Vampire Attack: Draining Life from Wireless Ad –hoc Sensor Networks

Priya S. Gosavi
PG student
M.B.E.Society's college of Engineering, Ambajogai, Maharashtra, India

B. M. Patil
Professor
M.B.E.Society's college of Engineering, Ambajogai, Maharashtra, India

ABSTRACT
Ad-hoc wireless networks are dynamic in nature. Ad-hoc networks are not depends on any predefined infrastructure. Whenever there is need of communication at that point these network can be deployed. In this paper we discuss Vampire attacks. All protocols susceptible for vampire attack. Vampire attacks are very easy to carry out throughout the network and difficult to detect. In this paper we discuss a method to mitigate vampire attacks. Then we compare new method with existing protocol and Beacon Vector routing. And we come to the conclusion that new protocol is better as it detect and prevent from vampire attack.

Keywords
PLGP, EDSA, BVR,PLGP

1. INTRODUCTION
In near future, ad-hoc wireless sensor network perform a very important role due to its instantly deployable property, and its application which are useful in number of areas like military, industrial purpose, environmental purpose, etc. These ad-hoc networks can be deploy anywhere at any time and provides continuous connectivity. These networks are vurnable to denial of service attack [1].

These attacks instantly destroy networks. But vampire attacks are different from Denial of Service attack as vampires do not disturb instant availability, it slowly drains resource i.e. nodes battery life and destroy whole network. Vampire attacks means malicious nodes creates and send unnecessary data packets, so that node's battery is consumed. Vampire attacks use protocol compliant messages, uses small data chunks for sending data packets, so more battery consumed for small data.

1.1 Attacks on stateless protocol
In these protocols, nodes are not aware of states of network. Source node defines the route on which packet must be travelled, so sender must ensure that the path which is defined must be exist. An intermediate node does not make any decision about packet forwarding. When sender sends packet at defined route, at that time, path is stored in packet header for some period which can be useful for another time. That's why intermediate nodes needs very little forwarding logic [2].

- *Carousel attack:* An adversary sends a packet with a route composed as a series of loops, such that the same node appears in route many times.
- *Stretch attack:* A malicious node construct artificially long source routes, causing packets to traverse a larger than optimal number of nodes.

1.2 Attacks on stateful protocols
In Stateful protocols each node knows the topology of network and aware of state of every other node in the network. Intermediate nodes make independent decision based on stored state. Following are some attacks:

- *Directional antenna attack:* Vampires have less control over packet when packet is forwarded independently, but malicious node may forward packet at any part of network that is called directional antenna attack.
- *Malicious discovery attack:* Sender node send discovery packet, malicious nodes also send discovery packet in network. The nodes who listens discovery message they send reply to the sender nodes but as some discovery messages are malicious, so reply to those messages may not reach at its destination due to malicious nodes not found. This leads traffic in network with loss of energy.

2. RELATED WORK
First Power draining attacks are not defined and not mitigates at routing layer, then [2] research and found 'sleep deprivation attacks' at routing layer. The proposed attack prevent nodes from entering sleep mode. Then [3] found that denial of sleep attack at mac layer. Then again more work on 'resource exhaustion attacks' at MAC layer and transport layer. Quality of service and Reduction of Quality attacks and defense against these attacks are discussed in [4, 5, 6, 7, 8, 9, and 10]. This produces degradation in network performance. Deng at al. discuss path based DoS attacks and defenses [11] and uses one way hash chain to limit the number packet sent by given node. Another path based attacks are one is wormhole attack introduced in [12], which creates virtual or private connection between two malicious nodes. Another is directional antenna attacks. These attacks disturb route discovery. [11] Defense against these two attacks using packet leashes [10] this comes at high cost and not always applicable.

3. MITIGATION METHOD
Carousel attack sends packet in loop. We can avoid this by using some extra forwarding logic and it will leads more overhead. In DSR, loop can be detected, but it will not check path in forwarded packets. In source routing protocol, path is signed by source. When loop is detected, it should be corrected and then sent on. Instead of correcting and sending again, dropping that packet is more convenient and beneficial.

Stretch attack is more difficult to prevent. If intermediate nodes not takes independent decisions and uses strict source routing that is packet must travel only the path which is defined by source. And if that path is not present or damaged

then problem is created. In loose source routing if header has defined path, but if intermediate node knows another better route then intermediate nodes may change the route. Now, In this case malicious nodes may send packet through longest route or on route which is not exist. Here is a advantage that intermediate nodes uses cached route, which are already discovered and stored.

4. CLEAN SLATE ROUTING

This protocol is used to resist vampire attack during forwarding phase. Also known as PLGP, as invented by Parno, Luk, Gaustad and Perrig [13]. This is original version which is discovered for security purpose but this protocol also susceptible for vampire attack.

