A LOCALIZED ARCHITECTURE FOR DETECTING DENIAL OF SERVICE (DoS) ATTACKS IN WIRELESS AD HOC NETWORKS

Mieso K. Denko
Department of Computing and Information Science,
University of Guelph, Guelph, Ontario, N1G 2W1

Abstract: In this paper we propose a reputation-based incentive scheme for detecting DoS attacks that target the network layer services. The scheme is based on clustering architecture to provide localized and scalable solutions. It involves a node history-based reputation update mechanism where more weights are given to the most recent reputation values. Load balancing feature was introduced to reduce the forwarding overhead on cooperative nodes. We evaluated the performance of the proposed scheme using simulation experiments. We studied a network with selfish nodes where the attack involves dropping packets. The effect of dropping control and data packets is investigated with and without load balancing. The results indicated that localized reputation-based incentive solutions can significantly increase packet delivery ratio in the presence of selfish nodes with limited communication and packet processing overheads.

Key words: Clustering, DoS attacks, MANET, Reputation, Wireless Networks

1. INTRODUCTION

Lack of cooperation in Mobile Ad Hoc Networks (MANETs) can occur due to misbehaving nodes or lack of sufficient resources. Enhancing cooperation among nodes in the network can help in detecting and mitigating DoS attacks caused by misbehaving nodes. Misbehaving nodes can be malicious or selfish. Selfish nodes are nodes that participate in the network to maximize their own benefit by using the resources of the network while saving their own resources. Malicious nodes directly attack the network by disrupting its normal operation. Existing incentive mechanisms for enforcing cooperation can be classified into trade-based [1,2,4] and reputation-based [3,5,6,7] mechanisms. While the former uses a payment-based incentive, the latter uses mutual ratings based on the services they provide to each other.

While extensive work has been carried out on confidentiality, integrity and privacy attacks [15], the threat to network availability has received less attention. Existing studies on Denial of Services (DoS) attacks concentrate on the analysis of various attack scenarios targeting a specific layer [16], or propose a probing mechanism to detect misbehaving node targeting a specific network layer function [14]. While using a probing mechanism can help in detecting DoS attacks, probing packets may introduce communication overhead in the larger network. Reputation rating coupled with localized probing mechanisms can alleviate the problem.

In this paper we propose a reputation-based incentive mechanism for detecting DoS attacks targeting packet dropping. We use a clustering architecture to provide a localized monitoring mechanism and enhance scalability. The main contributions of this paper are: (a) it provides a localized and scalable architecture for reputation management in a distributed manner (b) it provides a node history-based reputation maintenance mechanism which gives more weights to the recent reputation ratings and; (c) a load balancing mechanism was introduced to reduce the traffic on heavily used cooperative nodes.

The rest of this paper is organized as follows. Section 2 presents the description of the proposed scheme. Section 3 presents some optimization mechanisms to improve the reputation management. Section 4 provides the results of the performance evaluation. Finally, Section 5 presents conclusions and future work.

2. THE PROPOSED SCHEME

The DoS attacks can be active or passive. Active DoS attacks can modify the routing information or data packets, disrupt the network operation, or disable services by flooding the network or causing sleep-deprivation attacks. Active attacks on network routing include dropping packets, overloading routing traffic, routing table overflow and flooding. The passive DoS attacks do not alter the data but may result in packet dropping.

The two main schemes used in handling DoS attacks are detection and prevention. The detection scheme involves locating the attacker and taking appropriate actions. Monitoring nodes activity or tracing the attacker can help in detecting a DoS attack source. Several tracing and monitoring mechanisms have been proposed in the literature [8,9,17]. The prevention mechanism thwarts the DoS attacks before the attack is launched. It does so by identifying the attack packet and taking action before it reaches the target to be attacked. Common mechanisms used on the Internet include ingress or egress filtering and route-based packet-filtering mechanisms.

2.1 Assumptions and goals

A reputation based incentive mechanism was proposed for detecting the DoS attacks in MANETs. The mechanism motivates nodes to cooperate and detect DoS attacks caused by selfish nodes. It involves cluster formation, reputation data maintenance and the use of this information for DoS attacks detection and improving network performance. We make the following assumptions for the correct functioning of our scheme: (a) each mobile node has a unique ID and can join or leave the network freely. (b) each node knows its one-hop neighbors and operates in a promiscuous mode. (c) nodes are selfish but rational.

