A NOVEL CLUSTER BASED WORMHOLE AVOIDANCE ALGORITHM FOR MOBILE AD-HOC NETWORKS

Subhashis Banerjee[1] and Koushik Majumder[2]

[1]Department of Computer Science & Engineering, West Bengal University of Technology, Kolkata, India
koushik@ieee.org
[2]Department of Computer Science & Engineering, West Bengal University of Technology, Kolkata, India
mail.sb88@gmail.com

ABSTRACT

A severe type of network layer security attack called Wormhole attack can occur in MANET, during which a malicious node captures packets from one location in the network, and tunnels them to another colluding malicious node at a distant point, which replays them locally. This paper presents a hierarchical cluster based Wormhole attack avoidance technique to avoid such scenario. The concept of hierarchical clustering with a novel hierarchical 32-bit node addressing scheme is used for avoiding the attacking path during the route discovery phase of the DSR protocol, which is considered as the under lying routing protocol. Pinpointing the location of the Wormhole nodes in the case of exposed attack is also given by using this method.

KEYWORDS

MANET, Wormhole attack, DSR, Hierarchical clustering, Hierarchical node addressing.

1. INTRODUCTION

Many routing protocols have been proposed for mobile ad-hoc networks (MANETs). Most of the routing protocols, however, do not consider the security and attack issues because they assume that other nodes are trustable. This lack of security mechanism provides many opportunities for the attackers to conduct attacks on the network and Wormhole attack is one of them. It is a network layer attack. In this attack, one malicious node captures and tunnels the packets to another malicious node located at a distant point, which replays them locally.

In this paper we proposed a hierarchical cluster based Wormhole attack avoidance mechanism. At first the hierarchical clusters are formed up to 3-level, and during the cluster formation a unique 32-bit hierarchical address is assigned to each nodes within the cluster boundaries. With the help of the hierarchical ad-dressing scheme the receiver can compute the intermediate nodes address in a valid path on receiving of a packet. So when it receives a route request packet from the sender it can check for all valid addresses in the packet. If some mismatch occurs it reports this path as an attacking path and avoids the path in case of further communication.

David C. Wyld (Eds) : ICCSEA, SPPR, CSIA, WimoA - 2013
pp. 233–241, 2013. © CS & IT-CSCP 2013 DOI : 10.5121/csit.2013.3524

The remaining paper is organized as follows: in section 2 we give the literature review. In section 3 we give our proposed scheme with our assumptions and cluster formation technique. Different types of Wormhole attacks and there countermeasures have been given in section 4 and 5. The complete algorithm in pseudo code is presented in section 6, and we finally conclude the paper in section 7.

2. RELATED WORKS

Y. Hu et al. in [1] introduced two Wormhole attack detection and prevention schemes. One is called the Temporal Leashes which is a time based solution. An-other is Geographical Leashes which is location based solution. Though both of the Leashes are reliable and have a high detection rate, Temporal Leashes suffers from need of tightly synchronized clocks and the Geographical Leashes suffers from some hardware need like GPS information.

S. Jen et al. proposed simple Hop-Count Analysis based scheme [2] for avoiding Wormhole attacks in MANET called MHA. MHA uses the observation that the route under the Wormhole attack has a smaller hop-count than normal. As a result, users who avoid routes with relatively small hop-counts can avoid most Wormhole attacks. Delay per Hop Indication (DelPHI) [3] is another hop count analysis based solution that uses delay as a parameter for detecting Wormhole attack in MANET.

Wormhole Attack Prevention Algorithm (WAP) [4] is a neighbour monitoring based solution. In WAP all nodes monitor their neighbours' behaviour when they send RREQ messages to the destination, to detect neighbours that are not within the maximum transmission range but pretend to be neighbours. When a source node sends RREQ it starts a Wormhole prevention timer (WPT). If it receives some RREP messages after the timer got expire it detects a route under Wormhole attack among the routes. Once Wormhole node is detected, source node records them in the Wormhole node list. All the neighbour monitoring based solutions are less energy efficient. It assumes that a node can always monitor ongoing transmissions even if the node it-self is not the intended receiver.