There are two phases in this protocol first is topology discovery phase another is packet forwarding phase.

- *Topology discovery phase:* Discovery phase organizes nodes into tree for addressing purpose. Nodes announce their existence in network by broadcasting their certificate of id and its public key. By grouping process tree is formed. Each node starts grouping with its group size 1 and virtual address 0. Then neighboring nodes overhears and form a group with that node and address becomes 0 and 1 for each node. This process will continue until all nodes forms in a single tree. There are another groups are there which are away from another group and so they are out of radio range and they can't communicate each other. These two long distance groups communicate through gateway nodes.
- *Packet forwarding phase:* Once nodes are arranged in tree like structure then it is easy for packet to traverse a path using address defined. Sender sends hello message to near nodes they overhear and send reply. Sender sends data packets to neighboring nodes after receiving reply. Each node makes independent decision by using most significant bit of address field. See Algorithm 1.

4.1 Algorithm 1: Function Packet forward

s ← extract_source_address (*p*);

c ← closest_next_node (*s*);

if is_neighbore (*c*) **then** forward (*p*, *c*);

Else

r ← next_hop_to_nonneighbor (*c*);

forward(p,r);

Sender node send packet to near node, each node takes decision. So intermediate nodes does not know that on which route packet is travelled, So malicious nodes can divert that packet at anywhere away from destination or in loop. So Clean Slate Routing is vulnerable to vampire attack.

4.2 Provable security against vampire attack

To avoid vampire attacks there is need to keep track on the path travelled by packet so we can avoid to forward it at any part of network. First we define a no-backtracking property, when packet travelled same number of hops with and without presence of malicious nodes and packet makes continuous progress towards its destination at that time that packet satisfy no-backtracking property.

In Clean Slate routing protocol, paths are bounded by tree. In other protocols tree is not used for addressing purpose. So Clean Slate routing protocol is different from other protocols. Every node have same copy of tree for addressing. Every node can verify the optimal next logical hop. This is not enough for no-backtracking property because adversaries can always lie.

4.3 No-backtracking implies vampire resistance

No-backtracking property resists vampire attack in packet forwarding phase. The reason of success of stretch attack is the intermediate nodes do not check whether packet choses optimal path or does it makes continuous progress towards destination? Adversaries send packet at any part of network. Clever adversaries can affect any type of routing protocol, so we can check packet progress at each node, if packet makes continuous progress towards destination.

5. CLEAN SLATE PROTOCOL WITH ATTESTATION (PLGPa)

Clean Slate Routing with attestation (PLGPa) satisfies no-backtracking. To maintain no-backtracking property we added verifiable path history to each packet. Path history is similar to route authentications and path vector signature. This attestation is attached with PLGP packet, Packet is securely forward with history attached with it. If node n forward packet p, packet is attached with a nonreplyable signature. This signature form a chain attached to every packet. Every forwarding node can verify this signature chain to ensure that is packet makes progress or travels away from destination.

All messages are signed by originators. Adversaries can change some changeable part of packet field. Attestation field can be changed, altered, removed entirely. To prevent this we use one way signature chain, where attestation is append to the packet. There are number of signature algorithms are there, in this simulation SHA1 digital signature algorithm and elliptic curve cryptography ECDSA algorithm are used. See algorithm 2.

5.1 Algorithm 2: Secure Forward Function

s ← extract_source_address(*p*)

a ← extract_attestation*(p);*

If (*not* verify_source_sig (*p*)) or

(empty (a) and not is_neighbore (s)) or
(not saowf_verify(a))
then

return();

foreach node in a **do**

prevnode ← node;

if (*not* are neighbors(node, prevnode)) or
(*not* making progress (prevnode, node))
then

return();

c ← closest_next_node (s);

p´ ← saowf_append (*p*);

if is_neighbore (*c*) then forward(*p´*, *c*);

else forward(*p´*,next_hop_to_nonneighbor(*c*));

6. PERFORMANCE EVALUATION

Clean Slate routing protocol has increased setup cost than Beacon Vector Routing. Clean Slate routing protocol never floods but Beacon Vector Routing protocol floods sometimes depending on network size. Clean Slate routing protocol has more load distribution and path diversity than Beacon Vector Routing protocol.

Clean Slate Routing with attestation protocol includes path attestation, so packet size is increased, bandwidth use increased, and radio power use also increases. By adding packet verification at each intermediate node requires more processor utilization, more time, and more power. So to minimize bandwidth utilization we use chain signature which compact less than 30 bytes. Chain signatures requires bilinear maps and more costly computation, so to avoid that uses Tate pairing which is universally accepted. And used elliptic curve ECDSA for cryptography.

Finally we compare Beacon Vector Routing (BVR), Clean Slate Routing Protocol (PLGP), and Clean Slate Routing Protocol with attestation (PLGPa) to know that which one is better in all aspect.

