VAMPIRE ATTACKS: EXPLORATION & CONSEQUENCES

Virjot Kaur, Priyanka Rani, Dr. Satwinder Singh

Abstract— This paper provides a survey of the Vampire attacks which is the resource depletion attack. It permanently disables networks by quickly draining nodes' battery power. All the protocols are vulnerable to Vampire attacks as they are demoralizing, challenging to detect and are laid-back to carry out using as few as one malicious insider sending only one protocol-compliant message. Here the analysis of Vampire attacks and consequences are discussed, and solution provided till date are explored.

Index Terms — Vampire attacks, Stateless protocol, Stateful protocol, Clean state sensor network routing.

1 Introduction

Vampire attacks are a variant of DOS attacks which performs resource consumption on neighbor nodes. In Vampire attack, targeted packets are modified for preparing long routes or misguiding the packets. Resource depletion attack or Vampire attack is such attack where a compromised node involves in generating more network traffic which depletes the energy of the nodes. The Vampire node behaves as per the underlying protocol making the system difficult to detect such attack [4].

Vampire attacks can be defined as the configuration and broadcast of a message that causes additional energy to be used up by the network than if an honest node transmitted a message of identical size to the same destination [1].

Vampire attacks are not protocol-specific, as they do not depend on design properties or implementation faults of specific routing protocols, but rather exploit general properties of protocol classes such as link-state, distance-vector, source routing, geographic and beacon routing. Further, these attacks do not depend on submerging the network with an enormous quantity of data but somewhat try to transfer as tiny data as possible to achieve the largest energy drain.

Vampire attacks are equilateral to those used to protect routing infrastructure, and so existing secure routing protocols such as Ariadne [9], SAODV [11], and SEAD [8] do not guard against Vampire attacks. Current work on secure routing attempts is to ensure that intruder is unable to discover the route and return an invalid network path, but Vampires do not interrupt or modify identified paths, instead using existing valid network paths and protocol-compliant messages [1].

Vampire attacks have many attractive features that make them the most prominent attack:

- Vampire Attacks do not target particular protocol i.e.; they are not protocol-specific.
- They don't distrupt instant availability.
- Vampires use Protocol-Compliant messages.
- Transmission of trivial information with largest energy drain is achieved.
- They do not interrupt or modify discovered paths.

2 Vulnerable Protocols

Vampire attacks are mostly recognized in link state, distance vector, source routing, Geographic, beacon routing protocols and logical ID based sensor network. [3]

2.1 Link-State Protocols

Link-state protocols include two main classes such as Open Shortest Path First (OSPF) and Intermediate to Intermediate system (IS-IS). The basic theory of link-state routing is that every node constructs a map of the connectivity of the system, in the form of a graph, showing which nodes are connected to other nodes [4].

2.2 Distance-vector Protocols

Distance-vector protocols are based on computing the direction which means the next hop and the distance mean the cost to measure the cost of next node [3]. The router informs its neighbor about the changes in the topology when the transaction is in progress. It uses Bellman-ford algorithm, and Ford-Fulkerson algorithm to calculate paths.

2.3 Coordinate and Beacon based Protocols

Coordinate and beacon based protocols are the protocols which move according to the coordinates such as GPRS and BVR [13] [14]. In GPRS, it contours the barrier of packets until to the path of the target is grasped whereas in BVR the packets are routed towards the beacon closest to the target and then move towards the target [1].

2.4 Clean state sensor network routing

Clean-State Sensor Network routing protocol which is also called PLGP works in two phases namely topology discovery phase and packet forwarding phase. PLGP first version is sus-

- *Virjot Kaur is currently pursuing Masters degree program in Computer Science & Technology in Central University of Punjab, India, PH-9530696325. E-mail: kaurjot3511@gmail.com*
- *Priyanka Rani is currently pursuing Masters degree program in Computer Science & Technology in Central University of Punjab, India, PH-7307819573. E-mail: singla.priyanka44@gmail.com*
- *Dr. Satwinder Singh is currently working as Assistant Proffesor in the department of Centre for Computer Science & Technology, Central University of Punjab, India, E-mail: satwindercse@gmail.com*

ceptible to Vampire attacks, but its modified version with attestations (PLGP-a) is secure from Vampire attacks. Packet forwarding phase is safe from the Vampire attack as the no-backtracking property provides provable security against Vampire attacks [15] but no satisfactory solution is provided for topology discovery phase.

