Impact Factor (SJIF): 3.632

International Journal of Advance Research in Engineering, Science & Technology

e-ISSN: 2393-9877, p-ISSN: 2394-2444

(Special Issue for ITECE 2016)

Consistent Approach for Defending MANet from Jellyfish Attacks

Prof. Shreya Sanghvi

CSE Department, SLTIET.

Abstract — *The collection of wireless mobile nodes which forms the temporary network without using any kind of network infrastructure is called Mobile Ad hoc network. The channel is wireless, topology is dynamic so, there is no clear line of defense. Due to these reasons, the MANETs are prone to numerous security attacks. This paper focuses on the most important kind of routing attack –jellyfish attack. We use AODV routing protocol to understand the attack and its prevention. In jellyfish attack, the malicious node will delay the forwarding of the RREP packet to the next adjacent node in the MANET. Here we present the solution to this type of attack by taking the nodes of the network into promiscuous mode. It means other nodes in the topology can hear the packets sent from and received by the neighbor nodes.*

I. INTRODUCTION

The mobile ad hoc network is a collection of wireless mobile nodes dynamically forming temporary network. These types of network communicate without using any existing network infrastructure or centralized administration. The transmission range of wireless network interface is limited, so multiple hops may be needed for one node to communicate with other node in the network. In ad hoc networks, each mobile node act as the host as well as router for forwarding packets for other mobile nodes which are not in direct transmission range of each other. Each node participates in an ad hoc routing protocol that allows it to discover multi hop paths through the network to any other node. Many routing protocols have been developed such as AODV, DSR, DSDV, OLSR etc.

Security in Mobile Ad hoc Networks (MANETs) is the most important concern for the basic functionality of the network. The availability of the services of MANETs can be guaranteed only by assuring that the security issues have been resolved up to at least some extent. Certain characteristics of MANETs such as dynamic topologies, resource constraints, limited physical security, and no infrastructure which makes its security very vulnerable. It has no central monitoring and management and also no clear line of defense. This features imposes the security threats and results in to various attacks which may compromise the availability of network, confidentiality and integrity of the data been sent over the network.

Lack of centralized authority causes it to operate on the basis of mutual trust. This feature makes it most vulnerable to be exploited by an attacker inside the network. There are many types of security attacks, which risk the MANET. [1][2].

II. ROUTING PROTOCOLS

Ad hoc networks have limitations such as limited bandwidth, power constrain, highly dynamic topology, high error rates etc. Compared to wired networks, in an ad hoc network all nodes are mobile and are connected dynamically with each other in arbitrary manner. Nodes of mobile ad hoc network also work as router to discover, establish and maintain the route of each other. So the routing protocols for wired network cannot be directly used for wireless network. Various routing protocols have different characteristics and depending on the nature of application the appropriate protocol is selected.

The routing protocol can be classified as follows:

- Table driven (Proactive):
  - DSDV (Dynamic Destination Sequenced Distance Vector)
  - OLSR (optimized link state routing) protocol, etc.
- Demand driven (Reactive):
  - AODV (Ad Hoc on Demand Distance Vector),
  - DSR (dynamic source routing),
  - TORA (temporally ordered routing algorithm) protocol, etc.
- Hybrid routing protocol:
  - ZRP (zone routing protocol)

Reactive routing protocols can be dramatically reduce routing overhead because they do not need to search for and maintain the routes on which there is no data traffic. This property is very appealing in the resource limited environment.

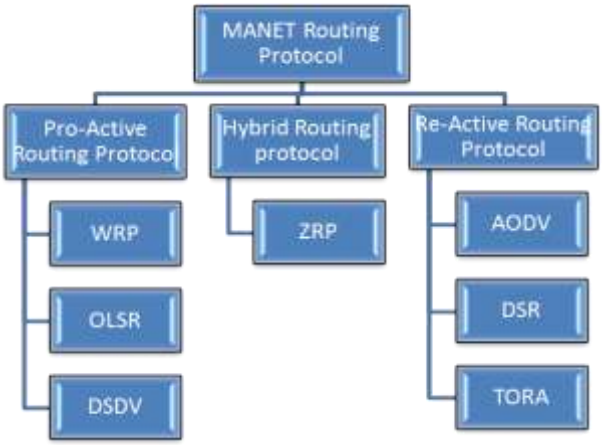

FIG 1: classification of routing protocols

III. OPERATION OF AODV (Ad Hoc On Demand Distance Vector) Protocol

It is a reactive routing protocol, meaning that it establishes a route to a destination only on demand. AODV is, as the name indicates, a distance-vector routing protocol. AODV avoids the counting-to-infinity problem of other distance-vector protocols by using sequence numbers on route updates, a technique pioneered by DSDV. AODV is capable of both unicast and multicast routing [3].