Clean Slate Routing Protocol (PLGP) is better than in Beacon Vector Routing (BVR) in case of flooding, overhead etc. Clean Slate Routing Protocol with attestation (PLGPa) increases some processing utilization, required time, additional power, extra packet verification requirement but as it satisfies no-backtracking property by checking packets at each node it makes progress or not and this way it resist vampire attack. We compare these three protocols to know which one is better in performance with some matrices like delay, throughput, control overhead, packet dropping rate, energy consumption, jitter, routing overhead etc. Beacon vector routing is based on geographic metrics that is beacons.

In BVR, packets floods after some specific periods. Packet dropping rate is increases as network size increases in above three protocols. PDR is defined as packets per unit time that are not reached at destination called packet dropping rate. Packet dropping rate is more in BVR than PLGP and PLGPa. In PLGPa, PDR is more than PLGP due to affected packets will be dropped immediately. Another comparison of these three protocols is Network size v/s Total energy consumption as shown in figure 1. Energy consumption is more in beacon vector routing than remaining both protocols due to it depends on beacons which are geographic coordinates assigned to nodes for communications. Energy cons is less in PLGPa which is our main motivation of this simulation. As PLGP a resist vampire attacks it will not consume unnecessary energy like PLGP.

Figure2 is Comparison graph of three protocol Network size v/s throughput. Throughput is the rate of total number of packets successfully delivered at destination per unit time. In all these three protocols throughput is decreases as network size increases. As nodes increases, congestion may increase so throughput of BVR is better as compared with remaining both. Because in BVR beacons are floods and packets delivered at destination. But it also decreases as network size increased. Then PLGPa is better than PLGP, as it is prevented from vampire attacks. Control overhead is increased as network size increase. And BVR control overhead is less than both. In PLGP and PLGPa co-Oh is slightly differentiated. It is more in PLGPa.

Our main, motivation is regarding with energy consumption. So our protocol is better than remaining both.

