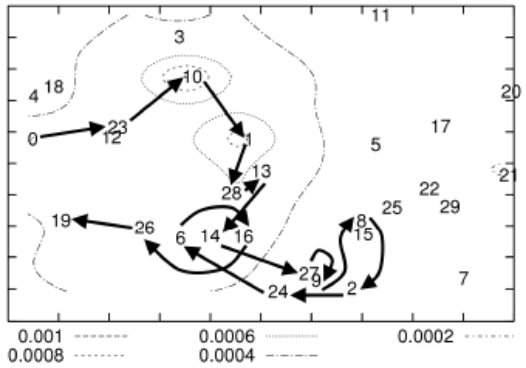
Another attack in the same vein is the stretch attack, where a malicious node constructs artificially long source routes, causing packets to traverse a larger than optimal number of nodes. An honest source would select the route Source $\rightarrow F \rightarrow E \rightarrow$ Sink, affecting four nodes including itself, but the malicious node selects a longer route, affecting all nodes in the network. These routes cause nodes that do not lie



(a) Honest scenario: node 0 sends a single message to node 19.



(b) Carousel attack (malicious node 0): the nodes traversed by the packet are the same as in (a), but the loop over all forwarding nodes roughly triples the route length (the packet traverses the loop more than once). Note the drastically increased energy consumption among the forwarding nodes.



(c) Stretch attack (malicious node 0): the route diverts from the optimal path between source and destination, roughly doubling in length. Note that while the per-node energy consumption increase is not as drastic as in (b), the region of increased energy consumption is larger. Overall energy consumption is greater than in the carousel attack, but spread more evenly over more network nodes.

Fig. 1. Energy map of the network in terms of fraction of energy consumed per node. Black arrows show the packet path through the network. Each dotted line represents an "energy equivalence zone," similar to an area of equal elevation on a topological chart. Each line is marked with the energy loss by a node as a fraction of total original charge.

Along the honest route to consume energy by forwarding packets they would not receive in honest scenarios. An example of this type of route is in Fig. 1b.The outcome becomes clearer when we examine Fig. 3c and compare to the carousel attack. While the latter uses energy at the nodes who 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.

Vampire detection method:-

Definition 1:-No-backtracking is satisfied if every packet p traverses the same number of hops whether or not an adversary is present in the network. (Maliciously induced route stretch is bounded to a factor of 1.)

This does not imply that every packet in the network must travel the same number of hops regardless of source or destination, but rather that a packet sent to node D by a malicious node at location L will traverse the same number of hops as a packet sent to D by a node at location L that is honest. If think of this in terms of protocol execution traces, no-backtracking implies that for each packet in the trace, the number of intermediate honest nodes traversed by the packet between source and destination is independent of the actions of malicious nodes. Equivalently, traces that include malicious nodes should show the same network wide energy utilization by honest nodes as traces of a network with no malicious actors. The only notable exceptions are when adversaries drop or mangle packets en route, but since we are only concerned with packets initiated by adversaries, safely ignore this situation: "premangled" packets achieve the same result—they will be dropped by an honest intermediary or destination.

```
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));
```

No-backtracking implies Vampire resistance. It is not immediately obvious why no-backtracking prevents Vampire attacks in the forwarding phase. Recall the reason for the success of the stretch attack: intermediate nodes in a source route cannot check whether the source-defined route is optimal, or even that it makes progress toward the destination. When nodes make independent routing decisions such as in link-state, distance-vector, coordinate-based, or beacon-based protocols, packets cannot contain maliciously composed routes. This already means the adversary cannot perform carousel or stretch attacks—no node may unilaterally specify a suboptimal path through the network. However, a sufficiently clever

adversary may still influence packet progress. Prevent this interference by independently checking on packet progress: if nodes keep track of route "cost" or metric and, when forwarding a packet, communicate the local cost to the next hop, that next hop can verify that the remaining route cost is lower than before, and therefore the packet is making progress toward its destination.(Otherwise we suspect malicious intervention and drop the packet.)