There are mainly three types of routing protocols are available for MANETs: Proactive routing protocols, Reactive routing protocols and Hybrid routing protocols. Proactive protocols are those which continuously checks for the topology of the network nodes. So when there is need of route to any source to destination then it is always available. This protocol increases the overhead as the cost of maintaining the network might be very high if network topology is changing frequently. Reactive protocol establishes the route only when it is required. It does not check the network topology continuously. So it is considered as resource preserving protocol. Hybrid protocols are combination of both. In this paper wee analyze the security threats on AODV protocol.

AODV is a well-known and one of the standard reactive routing protocol for MANETs which provides dynamic, self-starting and multi hop network. It is the reactive routing protocol which does not need to maintain the routing table in advance for every node [4]. AODV operates in two phases 1) Route Discovery and 2) Route maintenance. Whenever a source node needs to communicate with another node for which it has no routing information, Route Discovery process is initiated by broadcasting a Route Request (RREQ) packet to its neighbors. Each node has its own sequence number and this number increases when links change. Each node judges whether the channel information is new according to sequence numbers. Figure 2 illustrates the route discovery process in AODV. In this figure, node S is trying to establish a connection to destination D. First, the source node S refers to the route map at the start of communication. In case where there is no route to destination node D, it sends a Route Request (RREQ) message using broadcast-ing. RREQ ID increases one every time node S sends a RREQ. Node A and B which have received RREQ generate and renew the route to its previous hop. They also judge if this is a repeated RREQ. If such RREQ is re-ceived, it will be discarded. If A and B has a valid route to the destination D, they send a RREP message to node S. By contrast, in case where the node has no valid route, they send a RREQ using broadcasting. The exchange of route information will be repeated until a RREQ reaches at node D. When node D receives the RREQ, it sends a RREP to node S. When node S receives the RREP, then a route is established. In case a node receives multiple RREPs, it will select a RREP whose the destination sequence number (Dst Seq) is the largest amongst all previously received RREPs. But if Dst Seq were same, it will select the RREP whose hop count is the smallest [5].

In Figure 3, when node B detects disconnection of route, it generates Route Error (RERR) messages and puts the invalidated address of node D into list, then sends it to the node A. When node A receives the RERR, it refers to its route map and the current list of RERR messages

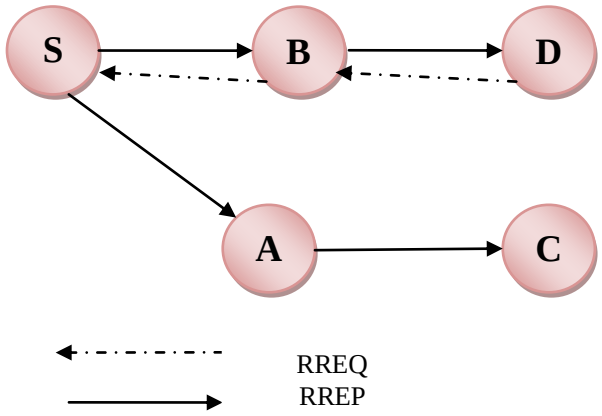

Fig2: route establishing

If there was a route to destination for node D included in its map, and the next hop in the routing table is a neighboring node B, it invalidates the route and sends a RERR message to node S. In this way, the RERR message can be finally sent to the source node S.[6]

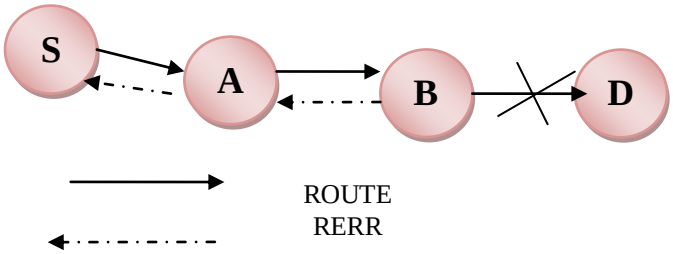

Fig3: route establishing

IV. LITERATURE SURVEY

In jellyfish attack, the attacker injects the unwanted delays in the network. In this type of attack, the attacker node first gets the access of the network and become a part of it. Then it introduces the delays in the networks by delaying all the packets that it receives, once the delays are propagated then packets are released in the network. This causes the high end to end delay, also the overall performance of the network decreases considerably .

