Journal of Computer Applications ISSN: 0974 – 1925,
Volume-5, Issue EICA2012-4, February 10, 2012

Secure Cluster Head Election for Intrusion Detection in MANET

Sivaranjani V
Department of CSE
RMD Engineering College,
Chennai, India,
E-mail:shivaranjaniav@gamil.com

Rajalakshmi D
Department of CSE
RMD Engineering College,
Chennai, India,
E-mail:draji2008@gmail.com

Abstract - In this paper, Leader election is studied in the presence of selfish node for intrusion detection in Mobile Ad Hoc Networks (MANETs).To balance the resource consumption among all the nodes, the most cost-efficient leaders with the most remaining energy must be elected as leader.But, the selfish nodes may behave selfishly by lying about their energy level and avoid them being elected. To address the issue of selfishnode anauction based routingmechanism is proposed. This mechanism can encourage selfish node to behave honestly before and after election by providing incentives in the form of credits by credit based techniques. Auction mechanism can always elect the most energy remaining nodes as cluster head. Simulation result shows that auction mechanism can effectively prolong the overall lifetime in MANET.

Keywords:*Leader election, intrusion detection systems, auction mechanism , selfish node, credit based technique and MANET security.*

I. INTRODUCTION

Mobile ad hoc networks (MANETs) are autonomous distributed system that comprises a number of mobile nodes connected by wireless links forming arbitrary time varying wireless network topologies. Mobile nodes functions both as hosts and routers. As hosts, they represent source and destination nodes in the network while as routers they represent intermediate nodes between a source and a destination nodes in the network while as routers they represent intermediate nodes between a source and a destination providing store and forward services to neighboring nodes. Security in MANET is particularly difficult to achieve.

Intrusion detection is one of the main challenges in MANET. To overcome the problem of intrusion detection, a common approach is to divide the MANET into a set of clusters where each node belongs to at least one cluster. The nodes in each cluster elect a cluster head to serve as IDS for entire cluster.

The auction mechanism is designed such that it not only selects out the most remaining nodes but also encourage selfish node to behave normally. In this paper the mechanism ensures truth telling is a dominant strategy for nodes based on auction mechanism. This pushes the nodes with most energy be elected as cluster head. At the same time the incentives are provided by credits to encourage nodes to honestly participate in election.Most of the routing algorithm designed for MANET such as AODV and DSR is based on the assumption that every node forwards every packet. But in practice some of the nodes may act as selfish nodes.

These nodes use the network and its services but they do not cooperate with other nodes. Such selfish nodes do not consume any energy such as CPU power, battery and bandwidth for retransmitting data of other nodes and they reserve them only for themselves. The original AODV and DSR algorithm can be modified to detect such selfish node. This paper discuss the credit based techniques used to detect selfish nodes in MANET. To address the issue of selfish node an auction based routing mechanism is proposed.

II. PROBLEM STATEMENT

The MANET is considered that each node has an IDS and a unique identity number. To elect the most cost efficient nodes as leader two challenges arise. First, the resource level is considered as private information. Second, the elected node will behave abnormally after the election process.

We consider MANET as an undirected graph G=(N,L).Cost of analysis $c_j$is used to reflect the remaining energy of the node. The cost of analysis vector is denoted as C= {c1, c2…..c_n} where n is the number of nodes in the network. The election process is denoted as

$$C(k, i) = \begin{cases} 1 \text{ if a node votes for a } k \\ 0 \text{ otherwise} \end{cases}$$

The objective of minimizing the global cost of analysis while serving all the nodes can be expressed by the following Social Choice Function (SCF),

$$SCF = S(C) = \sum_{k \in \phi} \sum_{j \in CS_k} (v_j(c_j) - c_k))$$

Where ϕdenoted as the set of elected cluster heads, vj(cj) is the social surplus *j* obtained when it's analysis cost is cj; ckis the analysis cost of cluster head in cluster k; CSkis the cluster which its cluster head is node *k*.

III. MODEL

Auction Mechanism

Mechanism design is a subfield of microeconomics and game theory. Mechanism design uses game theory to achieve the desired goals. The game theory is used to study what could happen when independent players act selfishly. Mechanism design allows the game designer to define rules in terms of SCF such that players will play according to these rules. The resource consumption problem can be modeled using Mechanism design theory with an objective function that depends on private information of the players.