2.2 An Overview of the Proposed Scheme

Most existing reputation systems for MANETs [1,6,7], use global reputation computation and maintenance mechanisms. Since monitoring and detecting DoS attacks is a difficult task in a larger network, it is essential to design a mechanism that helps in reducing packet processing and communication overheads. A more suitable management strategy in this environment requires use of a distributed solution. A clustering architecture provides a distributed and scalable architecture for network monitoring, reputation data management and topology control. The localized and distributed feature also reduces the storage and communication overhead, thereby optimizing network performance [10].

Our proposal is based on the incentive mechanism presented in [12] and uses clustering architecture for localized reputation management. However, it can be built on top of any reputation system that uses localized control and management. The novelty of our scheme is the use of clustering to reduce the reputation data management overhead and improve the monitoring capability. The global reputation maintenance schemes may provide more data for decision-making, however, such schemes have several shortcomings. First, maintaining reputation data at every node congests the network by requiring each node to process multiple packets. Second, the exchanged information traverses multiple intermediate nodes and may be lost or altered. Third, such schemes require global synchronization and also incur high storage and communication overhead. Fourth, global reputation computation and maintenance is not scalable.

We considered two categories of selfish nodes, namely, non-selective selfish nodes (denoted as type 0) and selective selfish nodes. The non-selective selfish nodes drop both control and data packets. There are two types of selective selfish nodes denoted as type 1 and type2. The type 1 selfish nodes participate in forwarding control packets but drop the data packets. The type 2 selfish nodes forward data packets but do not participate in forwarding the control packets.

2.3 Election of the RM

Each cluster has a RM, multiple nodes and gateways. A RM is a node that is responsible for allowing inter-cluster communications and probing misbehaving nodes. For cluster formation, we use an aggregate index (*I*), which takes the node stability (*T*) and Reputation rating (*R*) into account. The value of *I* is computed as follows:

$$I = \alpha_1 T + \alpha_2 R, \text{ where } \alpha_1 + \alpha_2 = 1 \quad (1)$$

A node is eligible to become a RM only if it possesses the maximum aggregate index (*I*) compared to all its neighbors. A Hello message is used to maintain connectivity information. The node stability is determined by monitoring its cluster membership changes. Since reputation rating is one of the criteria used for electing the RM, the chance of electing a selfish node as RM is low.

2.4 Localized Reputation Data Management

The reputation data management process involves the development of strategies for the computation, storage and dissemination of reputation data. To distinguish between new and existing nodes, we maintain and exchange information about the node's age. This eliminates punishing recently-joined nodes that have not built their reputation yet. When a new node joins the network, an initial reputation value is assigned and the node's status is labeled as new. Its status will be monitored and its reputation ratings will be adjusted based on the service it provides.

2.4.1 Reputation computation and maintenance

Global detection of selfish nodes is a challenging task in MANETs, observing one-hop neighbors makes the management task easier. In this approach, nodes in each cluster monitor the behavior of their neighbors and update the reputation ratings. This is achieved by implementing the Watchdog mechanism [5] at each node. A watchdog mechanism detects non-forwarding nodes by overhearing packet transmission from neighbors. It requires continuous monitoring by relying on a promiscuous mode of operation. The reputation information is assigned and maintained as follows. Each node maintains the reputation of its neighbors locally and reports it to the RM periodically. Whenever a node–say A, gets service from node B, it rates the service by assigning (+1) for satisfactory service and by assigning (-1) for unsatisfactory service. The reputation rating is not exchanged among non-neighbors but is reported to the RM periodically.

However, before assigning a negative rating (-1), a node makes multiple forwarding trials. If no response is obtained, (-1) is assigned and a new node is used for packet forwarding. The threshold time (k) for the forwarding trial is determined based on node mobility, link failure or network load. The value of k would be longer in the presence of higher node mobility, link failure or network load. At node level, the reputation rate is updated based on the node's own information. However, when there is a tie, or when a suspicious node is encountered, it uses the reputation maintained at the RM and combines it with its local reputation. The reputation rating of node B at node A is computed as the difference between the total number of packets forwarded and the total number of packets dropped, divided by the total number of packets received by the forwarding node. It is scaled to lie between –100 percent and 100 percent. The threshold value is experimentally determined to decide beyond which value a node is considered selfish or cooperative. At the RM, the average of the reputation rating of a node is computed based on the node's neighbors' reputation information.