D. B. Roy et al. proposed the first cluster based Wormhole attack detection method [5]. They divided the entire network in clusters. Each cluster has a cluster head and there is a guard node in the intersection of two overlapping clusters. A cluster head in the inner layer detects a malicious activity and informs the cluster head of the outer layer, and then the outer layer cluster head has the responsibility to inform the other nodes in the network about the malicious nodes. D. B. Roy et al. did not provide a practical method for cluster formation, the cluster head selection and the guard node selection. Also the method cannot pin point the location of the Wormhole and it cannot detect multiple Wormhole attack.

A detailed literature survey on Wormhole attack and their existing countermeasures with a comparison can be found in our previous work [6].

3. PROPOSED SCHEME

Now we will present our cluster based Wormhole attack avoidance mechanism. Where the receiver can identify whether there is a Wormhole in the routing path and avoid it during the route discovery phase of the DSR protocol. The proposed cluster based hierarchical mobile ad-hoc network model is shown in fig.1.

3.1. Cluster Formation:

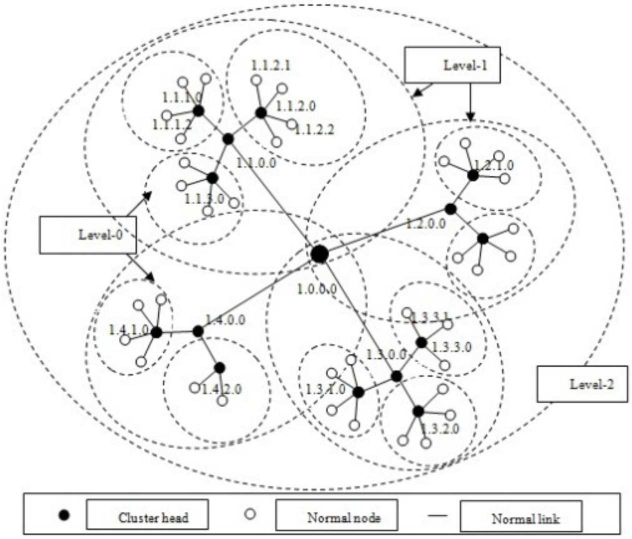

Figure 1. Hierarchical cluster formation and addressing

3.1.1. Hierarchy Definition

Here we consider a hierarchical (up to level-3) cluster model as described in [7]. All mobile nodes are first grouped into few disjoint level-0 clusters, and among them one node is selected as the cluster head (we will describe the cluster head selection criteria in “3 Cluster head selection”). All nodes in the cluster are in the direct communication range from the cluster head. All the level-0 clusters are grouped into few overlapping level-1 clusters and in every level-1 cluster a node is selected as the cluster head of that cluster. Then the level-1 and level2 clusters are formed recursively using the same procedure as level-0.

3.1.2. Hierarchical Node Addressing

Here we introduce a hierarchical addressing scheme for the nodes in the network. In the next section we will use the addressing scheme for detecting and preventing the Wormhole attack. All cluster heads at level-2 will get the address in this format X.0.0.0. The level-1 cluster heads will get the address like X.Y.0.0. The level-0 cluster head address is in the format X.Y.Z.0. And finally the nodes in the level-0 cluster will get the address in the format X.Y.Z.W where X, Y, Z, and W are any integer value in the range 0 to 255, e.g. 25.45.68.50.

3.1.3. Cluster head selection criteria

• *Remaining Power:* In order to ensure event dissipation of power by all the nodes and for increasing the overall network life time we need to select the cluster heads from among the nodes periodically on the basis of the maximum remaining power of a node power.
• *Reliability:* A node is a reliable one if other nodes in the network previously route the packet through it. In our approach each node should maintains a Neighbour Reliability table that stores the node id and the reliability value.
• *Node Mobility:* Node with the low mobility is selected as the cluster head. If the cluster head change its link to other nodes very frequently then we have to select a new cluster head.