3 TYPES OF VAMPIRE ATTACKS

Vampire attacks are primarily categorized into two classes by the protocol.

3.1 Attacks on Stateless Protocols

Stateless Protocols are those in which the node specifies the entire route in the packet header to the destination. Intermediate nodes cannot make independent forwarding decisions instead of the direction indicated by the source. To forward a message, the intermediate node finds itself in the path and transmits the message to the next hop. The load is at the node of origin to ensure that the route is legal at the time of transmitting data and that every node in the path is a physical neighbor of the previous route hop. [1]

Further, there are two types of attacks in stateless protocol as described below:

3.1.1 Carousel Attack

Carousel Attack is where the source is at the initial stage and sink is at the last stage [16]. In this kind of attack, an intruder composes packets with intentionally introduced routing loops. It is called carousel attack as it sends packets in circles as shown in figure 1.

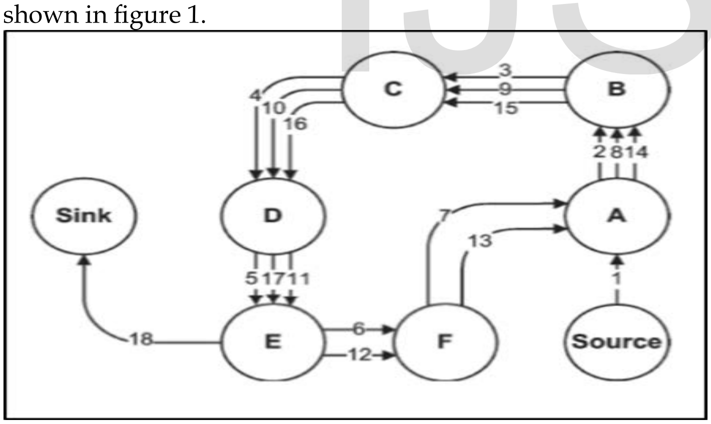

Fig. 1. Carousel Attack[1]

It aims source routing protocols by misusing the limited verification of message headers at forwarding node, allowing a single packet to traverse frequently the same set of nodes.

As shown in figure 1, source (malicious node) sends the packet to the loop twice before reaching the sink node. The honest node would immediately exit the loop from node E to sink. Carousel attack is used to increase the route length beyond the number of nodes in the network.

3.1.2 Stretch Attack

Stretch Attack also targets source routing, in this type of attack intruder builds artificially long routes, potentially crossing every node in the network. It is called a stretch attack, since it increases packet path lengths, causing packets to be processed by some of the nodes that are independent of hop count along the shortest path between the adversary and packet destination.

Figure 2 shows the stretch attack. In this honest route is dotted while the malicious route is dashed. The last link to sink is shared. Here source node act as a malicious node and intentionally send the packet through the long route. Instead of using the route (Source →F→ E→ Sink), affecting only four nodes including itself, it uses the long route. Due to this long route (dashed), nodes that are not along the honest route are forced to utilize energy by forwarding packets they would not receive in honest route scenario.

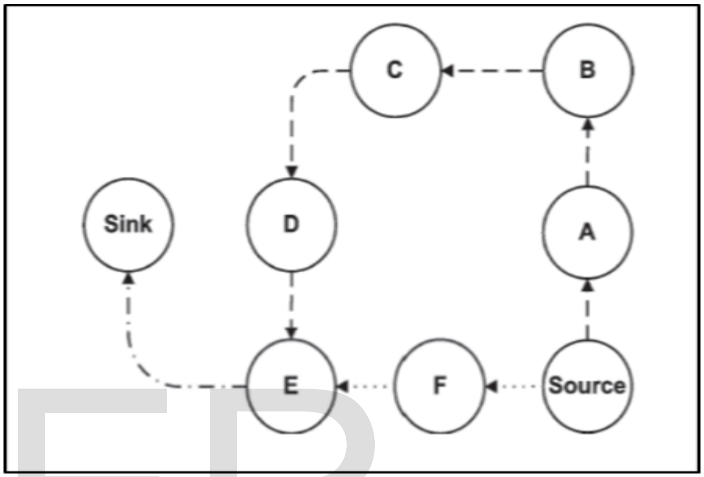

Fig. 2. Stretch Attack [1]

The stretch attack can increase the energy usage by up to an order of magnitude, depending on the position of the malicious node. Further, the impact of such attacks can be increased by combining them, by increasing the number of malicious nodes in the network or by simply sending more packets.