2.4.2 Packet probing at the RM

Distinguishing selfish nodes from congested nodes helps in avoiding the punishment of cooperative nodes with depleted resources. It also helps in finding alternative routes for packet forwarding until the nodes can recover from failure. Although the reputation ratings maintained at each node can be used to determine non-cooperation, it is not sufficient to distinguish between selfish and faulty nodes. We use a probe packet sent by the RM to the node's neighbors to distinguish selfish nodes from faulty nodes. The RM requests reputation data from each member of its cluster as part of the probing activity. We call a node faulty if it is unable to participate in the network services because of lack of sufficient resources due to reasons such as power outage, the node's current position in the network, and software fault. For this purpose, we use the probing packets generated by the RM. It is generated based on request from the nodes or periodically based on the status of received reputation ratings from nodes. The probe packet is sent to all neighbors of the suspected node. Upon probing, to avoid false accusations, the decision to warn or suspend a node is made only if at least 50 percent of the suspected node's neighbors report the misbehavior. A node with a warning status can be reinstated if it continues to cooperate.

Based on the probing results, a node that does not respond to all its neighbors is considered faulty, while a node that responds to only some nodes is considered selfish.

The actions taken after detecting faulty nodes are different from those taken against selfish nodes. Based on the information received from the desired nodes, the RM will issue a warning message or suspension from services. When a node is detected to be selfish, it will be warned and isolated temporarily or permanently. For faulty nodes, there will be no penalty leading to warning or permanent isolation, however, its reputation rating will be reset to the threshold value given to new incoming nodes. Routes via these nodes will then be temporarily unused until they recover.

3. OPTIMIZATION MECHANISMS

3.1 History-Based Reputation Updates Mechanism

The proposed incentive mechanism was built on top of a clustering architecture where nodes in each cluster collaborate in the detection of selfish nodes. Forwarding packets originated from cooperative nodes and refusing those generated from selfish nodes can motivate cooperation. Selfish nodes are isolated from the network only if they fail to cooperate after it's a period of warning.

To prevent a node from misbehaving after achieving a certain high-level reputation in the network, we assign weights, while updating the reputation ratings with more weights. The process gives more weight to recent values and less weight to past values. Let R_c and R_o be the current and the past reputation ratings respectively. Then, the updated reputation rating (R_u) is updated as follows:

$$R_u = \alpha R_c + (1 - \alpha)R_o \qquad (2)$$

Where α is a configurable parameter lying between 0.5 and 1. The values of R_c and R_o are computed as described in section 2.

3.2 Load Balancing for Cooperating Nodes

When a node issues a query or forwards a packet, it uses the reputation ratings to bias its decision towards forwarding data through more cooperative nodes. Each node normally forwards a packet via a node with a higher reputation rating. However, such a mechanism procedure may lead to overloading more cooperative nodes. Load balancing (LB) is one of the main issues that require attention among cooperative nodes that willingly forward packets to others. Load balancing enables distribution of the network load equally among all potential forwarding nodes. We have used randomization as a means of distributing the load among nodes with higher reputation ratings.

We have implemented a probabilistic packet forwarding strategy among eligible nodes based on their reputation ratings. In this strategy, the forwarding task is accomplished probabilistically by choosing the next hop among all candidate nodes. This helps in balancing the load within the networks while overcoming the effect of packet dropping and selective forwarding. The basic steps for the load balancing procedure are: First, the source node selects a set (*S*) of nodes from its neighbors with reputation ratings above a threshold value; next, the source node sends a packet to a randomly selected node from the set *S*; the process then continues until the packet reaches its destination.

4. PERFORMANCE EVALUATION

4.1 Performance Metrics

The effects of the fraction of selfish nodes, network size and simulation time on the performance were investigated using the following five metrics.