3.1.4. Cluster Creation

Once the cluster head has been selected according to the previously defined criteria it creates the HELLO packets, and set its TTL value to 1. Then flood the packet to discover all 1-hop neighbours. Then it creates a level-0 cluster. To organize the level-0 clusters level-1 clusters are created. After level-1 clusters have been created they create the level-2 clusters using the same technique described above.

4. PROPOSED SCHEME

Our proposed hierarchical cluster based mobile ad-hoc network model is susceptible of the following four types of attacks:

1) Intra level intra cluster Wormhole attack
2) Inter level-0 Intra level-1 cluster Wormhole attack
3) Inter level-0 inter level-1 cluster Wormhole attack
4) Inter level inter cluster Wormhole attack.

4.1. Intra Level Intra Cluster Wormhole Attack:

During this attack the attacker first place two malicious nodes in the same cluster, and then establish a Wormhole link between them. Consider the e.g. illustrated in fig.2 a Wormhole link is created between the nodes 1.1.2.1 and 1.1.2.2 by using two malicious nodes X and X'. During the root discovery the sender 1.1.2.2 floods the RREQ packet within the cluster. The cluster head 1.1.2.0 and the malicious node X will receive the packet, and then X encapsulates it to a packet destined to X'. X' then send it to the destination 1.1.2.2. Due to the encapsulation the hop count value of the packet does not increase and as a result the destination will find the source to its closed neighbour. And then it may select the compromised path (going through the malicious nodes) with low hop count. Afterwards the malicious nodes can drop the packets or spying on the content of the packets going through the compromised path.

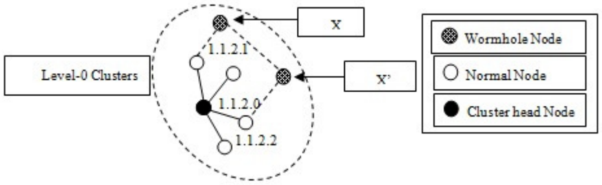

Figure 2. Intra level intra cluster Wormhole attack

4.2. Inter Level-0 Intra Level-1 Cluster Wormhole Attack:

During this attack two nodes which are in two different level-0 clusters are used for creating a Wormhole link between the sender and receiver which belongs to a same level-1 cluster. Consider the e.g. illustrated in fig. 3, where a wormhole link is created between the nodes 1.1.2.1 and 1.1.1.2 (which are in the same level-1 cluster but belongs to two different level-0 clusters) by using two malicious nodes Y and Y'. And then during the route discovery the Inter Level-0 Intra Lev-el-1 Cluster Wormhole attack is carried out by these malicious nodes described as above scenario.

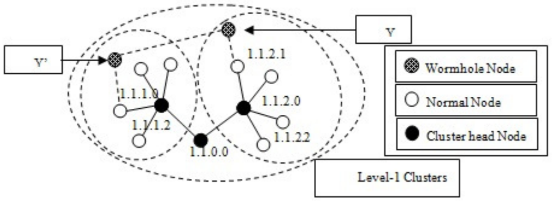

Figure 3. Inter level-0 intra level-1 cluster Wormhole attack

4.3. Inter Level-0 Inter Level-1 Cluster Wormhole Attack:

During this attack two nodes which belong to two different level-0 clusters are used for creating a Wormhole link between the sender and receiver which belongs to two different level-1 clusters. Consider the e.g. illustrated in fig. 4 a Wormhole link is created between the nodes 1.1.2.2 and 1.3.3.1 (which are in two different level-0 and level-1 clusters) by using two malicious nodes W and W'. During the route discovery phase the malicious nodes use the wormhole link between them to carry out the Inter Level-0 Inter Level-1 Wormhole attack same as previous.