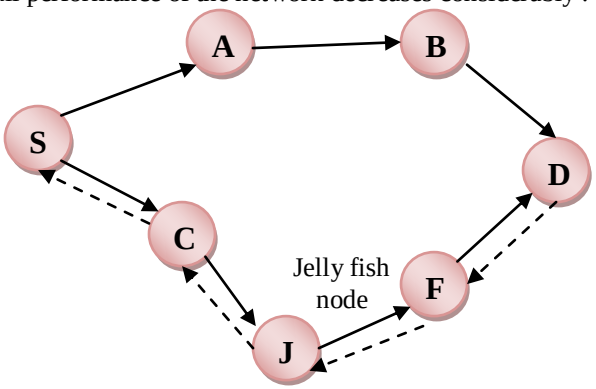

Fig4: jellyfish attack

It is not always the case that malicious node does not obey the rules of routing protocol. But it is difficult to distinguish the jelly fish attack from the network congestion and packet losses that occur naturally in the network. So jelly fish is hard and resource consuming to detect. This attack can be carried out by variety of mechanisms. [5]

One of the mechanisms of the jellyfish attack consists in a node delivering all received packets, but in scrambled order instead of the canonical FIFO (first in first out) order. This attack cannot be successfully opposed by the actual TCP packet reordering techniques, because such techniques are effective on sporadic and non-systematic reordering. The second mechanism is the same as that used in the shrew attack, and involves performing a selective black hole attack by dropping all packets for a very short duration. The flow enters timeout at the first packet loss caused by the jellyfish attack, then periodically re-enters the timeout. The third mechanism consists in holding a received packet for a random time before processing it, increasing delay variance which increases the average end to end delay of the network. This causes TCP traffic to be sent in bursts, therefore increasing the odds of collisions and losses, and it causes an incorrect estimation of the available bandwidth in congestion control protocols based on packet delays.

As shown in figure 4, the J node the jelly fish node. Here we concentrate on jelly fish delay attack. We consider that J node is imposing the unnecessary delay in the network. The jelly fish node will keep all the packets coming to it for some time and then it will forward it to its neighboring nodes [7]

V. PROPOSED METHOD TO PREVENT JELLY FISH ATTACK

We propose a method which uses the promiscuous mode to prevent black hole and jelly fish attack. This mode allows a node to intercept and read each network packet that arrives in its entirety. It means that if node A within the range of node B, it can overhear communication to and from B even if those communication do not directly involve A.

Jelly fish node imposes the unnecessary delay in forwarding the packets to its next nodes. The proposed algorithm detects if the node is making delay in forwarding the packets to other nodes. Once it is detected that particular node adds unnecessary delays to the network by not forwarding the packets received by it immediately, then we can announce that particular node as jelly fish node.

The method for detecting the jelly fish node is as follows.

Consider node S wants to communicate with node D, and the G is the malicious node. Node S will broadcast the RREQ in the network and will wait for RREP packet to obtain the fresh route to destination D. Here the nodes are in promiscuous mode. So each node can hear the packets sent to and from the other nodes in that range. Suppose node S has sent the RREQ packet to node G, then node N who is neighbor of node G is operating in promiscuous mode. So node N will come to know that packet has been sent to node G. Also whenever node G will forward the packet to the next hop, then also node N will come to know. We can use this important property of promiscuous mode to detect the delay. The neighbor node N will record the time of arrival of RREQ packet. It will also record the time at which that node forwards the RREQ to other node. If the time span between arrival of packet and forwarding of packet is in the permissible interval, then that node is good node. But if time span between arrival and dispatch or forwarding of the packets is higher than a permissible time, then that node is declared as jelly fish node.

Notations:
RREQ: Route Request Packet
RREP: Route Reply Packet
S: Source
D: Destination
flag: Flag value true for jelly fish node
packet_sending_time: Time at which packets are sent by the node
packet_receiving_time: Time at which packets are sent by the node
time_span: difference between packet_sending_time and packet_receiving_time
permissible_time: Allowable time delay in forwarding packets

Step 1: (Initialization Process)

Start the route discovery phase with the source node S.
Source node will broadcast the RREQ to its neighbors.

Step 2: (Set flag value)

Set original_destination_path = false;
flag=false;

Step 3: (Calculate the time_span)

// As the nodes are in promiscuous mode, they are able to listen the packets to and from the neighboring node.

Calculate the time_span value at its neighbor node.

time_span = packet_sending_time – packet_receiving_time;

Step 4: (Identify and remove jelly fish node using promiscuous mode)

```
If( time_span > permissible_time)
{
        Flag=true; // this node is jelly fish node
}
If(flag ==true) // if node is jelly fish
{
        Generate Alarm packets containing node_id of jelly fish node
        Broadcast the Alarm packets.
}
```

Step 5: (Storing jelly fish node id in detain_list)

Each node will store node_id in Alarm packet in the detain_list

detain_list will be check by each node before forwarding the RREQ.