1. **Average packet delivery ratio.** Defined as the ratio of the total number of data packets received by destinations to the total number of packets sent by the source.
2. **Communication overhead.** Defined as the ratio of the total number of routing and reputation related packets transmitted to the total number of packets transmitted including data packets.
3. **Processing overhead.** Defined as the ratio of processing overhead introduced by reputation system to the total processing overhead including route computation and maintenance.
4. **Selfish node detection rate.** Defined as the ratio of the total number of selfish nodes detected to the total number of selfish nodes in the network.
5. **False-positive ratio.** Defined as the ratio of well-behaving nodes wrongly classified as selfish nodes to the total number of well-behaving nodes in the network.

4.2 Discussion of Simulation Results

We carried out simulation experiments using NS-2 [11] with mobile nodes roaming in a 1000m x 1000m square area with a transmission range of 250 m. The percentage of selfish nodes in the network lies between 0 percent and 50 percent. The selfish nodes were randomly selected among 50-200 mobile nodes. The random waypoint mobility model [13] was used with an average speed of 10 m/s and pause time of 50 seconds. The communication pattern uses 20 Constant Bit Rate (CBR) traffic with a data rate of four packets per second. The Ad Hoc On Demand Distance Vector (AODV) [18] protocol was used for routing.

The simulation results are shown in Figures 1 to 6. The data points in the graphs are based on the average of 20 simulation runs. Figure 1 shows the average packet delivery ratio with and without load balancing as a function of the fraction of selfish nodes. The delivery ratio decreases with the increase in the fraction of selfish nodes for both cases but with consistently better performance when load balancing is used. The results confirm that the use of the probabilistic forwarding mechanism reduces congestion at nodes with good reputations by increasing the packet forwarding and improving the packet delivery ratio. The simulation results in Figure 2 show that the selfish nodes detection rate increases from 91 percent to 99 percent with 40 percent selfish nodes and from 86 percent to 97 percent with 20 percent selfish nodes. When the fraction of selfish nodes increases in the network, the probability of detecting them increases. This is because such a node can be a neighbor to at least one node and can easily be detected by these neighbors. However, as the simulation time increases, the detection rates for both scenarios become similar.

Figure 3 shows that the false-positive ratio is between 2 percent and 4.5 percent when 20 percent of the nodes in the network are selfish whereas the ratio is between 2.3 percent and 5 percent when 40 percent of the nodes are selfish. This implies that misclassification increases relatively with both network size and fraction of selfish nodes. Cooperative nodes can be classified as selfish due to reasons such as packet loss caused by link failure or congestion. Mobility also results in misclassification of nodes. Figure 4 shows the simulation results of communication overhead as a function of the fraction of selfish nodes. The results indicate that the communication overhead increases slightly with an increase in the fraction of selfish nodes.