The problem is modeled as a game, where the nodes are the players. Each node keeps a private information $\theta_i$which is the type of *i*. The type set Θ_i= {Normal, Selfish}, it describes how each player values all possible outcomes. The set of strategies for player *i*is defined as *S*

i. As we just simplified the QA-VCG which use direct revelation mechanism, this allows Θ*i*=*Si*. Each player *i*has a quasi linear utility function,

$$U_i(\theta_i) = P_i - c_i$$

P_i is the payment of the selected node given by the mechanism. The player usually seeks to maximize $U_i(\theta_i)$. While player *i*following a type $\theta_i$function $U_i(\theta_i)$ reflects the amount of benefits achieved by player *i*. When playing the game, each node reveals its analysis cost c_i. The cost vector *C* is the input of our mechanism. The mechanism calculates its corresponding output $o = o(\theta 1, \ldots, \theta_n)$ and a payment vector $p = (p1, \ldots, p_n)$ for each input vector *C*. The payment is used to motivate players to behave in accordance with the social goals. Assuming that the total services provided by the cluster head is *X*. The total services will be distributed among all nodes voted for the cluster head according to nodes' reputation. If the reputation of *ni*is *Ri*, then the service *n i* is

$$service = \frac{R_i}{\sum_{k=1}^{n} R_k} \times X$$

Where *n* is the number of nodes in the network. This motivates nodes to cooperate on every election round.

Payment

We denote Ckas the range of node k's ckso that node k can choose one value in Ck. If Ckis the true analysiscost, then ckis truth-telling strategy. Vk(ck) is the social surplus of node k.

The payment is defined as:

$$P_j(\hat{c}_j|\bar{c}_j) = V(J) + \hat{c}_j - V(J_{-J}, \bar{c}_j)$$

Where,

$$V(J) = max \sum_{j=1}^{n} [v_k(\hat{c}_j) - \hat{c}_j] \qquad V(J_{-j}, \bar{c}_j) = \sum_{k \neq j} [V_k(\hat{c}_k) - \hat{c}_k] + [V_j(\bar{c}_j) - \bar{c}_j]$$

$V(J–j, \bar{}cj)$ is the social surplus of the node when the analysis cost it reveals is $\bar{}cj$, but the other participant *k* shows $\hat{}ck$.

Cost of Analysis Function

To design the cost of analysis function two problems arise. First, energy level is considered as private information. Second, cost of analysis function is designed only in terms of nodes energy level. To solve these two problems cost of analysis is designed in terms of fairness and privacy. Fairness allows nodes with initially less resources to contribute and serve as leader in order to increase their reputation. Privacy is needed to avoid malicious use of the resource level. The cost of analysis is designed based on energy level E_i and the number of expected alive slot nT_i. So each node has a power factor,

$$PF=E_i/nT_i$$

Reputation of node *i*is defined as *Ri*, and Each node has a sample budget based on its reputation value.This is indicated by the percentage of sampling,

$$PS_i = \frac{R_i}{\sum_{i=1}^{n} R_i}$$

The *ci* notation represents the cost of analysis for a single packet and *Eids*is used to express the energy needed to run the IDS for one time slot. The analysis cost function is formulated as follows:

If $Ei<Eids$ $ci = \infty$, else

$$c_i = \frac{PS_i}{PF_i} = \frac{\frac{R_i}{\sum_{k=1}^{n} R_k} \times nT_i}{E_i}$$

IV. SELFISH NODE DETECTION

Credit Based Technique

The basic idea of Credit based schemes provides incentives for nodes to faithfully perform networking functions. In order to achieve this goal, virtual (electronic) currency or similar payment system may be set up. Nodes get paid for providing services to other nodes. When they request other nodes to help them for packet forwarding they use the same payment system for such services. Credit based schemes can be implemented using two models: The Packet Purse Model (PPM) and the Packet Trade Model (PTM).

An Auction Based AODV Protocol for Mobile AdHoc Networks with Selfish Nodes

In order to deal with selfishness, digital economy is created in the network. In this virtual economy, a source node has to pay some amount of a digital currency to intermediate nodes to have its packet forwarded, whereas the intermediate nodes bid and declare the amount of currency that they would request from the source if they forward the packet. These entire bids node chooses the route with the lowest bid. The source node sends the payment with every packet. Payment is set in a way that every node gets a payment that is greater than the amount that it bid. When nodes bid, they consider their energy level and the amount of currency that they have. Their bid increases when their energy level goes down, and decreases when their currency level goes down. Consequently, an auction based routing mechanism will help to deal with selfish nodes in the network.

An Auction based Routing Mechanism

This method proposes a 2-level Vickrey auction mechanism as follows.

Required Properties

(i) The routes should be selected according to the minimum cost, computed from the individual node bids;

(ii) The payment allocated to the winning route should be the one requested by the second smallest bidding route.

