International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:10, No:8, 2016

Misleading Node Detection and Response Mechanism in Mobile Ad-Hoc Network

Earleen Jane Fuentes, Regeene Melarese Lim, Franklin Benjamin Tapia, Alexis Pantola

***Abstract*—**Mobile Ad-hoc Network (MANET) is an infrastructure-less network of mobile devices, also known as nodes. These nodes heavily rely on each other's resources such as memory, computing power, and energy. Thus, some nodes may become selective in forwarding packets so as to conserve their resources. These nodes are called misleading nodes. Several reputation-based techniques (e.g. CORE, CONFIDANT, LARS, SORI, OCEAN) and acknowledgment-based techniques (e.g. TWOACK, S-TWOACK, EAACK) have been proposed to detect such nodes. These techniques do not appropriately punish misleading nodes. Hence, this paper addresses the limitations of these techniques using a system called MINDRA.

***Keywords*—**Mobile ad-hoc network, selfish nodes, reputation-based techniques, acknowledgment-based techniques.

I. Introduction

A MANET consists of mobile devices, called nodes, which form a temporary wireless network without a pre-existing network infrastructure, allowing for communication within a limited transmission range [5]. Furthermore, nodes can join and leave the network anytime. This makes MANET a flexible network. Thus, it is mostly used in remote areas where there is no existing network infrastructure or in disaster relief situations, search and rescue operations, vehicular networks, casual meetings, campus networks, robot networks, and so on [1]. Nodes in a MANET heavily rely on resources such as memory, computing power, and energy [6]; hence, it is beneficial that every node in the network participates in data forwarding, regardless of who the sender or the receiver is. If such cooperation is present in every node in the network, then it would be an ideal situation. However, nodes are not always cooperative. There are nodes that become uncooperative due to limited resources [6], or other circumstances [3]. Uncooperative nodes can either be misbehaving or malicious nodes. A selfish node, classified as a misbehaving node, uses the network only if it is beneficial to them. They send their packets, but refuse to forward packets for other nodes to conserve their resources. These selfish nodes can be further classified into three types, nodes that mislead other nodes in a sense that they make other nodes think that they are cooperative nodes (misleading nodes), nodes that do not participate in route discovering nor in packet forwarding (non-participative nodes), and lastly, nodes that misbehave differently according to energy levels (energy level-based nodes).

Earleen Jane Fuentes, Regeene Melarese Lim, Franklin Benjamin Tapia, Alexis Pantola are with the College of Computer Studies, De La Salle University, 2401 Taft Avenue, Manila, Philippines, 1004 (e-mail: ejane_fuentes@dlsu.edu.ph, regeene_lim@dlsu.edu.ph, franklin_tapia@dlsu.edu.ph, pantola@delasalle.ph).

II. Selfish Nodes

A selfish node is a misbehaving node that drops packets that are asked by the sender to be forwarded, which causes data not to be received by its destination across the MANET. These nodes aim to preserve its battery life and storage to fully utilize its limited resources most of the time. Furthermore, these nodes fail to share its resources such as CPU time, battery power, and memory space to other nodes.

A. Misleading Nodes

A misleading node is a selfish node that participates in the DSR Route Discovery and Route Maintenance phases, but is selective in forwarding data packets. These nodes participate in the network to mislead other nodes into thinking that they are cooperative nodes. Moreover, misleading nodes cause the Route Discovery process to initiate again or to find an alternate route to the destination, due to interruption of data flow when a node does not forward a packet. The alternate routes may still contain some of these misleading nodes, thus, the alternate routes will also fail. This process continues until the source of traffic concludes that data cannot be transmitted.

B. Non-Participative Nodes

A non-participative node is a selfish node that participates in neither the DSR Route Discovery, nor the Route Maintenance phases. They only use their energy for the transmission of their own packets to conserve their energy. These nodes do not pose a significant threat to the normal operation of the routing protocol because they are being ignored, even though they may degrade network performance.

C. Energy Level-Based Nodes

An energy level-based node misbehaves differently according to their energy levels. When their energy level ranges between a full energy to at least 80% energy, the node behaves properly. For an energy level of 50% to below 80%, it behaves like a misleading node. Finally, for an energy level lower than 50%, it behaves like a non-participative node.

Energy level-based nodes are detected only when they behave similarly to misleading nodes.