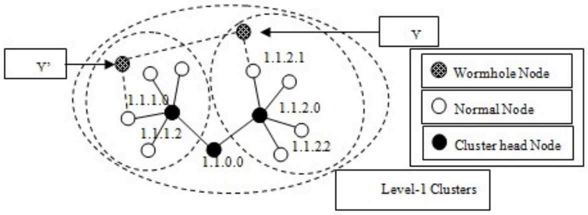

Figure 4. Inter level-0 inter level-1 Wormhole attack

4.4. Inter Level Inter Cluster Wormhole Attack:

To carry out this attack the attacker first place two malicious nodes in two different level clusters, and then establish a Wormhole link between them. Consider the e.g. illustrated in fig. 5 a wormhole link is created between the nodes 1.1.2.2 and 1.2.0.0 (which are in two different level clusters) by using two malicious nodes Z and Z'. And then during the route discovery phase the Inter Level Inter Cluster Wormhole attack is carried out by these malicious nodes described as above scenario.

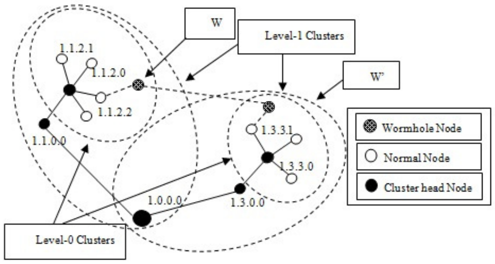

Figure 5. Inter level inter cluster Wormhole Attack

5. Procedure For Wormhole Attack Avoidance

Now in the next section we will show how our proposed algorithm avoids the above mentioned four types of Wormhole attacks with example.

5.1. Intra Level Intra Cluster Wormhole Attack Avoidance:

Consider the fig. 2 when the receiver receives the RREQ packet it extracts the source and destination addresses from it, in our example which are 1.1.2.1 and 1.1.2.2 respectively. After this destination will invoke a procedure which computes the intermediate cluster heads addresses like this: first it checks the level-2 id of the corresponding addresses, if same then checks level-1 id and after this level-0 and node id will be checked. In our example the destination observe that only the node id differs in two addresses, so it identify that the both sender and receiver nodes are within the same cluster, so there exist only one path between the sender and receiver via the cluster head of that cluster whose address is 1.1.2.0. Now the destination checks the RREQ packets that it received for the valid path (as we are using the DSR protocol the RREQ packet contains all the intermediate node ids), here which is 1.1.2.1→1.1.2.0→1.1.2.2.

5.2. Level-0 Intra Level-1 Wormhole Attack Avoidance:

Consider the fig. 3 when the receiver 1.1.1.2 receives the RREQ packet it extracts the source and destination addresses from it, in our example which are 1.1.2.1 and 1.1.1.2 respectively. Now the destination can see that the level-0 id differs in two addresses, so it identify that the both sender and receiver nodes belong to two different level-0 cluster, and the sender should send the packet through the cluster head at level-1, whose address is 1.1.0.0, then the receiver node search for the legitimate intermediate nodes address in the RREQ packet, and reject the packets that don't contain all the legitimate intermediate nodes address. Otherwise it accept the packet and sends a RREP through the reverse path contained in the DSR packet i.e., 1.1.1.2 –> 1.1.1.0 –> 1.1.0.0 –> 1.1.2.0 –> 1.1.2.1.

5.3. Inter Level-0 Inter Level-1 Wormhole Attack Avoidance:

Consider the fig. 4 after RREQ packet has been received; the destination extracts the source and destination addresses from the packet which are 1.3.3.1 and 1.1.2.2 respectively. Now the destination can see that the level-2 id differs in two addresses, so it identify that the both sender and receiver nodes belong to two different level-1 cluster, and the sender should send the packet through the cluster head at level-2, whose address is 1.0.0.0. Now the receiver searches the RREQ packet for the intermediate nodes address, and reject the packets that don't contain the cluster head id 1.0.0.0. Otherwise in case of a valid RREQ it sends a RREP through the reverse path contained in the DSR packet i.e., 1.1.2.2 –> 1.1.2.0 –> 1.1.0.0 –> 1.0.0.0 –> 1.3.0.0 –> 1.3.3.0 –> 1.3.3.1.