3.2 Attacks on Staeful Protocols

In the stateful protocols, nodes are aware of topology, state, and forwarding decisions. Nodes are independent and make local forwarding decisions on that stored state. Types of attacks on stateful protocol are:

3.2.1 Directional Antenna Attack

Vampires have slight control over packet progress when forwarding decisions are made self-reliant by each node, but they can still discard energy by resuming a packet in various parts of the network. Using a directional antenna intruder can send a packet to arbitrary parts of the system while also sending the packet locally. This utilizes the energy of nodes that would not have had to process the original packet. It can be accomplished more than once by placing the packet at various distant points in the network, at the added cost to the intruder for each use of the directional antenna [1].

3.2.2 Malicious Discovery Attack

Another attack on all earlier mentioned routing protocols (including stateful and stateless) is forged route detection. In maximum protocols, every node will forward route discovery packets, meaning it is possible to initiate a flood by sending a single message. A malicious node has numeral ways to induce a perceived topology change:

- It may falsely claim that a link is down, or claim a new link to a nonexistent node.
- Further, two cooperating malicious nodes may declare the link between them is down. However, nearby nodes might be able to monitor communication to detect a link failure (using some neighborhood update scheme) [1].
- More severe attacks become possible when nodes declare that a long distance route has changed.

This attack is trivial on open networks with unauthenticated routes since a single node can emulate multiple nodes in neighbor relationships [12], or falsely claim nodes as neighbors.

4 MITIGATION TECHNIQUES

Many ramifications have been provided for detection of Vampire Attacks. The Carousel Attack can be prohibited completely by having forwarding nodes check source route for loops, but this adds additional forwarding logic and thus more overhead. Another alternate solution is to alter how intermediate nodes process the route. To forward a message, a node must regulate the next hop by locating itself in the source route. A node can search for itself from the destination backward instead from the source forward so that any loop that includes the current node will be automatically pruned.

Stretch Attacks are more challenging to prevent than carousel attacks. Loose source routing technique can be used in which intermediary nodes may substitute a portion of the path in the packet header if they know a better route to the destination, but this makes necessary for the nodes to discover and find optimal routes for at least some proportion of other nodes, somewhat defeating the as-needed discovery advantage.

Directional antenna attacks are further tough to prevent. Packet Leashes cannot inhibit this attack since they are not meant to defend against malicious message sources, only intermediaries [10].

Malicious discovery attacks are still vulnerable to Vampire attacks, and no particular solution is provided to date.

Review of countermeasures and algorithms provided till date discussed are as follows:

- In [2] new protocol is proposed that provides provable security against Vampire attacks. This protocol requires no specialized hardware and provides message delivery even in an environment with active adversaries. Three approaches were discussed to design secure routing protocol. This protocol has to be analyzed under attack and investigation has to be done to improve its efficiency.
- [5] Proposed a system that overcomes the challenges of Vampire attacks by using the Energy load observing Algorithm (ELOA) and the energy utilization is reduced profoundly. ELOA functions in two phases namely Network configuring phase and Communication phase. The performance of existing protocol is quantified using a small number of adversaries but not applicable when malicious nodes are increased.
- [4] Implemented a prototype application that simulates the Wireless Ad Hoc Sensor Network with the resource depletion attack model. Empirical results reveal that the prototype can provide encouraging results. The results show that such attacks can be prevented, and will help in real time implementation of a protocol to prevent attacks.
- [6] Evaluated the Vampire attacks by assessing the vulnerabilities of existing protocols and modifying existing protocol to deplete Vampire attacks. Secure routing protocol PLGPa with ECC is proposed to prevent Vampire attacks by confirming that packets make progress towards their destination but has the limitation that no way to ensure if the packet has reached the destination or dropped.
- [7] Proposed a method to detect and secure data packets from Vampires during the packet forwarding phase. PLGP with attestations is used for identifying the malicious attack. M-DSDV routing protocol is used to detect and eliminate the resource depletion attack from the network attacks. Defense against forwarding phase is discussed, but no solution is provided in topology discovery phase.
- From all the above-discussed countermeasures, the effective and best solution is clean state sensor network routing protocol proposed by [2].