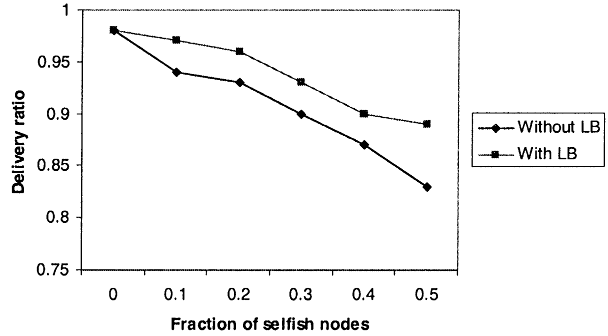

Figure 1. Packet delivery ratio with 100 mobile nodes

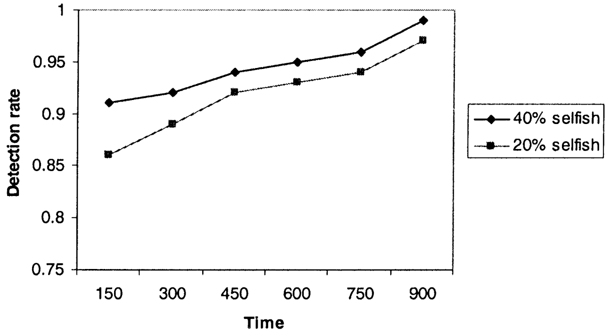

*Figure 2.*Detection rate with 100 mobile nodes

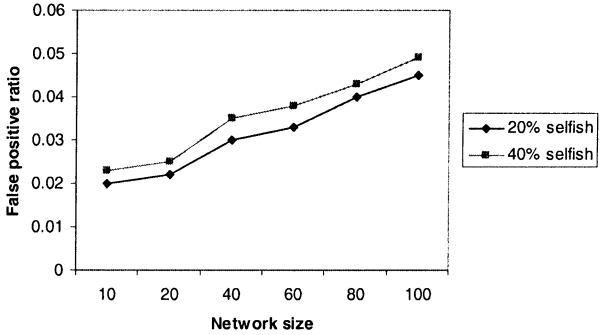

Figure 3. False-positive ratio with 100 mobile nodes

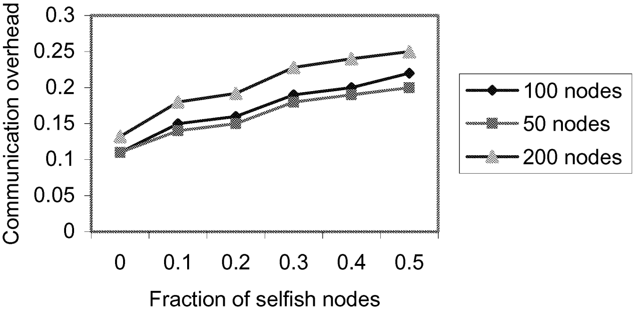

Figure 4. Communication overhead with 50, 100 and 200 mobile nodes

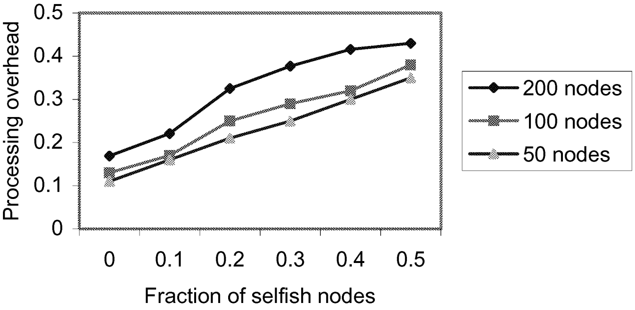

Figure 5. Processing overhead with 50, 100 and 200 mobile nodes

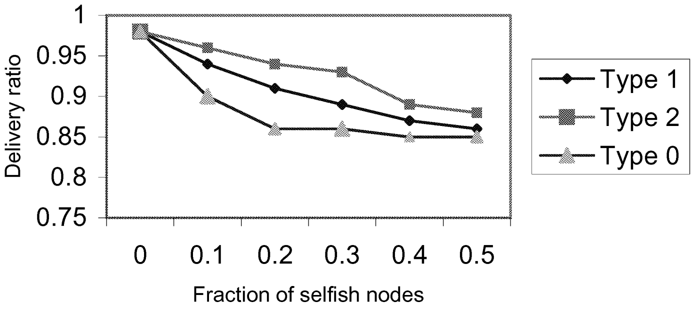

Figure 6. Packet delivery ratio with 200 mobile nodes

Little difference was observed, however, between the networks of size 50 and 100 nodes. The higher the percentage of selfish nodes, the slower the rate of increase, for larger network sizes. This implies that the use of clustering as a localized reputation data management scheme has introduced scalability and reduced communication overhead.

Figure 5 shows the results of packet processing overhead for three different network sizes. The packet-processing overhead increases slightly with an increase in the fraction of selfish nodes and network size. There is, however, a slight difference between the networks of size 50 and 100. The difference between the overheads caused by the simulated network sizes decreases slightly with an increase in the percentage of selfish nodes. The overall results indicate that the clustering architecture is effective in reducing the packet-processing overhead. Figure 6 shows the average packet delivery ratio for the three classes of selfish nodes as a function of the fraction of selfish nodes. The use of the probabilistic forwarding mechanism reduces congestion that could occur at cooperative nodes by introducing load balancing at each node. Both type 1 and type 2 selfish nodes have less effect on the delivery ratio than type 0 selfish nodes. However, the difference between the effects of the three classes of selfish nodes decreases slightly with an increase in the fraction of selfish nodes. This is partly due the possibility of direct communication between source and destination pairs. The little difference between the effects of type 0 and type 1 selfish nodes on packet delivery ratio suggests that the packet forwarding function is more crucial in improving the packet delivery ratio. Thus, a mechanism that enables selfish node to perform only the route request or reply operations correctly does not guarantee that the packet forwarding function will be properly performed.