III. Existing Techniques

Several existing techniques aim to detect selfish nodes that are present in a MANET. The techniques can be monitoring techniques, acknowledgment-based techniques, and reputation-based techniques. Monitoring techniques are

merely for detection and do not punish selfish nodes. Acknowledgment-based techniques detect and avoid the nodes. These techniques can punish selfish nodes. However, they also punish the cooperative nodes in the network. Moreover, these techniques contribute large overhead to the network. Reputation-based techniques also detect and punish selfish nodes by denying the use of the network. These techniques use second-hand reputation, meaning, they rely on the information of other nodes, therefore, these techniques have high false positive rate (i.e. a cooperative node tagged as misleading).

A. Watchdog and Pathrater [5], [8]

The Watchdog proposed by Kachirski et al. [5] monitors and observes the nodes near the forwarding node to determine their misbehaviors. After a sending node has forwarded a packet to the next hop node, it monitors if the next hop node in the path has already forwarded the packet or not. If the node has not forwarded the packet, the node will be considered selfish. The Watchdog removes the selfish node from the network. In addition to the Watchdog, Kachirski et al. [5] proposed the Pathrater to select a reliable path, which is calculated for each node. A rating is maintained for each node. Each node computes the "path metric" for each node, and the path with the highest metric is selected as the reliable path so as to avoid paths with selfish nodes.

Although the Watchdog can determine the misbehavior of a node in the link and network layer, and the Pathrater's throughput increases with the increase in the mobility of a node, the Watchdog and Pathrater do not punish a selfish node because the sole purpose of these techniques is to detect selfish nodes, and avoid using paths with such nodes. Other existing techniques punish selfish nodes by refusing to forward their packets, similar to what acknowledgment-based techniques and reputation-based techniques are doing.

B. Acknowledgement-Based Techniques

Acknowledgment-based techniques detect and avoid selfish nodes; however, it can also unintentionally punish cooperative nodes. It can either be passive or active. Passive acknowledgment-based techniques monitor on promiscuous mode, thus, this technique may not be able to handle ambiguous collisions, partial dropping, and unidirectional links. On the other hand, active acknowledgment-based techniques explicitly send acknowledgment packets once a packet is received, thus, it costs more memory and generates large overhead.

Once misbehavior is detected regardless through passive or active approach, selfish nodes are punished by avoiding them and denying them of using the resources of the network.

1) TWOACK [2], [3]

The TWOACK proposed by Balakrishnan et al. [3] can be implemented as an addition to any routing protocol, such as the Dynamic Source Routing or DSR. The nodes use TWOACK packets, a special type of acknowledgement packet to acknowledge if the data have been successfully received by the destination. These acknowledgment packets travel two hops from the sending node. If the sending node does not receive the TWOACK packet, it is claimed that the next-hop's forwarding link is misbehaving. However, this constant sending of acknowledgment packets contributes to traffic congestion, which degrades the overall performance of the network.

2) S-TWOACK [2], [3]

The Selective-TWOACK (S-TWOACK) has been proposed in order to resolve traffic congestion, which is the issue with regard to the TWOACK technique. In S-TWOACK, the nodes wait until there are three nodes that received the data packet before the third node sends a TWOACK packet to the first node. This results in sending just one TWOACK packet that acknowledges several received data packets.

Acknowledgment-based techniques have high false positive rate because they detect a link that contains a selfish node, and avoids the link. Hence, the cooperative nodes in the link are affected as well.

C. Reputation-Based Techniques

A node is responsible for observing the relaying of a packet to a nearby node and acquiring the status of the other nodes from a consolidated node in the network. The reputation of a node increases when it forwards the data it receives to other nodes and concurrently decreases when it chooses to do otherwise. After a node's reputation decreases and goes below the threshold that is defined by the developer, the node is punished or disregarded from the network.

1) CONFIDANT [8]

Cooperation of Nodes: Fairness in Dynamic Ad-hoc Networks (CONFIDANT) proposed by Buchegger and Boudec [4] uses both direct and second-hand reputation. The Monitor module uses a Watchdog-based mechanism in observing its neighbors. A copy of sent packets which are used for rating the nodes is kept. The Reputation Manager module uses the accumulated ratings to compare against a certain threshold to determine the selfish nodes. The Trust Manager module sends out an ALARM message to its neighbors, and these ALARM messages are validated by the neighbors. In addition, a node re-evaluates its paths to remove the selfish node and deny it from using the network.

CONFIDANT only considers negative indirect ratings. Thus, it makes the mechanism vulnerable to slandering attacks, which are malicious acts done by a node to cause harm on cooperative nodes by intentionally decreasing the reputation of other nodes in the network.