5 CLEAN STATE SENSOR NETWORK ROUTING

A clean-slate secure sensor network routing protocol proposed by [2], which provides the provable security against Vampire attack during packet forwarding phase. It works on a clean slate approach i.e. it prevents the entry of any malicious node into the network. It operates in two phases as follows:

5.1 Topology Discovery Phase

It is repetitive on a fixed schedule to ensure that topology information stays current. When discovery initiates, each node has a partial view of the network — the node knows only itself. Nodes find their neighbors using local broadcast, and form ever-expanding "neighborhoods," stopping when an entire network is a single group.

During this process, nodes form a tree of neighbor relationships and group membership that will later be used for addressing and routing. At the completion of discovery, each node should calculate the identical address tree as other nodes [3]. All nodes learn about each other's virtual addresses and cryptographic keys. The final address tree is provable after network convergence, and all forwarding decisions can be individually verified. Moreover, assuming each authentic network node has a unique certificate of membership (assigned before network deployment), nodes who effort to join multiple groups, produce duplicates of themselves in various locations, or otherwise cheat during discovery can be identified and removed.

Discovery commences with a time-limited period during which every node must broadcast its existence by broadcasting a certificate of identity, including a public key, signed by a trusted offline authority.

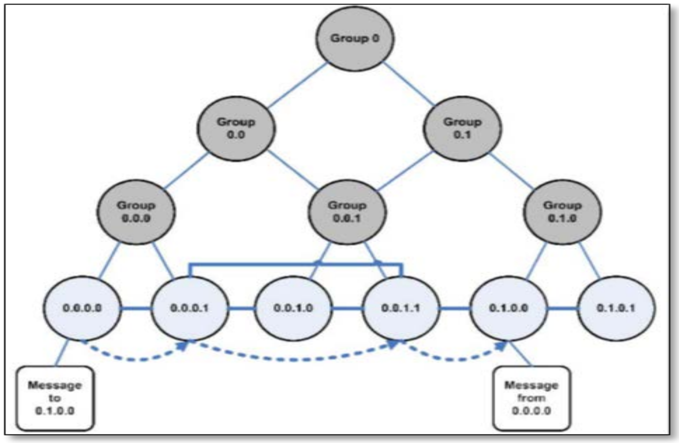

Fig. 3. Working of Protocol [2]

Each node starts as its group of size one, with a virtual address 0. Nodes who overhear broadcasted messages create groups with their neighbors. When two individual nodes (each with an initial address 0) merge to build a group of size two, one will take the address 0, and the other becomes 1. Groups integrate favorably with the smallest neighboring group, which may be a single node. Like individual nodes, each group will primarily pick a group address 0, and will select 0 or 1 when combining with another group. Each group member appends the group address to their address, for example, node 0 in group 0 becomes 0.0, and node 0 in group 1 becomes 1.0, and so on [2].

Each time two groups incorporate, the address of each node is extended by one bit. Implicitly, this forms a binary tree of all addresses in the network, with node addresses as left. Nodes will appeal to join with the smallest group in their vicinity. When larger groups merge, they both broadcast their group IDs (and the IDs of all group members) to each other and continue with a merge protocol identical to the two-node case. Groups that have developed large enough that some members are not within radio range of other groups will join through “gateway nodes,” which are within range of both groups. Each node reserves the identity of one or more nodes through which it received an announcement that another group exists. Topology discovery continues in this manner until all network nodes are members of a single group. By the completion of topology discovery, each node learns every other node’s virtual address, public key, and certificate, since every group member know the identities of all other group members and the network converges to a single group [3].

5.2 Packet Forwarding Phase

Throughout the forwarding phase, all choices are made autonomously by each node. When receiving a packet, a node decides the next hop by finding the most significant bit of its address that varies from the message originator’s address. Thus, every progressing event (excluding when a packet moves within a group to reach a gateway node to proceed to the next group) reduces the logical distance to the target since node addresses should be firmly closer to the destination [3].