Further, the payments that the nodes in the winning route are getting should not explicitly depend on their requested bids, and should be larger than what they originally requested.

Bidding Formula

To model a realistic relationship among the node's bid, its energy, and its current currency level, authors have impose

Several Properties

• When a node's energy goes down and its currency stays the same, its bid should go up. This means that when a node has less energy, its willingness to forward packets for others will decrease.

• When a node's currency goes up and its energy goes down, its bid should go down, since its willingness to forward packets for others will not be as high as when it had less currency and high energy When both of a node's currency and energy go down, the decreasing energy will drive the bid up, whereas decreasing currency will drive the bid down. A bidding valuation computation formula is given as:b = a * (log (C) / E_r) Where,b: The bid value of the node,C: Node's current currency amount,Er: The node's energy ratio (Current energy / initial Energy).

V. ELECTION PROTOCOL

Leader election algorithm is proposed to elect the most cost efficient leader with less performance overhead. Moreover, we consider the addition and removal of nodes to/from the network due to mobility reasons. To design the leader election algorithm, the following requirements are needed: 1) To protect all the nodes in a network, every node should be monitored by a leader and 2) to balance the resource consumption of IDS service, the overall cost of analysis for protecting the whole network is minimized.

Leader Election Algorithm

The election algorithm uses four types of messages. Hello, used by every node to initiate the election process; 2Begin-Election, used to announce the cost of a node; Vote, sent by every node to elect a leader; Acknowledge, sent by the leader to broadcast its payment, and also as a confirmation of its leadership. The notations used in the algorithm are:service−table(j):If j is a cluster head then service − table(j) list the nodes jserves; reputation − table(j): The reputation table of node j records the reputation of other nodes;

Algorithm 1

1: each node sends *Begin*(*H*(*k*, *ck*));
2: ifeach node k received *Begin*(*H*(*k*, *ck*)) then
3: sends *Hello*(*IDk*, *costk*);
4: end if
5: if*n neighbor*(*k*) : *ci* <*cn*then

6: sends *Vote*(*k*, *i*);
7: *clusterhead– node*(*k*) = *i*;
8: end if
9: if *clusterhead*(*i*)=true then
10: Compute Payment, *Pi*;
11: Update service-table(*i*);
12: Update reputation-table(*i*);
13: Acknowledge=*Pi*+ all the votes;
14: Sends Acknowledge(*i*);
15: Launch IDS.
16**:** end

Initially, each node starts the election procedure by broadcasting a Hello message to all the nodes. This message contains the hash value of the node's cost of analysis and its unique identifier (ID). This message is needed to avoid cheating. Nodes from whom the Hello message has not received are excluded from the election. On receiving the Hello from all neighbors, each node sends Begin-Election as in Algorithm, which contains the cost of analysis of the node. If node k is the only node in the network or it does not have any neighbors, and then it launches its own IDS, the node k compares the hash value of Hello to the value received by the Begin-Election to verify the cost of analysis for all the nodes. Then, node k calculates the least-cost value among its neighbors and sends Vote for node. The Vote message contains the ID_k of the source node, the ID_i of the proposed leader, and second least cost among the neighbors of the source node cost. Then, node k sets node i as its leader in order to update later on its reputation. Note that the second least cost of analysis is needed by the leader node to calculate the payment. If node k has the least cost among all its neighbors, then it votes for itself . The elected node calculates its payment using and sends an Acknowledge message to all the serving nodes as in Algorithm.
Adding a new node

Algorithm 2

1. if (leader(k)= TRUE) then
2. Status =Costk;
3. else
4. Status =leadernode(k);
5. end if;
6. send Status(k,n);

When a new node is added to the network, it either launches its own IDS or becomes an ordinary node of any leader node. To include a new node to the IDS service, four messages are needed: Hello, Status, Join, and Acknowledge. Hello is sent by a new node n to announce its presence in the network. This Hello message is similar to the one presented in the previous section. Upon receiving the Hello, all the neighbors of the new node reply with a Status message. If the neighbor node k is a leader node, then the Status message contains its cost. On the other hand, if node k is an ordinary node, the Status message contains the ID of its leader node.

Removal of a Node

Algorithm 3

1. if (leadernode(k) =n) then
2. leadernode(k) =NULL;
3. update reputation(k);
4. send Begin _ Election as in Algorithm 1;
5. end if;
6. if (leader(k)TRUE) then
7. if (n _ service(k) then
8. update service();
9. end if;
10. end if;

When a node is disconnected from the network due to many reasons; such as mobility or battery depletion, then the neighbor nodes have to reconfigure the network. We assume that whenever a node dies, its neighbors are aware of it. At first, a Dead (n) message is circulated to all neighbors to confirm the removal of node n. On receiving the Dead (n) message, the neighbor node k checks whether node n is then its leader node or not. If node n is the leader node of node k, node k announces a new election and updates its reputation table.