5. CONCLUSIONS AND FUTURE WORK

In this paper we proposed a reputation-based incentive mechanism for detecting DoS attacks in MANETs. A clustering architecture was proposed for performing reputation data management in a localized and distributed manner. The node's reputation ratings and stability were taken into account for electing the RM. Load balancing mechanism was proposed to reduce the traffic on heavily used cooperative nodes. We have used the simulation technique to evaluate the network performance in the presence of selfish nodes. Our simulation results indicated that the reputation-based incentive mechanism is effective in tackling DoS attacks that occur due to selfish nodes. We will continue to investigate the performance of our incentive mechanism for tackling the DoS attacks by incorporating security mechanisms to improve network performance further. Our future work will also include comparisons of our scheme with existing similar schemes.

Reference

1. L. Buttyan, and J. Hubaux, Stimulating cooperation in self-organizing mobile ad hoc networks. ACM/Kluwer Mobile Networks and Applications (MONET) 8 (2003).
2. M. Baker, E. Fratkin, D. Guitierrez, T. Li, Y. Liu, and V. Vijayaraghavan, Participation incentives for ad hoc networks. http://www.stanford.edu/~yl31/adhoc (2001).
3. D. Barreto, Y. Liu, J. Pan, and F. Wang, Reputation-based participation enforcement for adhoc networks. http://www.stanford.edu/~yl314/adhoc (2002).
4. S. Zhong, J. Chen, and Y.R. Yang, Sprite - A simple, cheat-proof, credit-based system for mobile ad-hoc networks. Technical Report 1235, Department of Computer Science, Yale University (2002).
5. S. Marti, T.J. iuli, K. Lai, and M. Baker, Mitigating routing misbehavior in mobile ad hoc networks. In: Mobile Computing and Networking. (2000) 255–265.
6. S. Buchegger and J.Y.L Boudec, Performance Analysis of the CONFIDANT Protocol: Cooperation Of Nodes — Fairness In Distributed Ad-hoc NeTworks. In: Proceedings of IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHOC), Lausanne, IEEE (2002) 226–236.
7. P. Michiardi, and R. Molva, Making greed work in mobile ad hoc networks. Technical report, Institute Eur´ecom (2002).
8. A. Kuzmanovic, and E.W. Knight, Low-Rate TCP-Targeted Denial of Service Attacks. SIGCOMM'03, August 2003.
9. A.D. wood, and J.A. Stankovic, Denial of Service in Sensor Networks. IEEE October 2002.
10. W. R. Heinzelman, A.Chandrakasan, and H. Balakrishnan, Energy efficient communication protocol for wireless micro-sensor networks. Proceedings of IEEE Hawaii Int. Conf. on System Sciences, January 2000.
11. S. McCanne, and S. Floyd, Network Simulator. http://www.isi.edu/nsnam/ns/.
12. M.K. Denko: An Incentive-Based Service Differentiation in Mobile Ad Hoc Networks. IEEE International conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2005), August 2005, Montreal, Canada, to appear.
13. D.B. Johnson, and D.A. Maltz, Dynamic Source Routing in Ad hoc Wireless Networks'. In *Mobile computing* pages 153-181. Kluwer Academic Publishers,1996.
14. M. Just, E. Kranakis, and T. Wan, Resisting Malicious Packet Dropping in Wireless Ad Hoc Networks. Proc. of ADHOCNOW'03, Montreal, Canada.
15. I. Aad, J.P, Hubaux, and E.W. Knightly, Denial of Service Resilience in Ad Hoc Networks. ACM MOBICOM 2004, Philadelphia, PA, USA.
16. V. Gupta, S. Krishnamurthy, and M. Faloutsos, Denial of Service Attacks at the MAC Layer in Wireless Ad Hoc Networks. In Proc. of MILCOM, 2002.
17. A. Habib, M. H. Hafeeda, and B. Bhargava: Detecting Service Violation and DoS Attacks. Proc. of Network and Distributed System Security Symposium (NDSS), 2003.
18. C.E. Perkins, Ad hoc On-Demand Distance Vector (AODV) Routing, Internet Draft 17 February 2003.