2) CORE [8]

Collaborative Reputation Mechanism Enforce Node Cooperation (CORE) is a combination of reputation mechanism and a Watchdog mechanism proposed by Michiardi and Molva [9]. Each node maintains a reputation table, and the reputation associated to each node is computed using subjective, direct, and indirect observations from neighbor nodes.

In combination with the Watchdog mechanism, CORE sets a timeout when observing nodes, giving a negative rating to a node that does not send the packet within the timeout. When the rating of a node is below the specified threshold, it is denied of using the network.

CORE enforces cooperation in the network by making reputation difficult to build, therefore, discouraging misbehavior. Furthermore, it considers only positive ratings for indirect reputation. Thus, it is not susceptible to slandering attacks. However, this makes the mechanism vulnerable against self-promoting nodes, which consistently increase their reputation.

3) SORI

Secure and Objective Reputation-based Incentive (SORI) uses the concept of confidence being directly proportional to the packets sent. A node's confidence with another node increases as the packets that it sends increases. SORI uses the credibility of a node, which is calculated using the ratio between the packets sent by node A to node B and packets B sent for A, which A detected, and the corresponding confidence value. The credibility is used to compare to a certain threshold to determine if a node is selfish. The node deemed to be selfish is denied of services.

SORI is integrated with One-Way-Hash Chain to prevent selfish nodes from impersonating nodes with good reputation. Although computationally efficient, it is unable to distinguish malicious nodes and selfish nodes [5]. Moreover, false positive rate is partially restricted in SORI because reputation is only sent to one-hop neighbors of a node. However, the simplicity of the algorithm of SORI results in the poor performance within a cooperative environment.

4) LARS

Locally Aware Reputation System (LARS) proposed by Hu is a stand-alone reputation-based scheme. A node observes its one-hop neighbors using a Watchdog-based mechanism: increasing the rating if a positive event is detected; otherwise, the rating is decreased. In order to discover the selfish node, a trace process occurs: the source node sends out a trace packet, which will be sent out along the same path where selfish behavior is observed. The neighbors of the receiving nodes will verify the participation of the node by sending an acknowledgment; the node whose neighbors do not send an acknowledgement will be determined to be selfish. The selfish node will be excluded from the network, but can return after the timeout.

In contrast to SORI, LARS uses a more complex algorithm, particularly for tracing the selfish node. This process in LARS involves sending additional packets resulting to high-energy consumption and contributing to the overhead of the network [7].

5) OCEAN

Observation-based Cooperation Enforcement in Ad-hoc Networks (OCEAN) is a stand-alone selfish node detection scheme proposed by Bansal and Baker. The NeighborWatch uses a Watchdog-based mechanism for observing neighbors: decreasing the ratings if the neighbor does not forward the packet within the timeout or does not forward it at all; otherwise, increases it. OCEAN modified the Dynamic Source Routing (DSR) protocol such that the Route Request (RREQ) packet will contain another field, the avoid list - containing the faulty list of every node that forwards the RREQ; the nodes in this list are either avoided or denied of services. The Second Chance mechanism removes nodes from the faulty list after a specified unit of time has passed.

The algorithm of OCEAN is simpler than LARS and does not involve sending out additional packets. Furthermore, OCEAN is tested against a second-hand reputation technique and performed fairly well even with local information only. However, OCEAN is unable to appropriately punish misleading nodes, thus allowing misleading nodes to maintain good throughput [7].

IV. MINDRA

Misleading Node Detection and Response Mechanism in MANET (MINDRA) is a reputation-based and a time-based technique that aims to solve the limitation of OCEAN by detecting and punishing misleading nodes. MINDRA consists of the Path Manager module, the Sender module, the Forwarder module, the Punishment module, the Receiver module, the Second-Chance module, and the Detection module.

The Path Manager module checks if the source node has a path to the destination node. The Sender module is responsible for sending packets that originate from the source node. The Forwarder module is responsible for sending packets that do not originate from the source node. The Punishment module of the next hop node receives the packet from the node and checks if the node is misleading or not, and is responsible for dropping packets that come from misleading nodes. The Receiver module acquires different types of packets, namely, DSR packets and data packets permitted by the Punishment module. The Second-Chance module keeps track of the nodes that are tagged as misleading. The Detection module is comprised of the Watchdog submodule and Assessment submodule. The Watchdog submodule monitors the next hop node. The Assessment submodule considers the following metrics: Drop Streak, Drop History, and Favor Ratio.

Most of the modules of MINDRA are very similar to the implementation of the modules of OCEAN. The Assessment submodule is the main difference between MINDRA and OCEAN.