Example for Leader Election

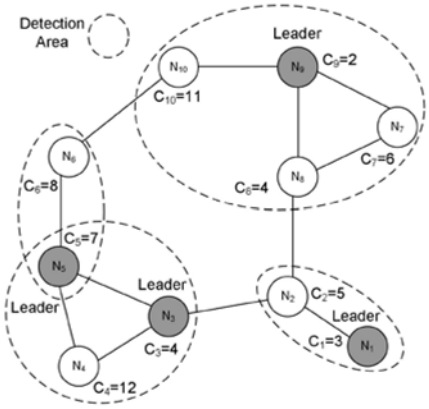

Figure 1: An Example of Leader Election

VI. CONCLUSION

The imbalance resource consumption for the IDS and the presence of selfish nodes motivated to propose an auction based routing algorithm to extend the survival time of nodes and the IDS and to prevent the nodes being selfish. The solution motivates all nodes in the network behave honestly to elect the least analysis cost nodes to handle the detection duty of the network. Also it gives incentives in the form of credits by credit based technique. The auction mechanism maintains the strategy-proof. The simulation results show that this model can prolong the survival time of the nodes.

ACKNOWLEDGMENT

I wish to express my sincere thanks to all the staff members of CSE Department, RMD Engineering College for their support and motivation throughout the work.

REFERENCES

1. N. Mohammed, H.Otrok, L. Wang, M. Debbabi, and P.Bhattacharya, "A Mechanism Design-Based Multi-Leader Election Scheme for Intrusion Detection in Manet," Proc. IEEE Wireless Comm. and Networking Conf. (WCNC), 2008.
2. Bin Yang, Jianhong Yang, JinwuXu, Deben Yang. Hybrid Cluster-head selected algorithm for wireless sensor network [*J*]. Application Research of Computers, 2008.4, 4(25): 1-3.
3. S. Vasudevan, J. Kurose, and D. Towsley, "Design and Analysis of a Leader Election Algorithm for Mobile Ad Hoc Networks," Proc. IEEE Int'l Conf. Network Protocols (ICNP), 2004.
4. Y. Zhang and W. Lee, "Intrusion Detection in Wireless Ad-Hoc Networks," Proc. ACM MobiCom, 2000.
5. H. Otrok, N. Mohammed, L. Wang, M. Debbabi, and P. Bhattacharya, "A Game-Theoretic Intrusion Detection Model for Mobile Ad-Hoc Networks," J. Computer Comm., vol. 31, no. 4, pp. 708-721, 2008.
6. Y. Huang and W. Lee. A cooperative intrusion detection systemfor ad hoc networks. In Proceedings of the 1st ACM Workshop Security of Ad Hoc and Sensor Networks, pages 135C147, Virginia, 2003. ACM.
7. K. Sun, P. Peng, P. Ning, and C. Wang, "Secure Distributed Cluster Formation in Wireless Sensor Networks," Proc. IEEE Computer Security Applications Conf. (ACSAC), 2006.
8. S. Vasudevan, B. DeCleene, N. Immerman, J. Kurose, and D. Towsley, "Leader Election Algorithms for Wireless Ad Hoc Networks," Proc. IEEE DARPA Information Survivability Conf. and Exposition (DISCEX III), 2003.
9. S. Vasudevan, J. Kurose, and D. Towsley, "Design and Analysis of a Leader Election Algorithm for Mobile Ad Hoc Networks," Proc. IEEE Int'l Conf. Network Protocols (ICNP), 2004.

V.Sivaranjanipursued B.E in Computer Science from Avinashilingam University, Coimbatore, in 2009 and currently pursuing M.E in Computer Science & Engineering from RMD Engineering College, Chennai. Area of specialization includes data privacy, economics of network security, and secure distributed computing. The conferences attended are National conference on research issues in computer science & engineering, International conference on signal & image processing. The workshops attended are Cloud computing workshop conducted by VMWARE SOFTWARE INDIA PVT, Mobile Computing Issues and Trends in Data Mining. I have done project in Digital-Invisible-Ink Data hiding based on Spread-Spectrum and Quantization Techniques during under graduation.

D.Rajalakshmi pursued M.E in Computer Science in RMK Engineering College. She is a gold medalist in Anna University ranking. She is currently working as Assistant Professor in RMD Engineering College. Research interest includes Data mining, Network security, and Data Structures.