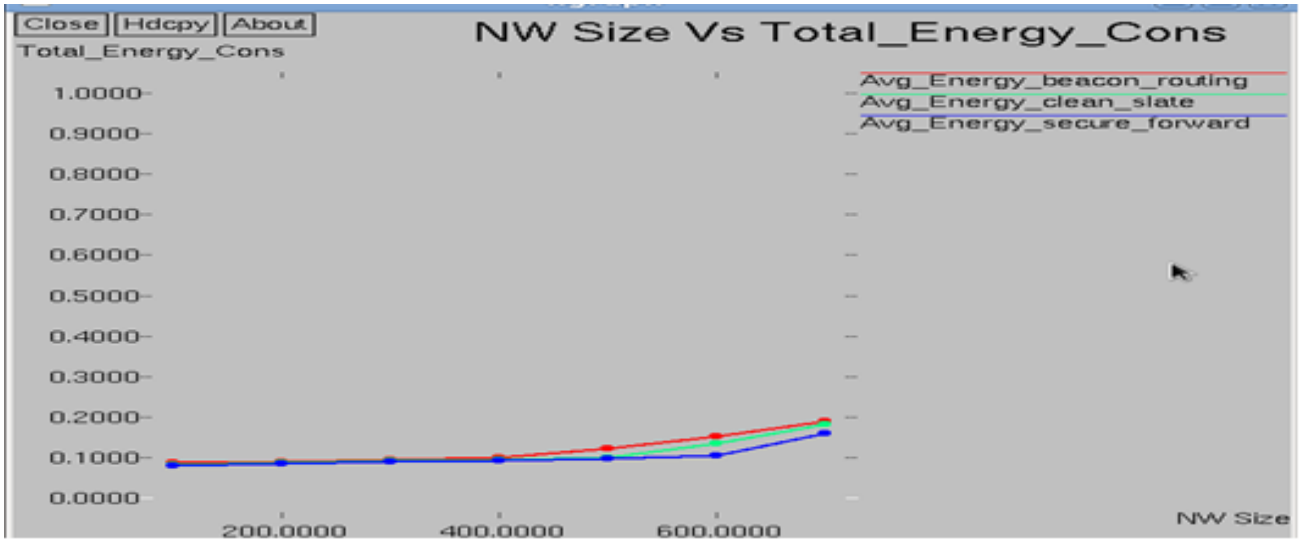

Figure 1 Comparison Graph of Network Size v/s Total energy consumption

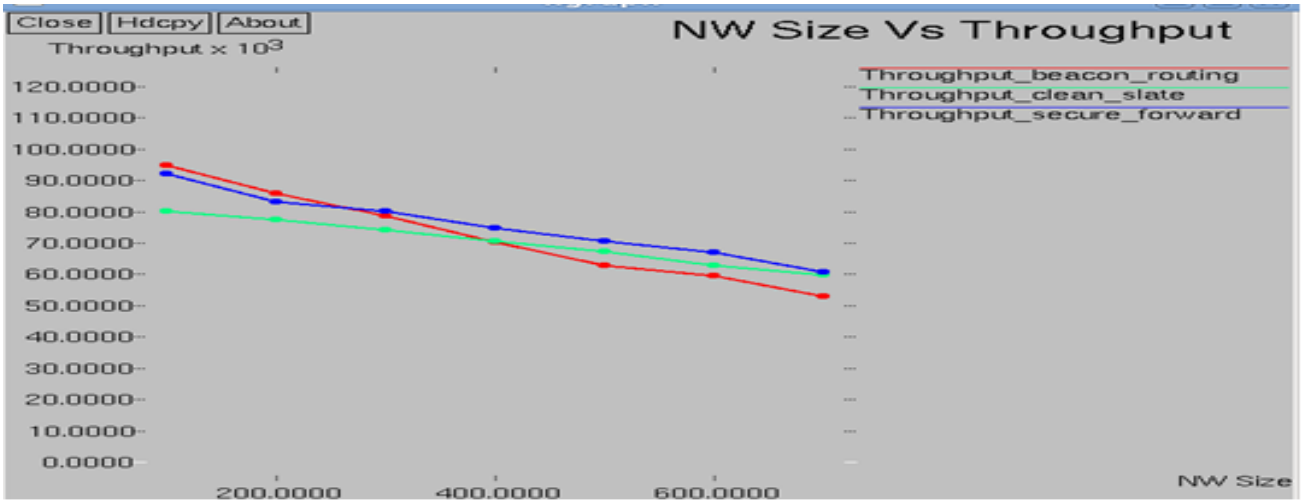

Figure 2 Comparison Graph of Network Size V/S Throughput

7. FUTURE WORK

In this simulation, we secure packet forwarding phase of clean slate routing protocol and prevent packets from vampire attacks. But we have not secured discovery phase. In future by using synchronous discovery and ignoring discovery messages during intervening period. We can secure and damage limitations, and defense discovery phase.

8. CONCLUSION

We study shortly some vampire attacks, how they occurs in Clean Slate routing protocol (PLGP), how it can be avoided, detected in Clean Slate routing protocol with attestation (PlGPa), lastly we compare three protocols in some performance metrics. In the view of energy consumption better protocol is PLGPa as it prevent packet from circulating in loop. As in another metrics PLGP looks better due to some increased processing. PLGPa has little drawbacks but they can be avoided.

9. REFERENCES

[1] Anthony D. Wood and John A. Stankovic, Denial of service in sensor networks, Computer35 (2002), no. 10.

[2] Frank Stajano and Ross Anderson, The resurrecting duckling: security issues for ad-hoc wireless networks, International workshop on security pdecisions", Journal of Systems and Software, 2005, in press.

[3] David R. Raymond, Randy C. Marchany, Michael I. Brownfield, and Scott F.Midkiff, Effects of denial-of-sleep attacks on wireless .sensor network MAC protocols, IEEE transaction on vehicular technology.

[4] Sharon Goldberg, David Xiao, Eran Tromer, Boaz Barak, and Jennifer Rexford, Path-quality monitoring in the presence of adversaries, SIGMETRICS,2008.

[5] Mina Guirguis, Azer Bestavros, Ibrahim Matta, and Yuting Zhang, Reduction of quality attacks on internet and end system, INFOCOM 2005.

[6] Aleksandar Kuzmanovic and Edward W. Knightly, Low-rate TCPtargeted denial of service attacks: the shrew vs. the mice and elephants, SIGCOMM, 2003.

[7] Yu-Kwong Kwok, Rohit Tripathi, Yu Chen, and Kai Hwang, HAWK: Halting anomalies with weighted choking to rescue well-behaved TCP sessions from shrew DoS attacks, Networking and mobile computing, 2005.

[8] Xiapu Luo and Rocky K. C. Chang, On a new class of pulsing denial of-service attacks and the defense, NDSS, 2005.

[9] Haibin Sun, John C. S. Lui, and David K. Y. Yau, Defending against low-rate TCP attacks: dynamic detection and protection, ICNP, 2004.

[10] Guang Yang, M. Gerla, and M.Y. Sanadidi, Defense against low-rate TCP-targeted denial-of-service attacks, ISCC, 2004

[11] Jing Deng, Richard Han, and Shivakant Mishra, Defending against path based DoS attacks in wireless sensor networks, ACM workshop on security of ad hoc and sensor networks, 2005.

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

[13] Rushing attacks and defense in wireless adhoc network routing protocols, WiSE, 2003.

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

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