5.4. Inter Level Inter Cluster Wormhole Attack Avoidance:

Consider the fig. 5 after the destination receives a RREQ packet it extracts the source and destination addresses from the packet which are 1.1.2.2 and 1.2.0.0 respectively. Now the destination starts to match the addresses from MSB and find that the level-1 id differs in two addresses, so it identify that the both sender and receiver nodes belong to two different level cluster one is in level-0 and one is in level-1, and the sender should send the packet through the cluster head at level-2, whose address is 1.0.0.0. Now the receiver checks the intermediate nodes address in the RREQ packet, and reject the packets that don't contain the cluster head id 1.0.0.0.

After that it sends a RREP through the reverse path contained in the legitimate RREQ packet i.e., 1.1.2.2–> 1.1.2.0 –> 1.1.0.0 –> 1.0.0.0 –> 1.2.0.0.

6. Proposed Algorithm

Algorithm: RREQ packet forwarding and Wormhole attack avoidance

```
Step 1. The sender node initiates a route discovery by flooding the RREQ packets within the
    cluster.

Step 2. The cluster head of this cluster that the sender belongs to, receives the packet.

Step 3. The Cluster head extracts the source and destination addresses from the packet, and
    identify the mode of communication – a) Intra cluster b) Inter cluster c) Intra level or d)
    Inter level and also sets the Next_Hop address like follows:
    3.1. The cluster head starts matching the receiver address with its own address from the MSB
         (during the matching the cluster head considers only the non zero bits of the addresses).
    3.2. If (mismatch occurs) then
        3.2.1. Set the Next_Hop address value = Current cluster head address.
        3.2.2. Replace the first right most non zero bit of Next_Hop address value with zero.
      Else
        3.2.3. Set the Next_Hop address value = Current cluster head address.
        3.2.4. Replace the first left most zero bit value of Next_Hop address with the
               corresponding receiver address value.
     End if
Step 4. The cluster head sends the packet to the address specified in the Next_Hop address.
Step 5. Repeat step – 3 to 4 until the packet reaches the destination.
Step 6. After the destination receives a RREQ packets, it can drop the packets if it came through
    a Wormhole link as follows:
    6.1. It first extracts the source and the destination address from the packet.
    6.2. Starts matching the two addresses and take the decision as follows:
Step 7.
    7.1. If (the level-1 id mismatches) then
        /*sender and receiver belongs to two different level-1 clusters*/
        7.1.1. Case 1: both the level-0 id and node id are non zero
        /*both of them are non cluster head nodes*/
```

The receiver calculates the level-2 and level-1 and level-0 cluster heads ids addresses from the source address. As a legal RREQ packet is suppose to pass through all the determined cluster heads, therefore, the destination node searches the entire routing path recorded in the RREQ packet for the respective cluster heads ids. Even if a single cluster head id is missing from the routing path in the packet, it means that the packet has come through some compromised path. In that case the packet is rejected by the receiver.

7.1.2. Case 2: only the node id is zero

/*sender is a level-0 cluster head*/

The receiver calculates the level-1 and level-2 cluster heads ids, and validates the route information stored in the packet using the procedure described in Case1. If the validation is successful then the receiver keeps the packet, otherwise it rejects it.

7.1.3. Case 3: both the level-0 id and node id are zero

/*sender is a level-1 cluster head*/

The receiver only calculates the level-2 cluster head id and validates the route information stored in the packet using the procedure described in Case1. If the validation is successful then the receiver keeps the packet, otherwise it rejects it.

7.2. Else if (the level-0 id mismatches AND the node id is non zero) then

/*sender and receiver belongs to two different level-0 clusters*/
Then the sender calculates only the level-1 cluster head id and validates the route information stored in the packet using the procedure described in Case1. If the validation is successful then the receiver keeps the packet, otherwise it rejects it.