VI. TESTING AND RESULTS

A. End to end delay

Table 1 End to end delay for jelly fish attack

Pause time	AODV	JFAODV	PJFAODV
100	1.3	1.54	1.4
200	1.5	1.59	1.51
300	0.7	1.56	1.23
400	0.9	1.8	1.2
500	1.1	1.98	1.3

Figure 5 shows the end to end delay versus pause time graph for jelly fish attack. The average end to end delay for AODV is 1.1 sec, for jelly fish delay AODV is 1.694 sec and for prevented jelly fish attack is 1.3 sec. the proposed solutions reduces the delay by 0.4 seconds.

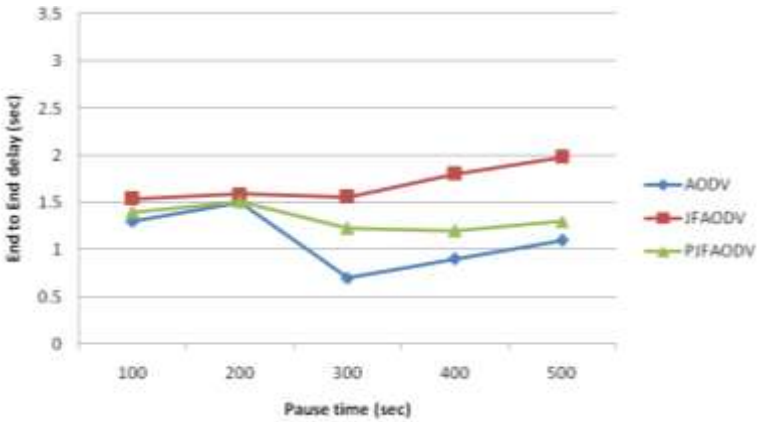

Fig 5: Graph for end to end delay

B. Throughput The table 2 shows the throughput for AODV, AODV under jelly fish delay and prevented AODV against jelly fish attack. The throughput of AODV is 144.51 Kbps, for AODV under jelly fish attack is 73.31 Kbps and for prevented AODV against jelly fish attack is 98.5 Kbps.

Table 2 throughput for jelly fish attack

	AODV	JFAODV	PJFAODV
100	99.01	50.03	70.01
200	99.08	57.89	69.34
300	137.27	78.23	89.78
400	167	81.09	112.76
500	220.23	99.34	150.72

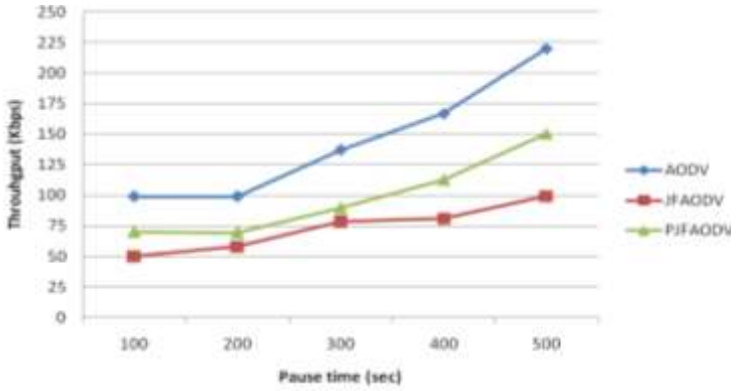

Fig 6: Graph for throughput versus pause time for jellyfish attack

The figure 6 shows the graph of throughput versus pause time. The AODV is having highest throughput, the jelly fish AODV decreases the throughput quiet considerably. And the prevented AODV against jelly fish delay improves the throughput by 25 Kbps.

VII. CONLCUSION

The proposed algorithm PJFAODV reduces the time delay by on an average 0.4 sec and improves the performance. The proposed algorithm improves the throughput by 25Kbps.

REFERENCES

1) Subir Kumar Sarkar, T. G. Basavaraju, C. Puttamadappa, "Ad Hoc Mobile Wireless Networks Principles, Protocols And Applications" Chapter 3, Page59-100.
2) Moitreyee Dasgupta, S.Chaudhary, N. Chaki, "Routing Misbehavior In AdHoc Network", International Journal Of Computer Applications.
3) Luis Girones Quesada, "A Routing Protocol for Manets".
4) Tony Larsson, Nicklas Hedman, "Routing Protocols In Wireless Ad HocNetworks- A Simulation Study".
5) Xiaoyan Hong, Kaixin Xu, and Mario Gerla. "Scalable routing protocols for mobile ad hoc networks", Journal of Network, IEEE, Jul/Aug 2002, page11-21.
6) Shreya Sanghvi, Tejas Patalia, Naren Tada, "Mitigating the Attacks in the Mobile Ad Hoc Network: Proposals And Challenges", International Journal of Computer and Electronics Engineering.
7) Nidhi Purohit, Richa Sharma, Hiteishi Diwanji, "Simulation Study Of Black Hole Anf Jellyfish Attack On Manet Using Ns3", International Journal Of Computer Application.