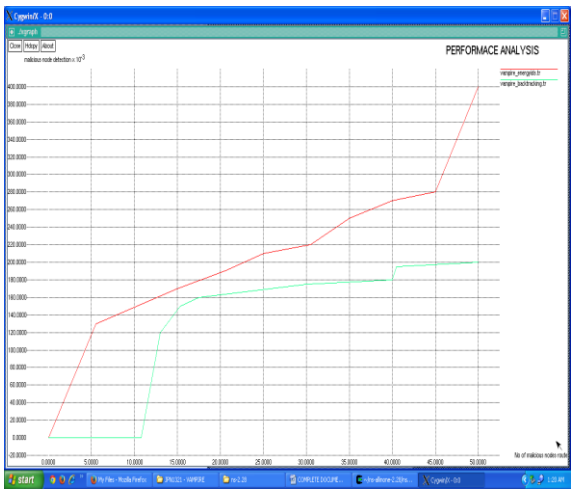
PLGP does not satisfy no-backtracking. In nonsource routing protocols, routes are dynamically composed of forwarding decisions made independently by each node.

PLGP differs from other protocols in that packets paths are further bounded by a tree, forwarding packets along the shortest route through the tree that is allowed by the physical topology. In other words, packet paths are constrained both by physical neighbor relationships and the routing tree. Since the tree implicitly mirrors the topology (two nodes have the same parent if and only if they are physical neighbors, and two nodes sharing an ancestor have a network path to each other), and since every node holds an identical copy of the address tree, every node can verify the optimal next logical hop. However, this is not sufficient for no-backtracking to hold, since nodes cannot be certain of the path previously traversed by a packet. Communicating a local view of route cost is not as easy as it seems, since adversaries can always lie about their local metric, and so PLGP is still vulnerable to directional antenna/wormhole attacks, which allow adversaries to divert packets to any part of the network.

Performance Considerations:-

PLGP imposes increased setup cost over BVR, but compares favorably to in terms of packet forwarding overhead. While path stretch increases by a factor of 1.5-2, message delivery success without resorting to localized flooding is improved: PLGP never floods, while BVR must flood 5-10 percent of packets depending on network size and topology. PLGP also demonstrates more equitable routing load distribution and path diversity than BVR.

Since the forwarding phase should last considerably longer than setup, PLGP offers performance comparable to BVR in the average case.



In total, the overhead on the entire network of PLGPa (over PLGP) when using 32-bit processors or dedicated cryptographic accelerator is the energy equivalent of 90 additional bytes per packet or a factor $O(x\lambda)$ where λ is the path length between source and destination and x is 1.2-7.5,depending on average packet size (512 and 12 bytes, respectively). Even without dedicated hardware, the cryptographic computation required for PLGPa is tractable even on 8-bit processors, although with up to a factor of 30 performance penalty, but this hardware configuration is increasingly uncommon.

CONCLUSION:-

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 of 30 nodes. Show that depending on the location of the adversary, network energy expenditure during the forwarding phase increases from between 50 to 1,000 percent. Theoretical worst case energy usage can increase by as much as a factor of O (N) per adversary per packet, where N is the network size. The 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

REFERENCES:-

1. Eugene Y. Vasserman and Nicholas Hopper "Vampire Attacks: Draining Life from Wireless Ad Hoc Sensor Networks"- IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL. 12, NO. 2, FEBRUARY 2013.

2. G. Acs, L. Buttyan, and I. Vajda, "Provably Secure On-Demand Source Routing in Mobile Ad Hoc Networks,"- IEEE Trans. Mobile Computing, vol. 5, no. 11, pp. 1533-1546, Nov. 2006.

3. J.-H. Chang and L. Tassiulas, "Maximum Lifetime Routing in Wireless Sensor Networks," IEEE/ACM Trans. Networking, vol. 12, no. 4, pp. 609-619, Aug. 2004. T. Aura, "Dos-Resistant Authentication with Client Puzzles,"Proc. Int'l Workshop Security Protocols,2001.

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

5. D. Bernstein and P. Schwabe, "New AES Software Speed Records,"Proc. Ninth Int'l Conf. Cryptology in India: Progress in Cryptology (INDOCRYPT),2008.

6. D.J. Bernstein, "Syn Cookies," http://cr.yp.to/syncookies.html, 1996.

7. I.F. Blaked, G. Seroussi, and N.P. Smart, Elliptic Curves in Cryptography,vol. 265. Cambridge Univ., 1999.

8. J.W. Bos, D.A. Osvik, and D. Stefan, "Fast Implementations of AES on Various Platforms," Cryptology ePrint Archive, Report 2009/ 501, http://eprint.iacr.org, 2009.

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

10. Cryptographic Hardware and Embedded Systems(CHES), 2004.[21] R. Fonseca, S. Ratnasamy