7.3. Else if (the node id mismatches) then

/*sender and receiver belongs to same level-0 cluster*/
Then the sender calculates only the level-0 cluster head id using the procedure previously described. Then it rejects the RREQ packet that does not contain that id.

Step 8. After this the receiver sends a RREP packet through the valid reverse path contained in the packet which has come through the valid path.

Step 9. After the sender receives the RREP packet, a link is established between the sender and the receiver through the path contained in the RREP packet and then the data transmission continuous using the path.

Step 10. End.

7. CONCLUSION

The main advantage of our proposed method is that it is an avoidance technique and the receiver can detect that a packet has come through some compromised (Wormhole) path during the route discovery phase of the DSR protocol. So, it does not need another phase or a periodically checking for the existence of the Wormhole in the path during data transmission. Our proposed countermeasure unlike of its predecessors neither requires any special H/W nor tightly synchronized clocks. It also does not use any statistical analysis or data. It detects if there is a Wormhole during the route discovery phase of the DSR protocol and avoids this path during further communication. So, nodes do not need to monitor its neighbour behaviour during the data transmission, and also the detection process is carried out in the route discovery phase of the DSR so it does not require a separate phase for it.

REFERENCES

[1] Y. Hu , A. Perrig & D. Johnson, (2002) "Packet Leashes: A Defense against Wormhole Attacks in Wireless Networks", IEEE INFOCOM.

[2] Shang-Ming Jen, Chi-Sung Laih & Wen-Chung Kuo, (2009) "A HopCount Analysis Scheme for Avoiding Wormhole Attacks in MANET", Sensors (Basel), Vol. 9, No. 6, pp 5022-5039.

[3] H. S. Chiu & K-S. Lui, (2006), "DelPHI: Wormhole Detection Mechanism for Ad Hoc Wireless Networks", International Symposium on Wireless Pervasive Computing, Phuket, Thailand.

[4] C. Sun, K. Doo-young, L. Do-hyeon, & J. Jae-il, (2008) "WAP: Wormhole Attack Prevention Algorithm in Mobile Ad Hoc Networks," IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC), pp 343-348.

[5] D. B. Roy, R. Chaki, N. Chaki, (2009), "A new cluster-based Wormhole intrusion detection algorithm for mobile ad-hoc network", Journal of Network Security & Its Applications , Vol. 1, No. 1, pp 44-52.

[6] S. Banerjee, & K. Majumder, (2012), "A Comparative Study on Wormhole Attack Prevention Schemes in Mobile Ad-Hoc Network". Recent Trends in Computer Networks and Distributed Systems Security, Vol. 335, pp 372-384.

[7] J. Sucec, I. Marsic, (2002), "Clustering overhead for hierarchical routing in mobile ad hoc networks", INFOCOM, pp 1698-1706

AUTHORS

Subhashis Banerjee has received his B. Sc. (Honours) and M. Sc. degrees in Computer Science in the year 2009 and 2011 respectively. He has obtained M. Tech. degree in Software Engineering in the year 2013 from West Bengal University of Technology, Kolkata, India. He is presently working as a researcher at Machine Intelligence Unit, Indian Statistical Institute, Kolkata, India. He has published several papers in International journals and conferences.

Koushik Majumder has received his B.Tech and M.Tech degrees in Computer Science and Engineering and Information Technology in the year 2003 and 2005 respectively from University of Calcutta, Kolkata, India. He obtained his PhD degree in the field of Mobile Ad Hoc Networking in 2012 from Jadavpur University, Kolkata, India. Before coming to the teaching profession he has worked in reputed international software organizations like Tata Consultancy Services and Cognizant Technology Solutions. He is presently working as an Assistant Professor in the Dept. of Computer Science & Engineering in West Bengal University of Technology, Kolkata, India. He has published several papers in International and National level journals and conferences. He is a senior Member of IEEE.