To repel against the Vampire attacks this phase plays an important role. The property of no-backtracking must be conserved to resist the Vampire attacks. This property of No-backtracking is satisfied if every packet p traverses the same number of hops whether or not an attacker is present in the network. Thus it emphases on not allowing the packet to diverge from its destination too much. To reserve no-backtracking, a verifiable path history is added to every PLGP packet, similar to route authentications. This packet history is used together with PLGP’s tree routing structure so that every node can firmly verify progress, avoiding any significant adversarial influence on the path taken by any packet which traverses, at least, one honest node [2].

This protocol is secure from Vampire attacks in forwarding phase, but it is vulnerable in topology discovery phase. The protocol assumes that there is a network authority present which assigns every node with a unique identity. This protocol is further modified with attestations but does not provide security during the topology discovery phase. Furthermore, the detection techniques cannot prevent this as they will work post merge or after the end of the recursive grouping algorithm but the damage will be done till that, and there are chances that the Vampire will not be found by the detection techniques.

6 Conclusion

The paper gave an overview of the Vampire attacks and the protocols vulnerable to this attack. Further, the mitigation techniques were discussed, and the clean state protocol proves to be secure from the Vampire attacks. Many other solutions are considered but do not provide satisfactory resistant against Vampire attacks. The clean state protocol provides provable security but has one limitation in topology discovery phase, and this makes it vulnerable to Vampire attacks. So the further modification is required in the topology phase to secure the protocol entirely from Vampire attacks.

References

[1] Eugene Y. Vasserman, Nicholas Hopper, “Vampire Attacks: Draining life from Wireless Ad-hoc Sensor Networks,” IEEE transactions on mobile computing,vol.12 no.2, 2014.

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

[3] Kirthika.K, B. Loganathan, “Vampire Attacks in Wireless Sensor Networks- A Survey,” *International Journal of Advanced Research in Computer Engineering & Technology*(IJARCET), vol. 3, pp 2460-2465, July 2014.

[4] Haseena, M. Venkatesh Nayak, “Detection and Prevention of Resource Depletion Attacks in Wireless Ad Hoc Sensor Networks,” *International Journal of Advanced Research in Computer and Communication Engineering* (IJARCCE), vol. 3, pp 8350-8352, October 2014.

[5] P. Geeta Prasanthi, P.Seshu Babu, "ELOA beside Vampire Attacks in Wireless Sensor Networks," *International Journal of Research* (IJR), vol. 1,pp 1263-1271, December 2014.

[6] K.Vanitha, V.Dhivya, " A Valuable Secure Protocol to Prevent Vampire Attacks in Wireless Ad Hoc Sensor Networks," *International Journal of Innovative Research in Science, Engineering, and Technology* (IJIRSET), vol.3, pp 2441-2445, March 2014.

[7] S.R.Singh, N.Babu CR, "Improving the Performance of Energy Attack Detection mechanism," *International Journal of Scientific and Research Publications*, vol.4, pp 1-5, July 2014.

[8] Y.-C. Hu, D.B. Johnson, and A. Perrig, "SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks," Proc. IEEE Workshop Mobile Computing Systems and Applications, 2002.

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

[10] Y.-C. Hu, D.B. Johnson, and A. Perrig, "Packet Leash: A Defense against Wormhole Attacks in Wireless Ad Hoc Networks," Proc. IEEE INFOCOM, 2003.

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

[12] J.R. Douceur, "The Sybil Attack," Proc. Int'l Workshop Peer-to-Peer Systems, 2002.

[13] B. Karp and H.T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," Proc. ACM MobiCom, 2000.

[14] R. Fonseca, S. Ratnasamy, J. Zhao, C.T. Ee, D. Culler, S. Shenker, and I. Stoica, "Beacon Vector Routing: Scalable Point-to-Point Routing in Wireless Sensor nets," Proc. Second Conf. Symp. Networked Systems Design & Implementation (NSDI), 2005

[15] M.R. Arjunkar, A.S. Sambare, S.R. Jain, "A Survey On: Detection & Prevention of Energy Draining Attacks (Vampire Attacks)," *International Journal of Computer Science and Application*, vol.8, no. 1, pp 13-16, Jan-Mar 2015.

[16] S.P. a. V.G. Kasabegoudar, "Secure Transmission against Vampire Attack using Wireless Adhoc Sensor Network," *International Journal of Computer Applications*, vol.125, no.3, pp. 39-43, September 2015.