The Assessment submodule uses the Drop Streak, Drop History and Favor Ratio for assessing if a node is misleading or not. The Assessment submodule considers the Drop Streak of a node as an indication of its misleadingness. The Drop Streak is directly proportional to the misleadingness of a node. When the Drop Streak is broken and resets, the behavior of the node in the previous time frame is considered. The ratio between the packets the dropped in the previous time frame and the number of packets it is requested to forward in that time frame quantifies its misleadingness. The Drop History is merely incrementing as the node sends its packets. The Favor

Ratio verifies this behavior and also gives consideration in the event that the node has a malfunctioning network card. If there is no activity in the previous time frame, misleadingness will be based on the Favor Ratio. Table I shows the values of the metrics that are used in the Assessment submodule.

TABLE I
METRICS AND THEIR CORRESPONDING VALUES

Metric	Value
Drop History Count Reset	3
Forward Request History Count Reset	3
Punishment Timeout	(No. Of Times Tagged) * 2 seconds
Watch Expiration	1 second
Watch List Check Intervals	1 second
Drop Streak Threshold	150
Drop History Threshold	0.8
Favor Ratio Threshold	0.5
No. Of Packets Needed To Be Assessed	10

V. RESULTS AND ANALYSIS

To evaluate the performance of MINDRA, it is tested against the existing DSR. The Packet Delivery Ratio Test is performed.

Packet Delivery Ratio (PDR) is the percentage of packets received by cooperative nodes relative to packets sent by cooperative nodes. In a MANET, as the number of misleading nodes increases, PDR is expected to decrease. The PDR test assesses the effectiveness of MINDRA in reducing the effect of misleading nodes.

The test set-up involves a simulation with 50 nodes using purely DSR (i.e., MINDRA is disabled). The percentage of misleading nodes is increased in each run of the simulation. As an example, simulation is run with 0% misleading nodes and PDR is measured. The number of misleading nodes is adjusted to 10% and PDR is again measured. This procedure is repeated until there are 90% misleading nodes. The same set-up and procedure is replicated but with MINDRA enabled.

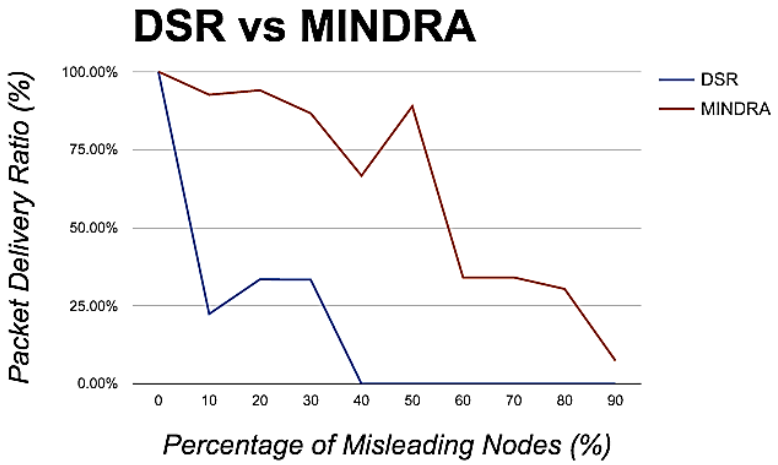

Fig. 1 Packet Delivery Ratio of DSR and MINDRA

The test is conducted on three different network topologies. For each percentage of misleading nodes, three tests are conducted. Thus, three different sets of misleading nodes are used for testing. As shown in Fig 1., as the percentage of misleading nodes increased, a MANET with MINDRA performs with a higher Packet Delivery Ratio than purely DSR, which indicates that it performs better.

The Packet Delivery Ratio should consistently decrease as the percentage of misleading nodes increase. A possible explanation as to why the Packet Delivery Ratio increases at some point at a much higher percentage of misleading node is that the misleading node is not part of the path. As a result, it is not able to drop the packets, which leads to a higher Packet Delivery Ratio.

VI. CONCLUSION

In MANET, it is assumed that all nodes are cooperative; however, in reality, misleading nodes are present. These nodes are selective in packet forwarding to conserve resources. Thus, these nodes degrade the network performance of the MANET. MINDRA is a system that detects misleading nodes to appropriately punish these nodes to maintain the network performance of the MANET. In this case, MINDRA is tested against purely DSR and shows that as the percentage of the misleading nodes in a MANET increases, the performance of MINDRA is better than purely DSR since it results in a higher Packet Delivery Ratio as compared with purely DSR.

REFERENCES

[1] Abbas, S., Merabti, M., and Llewellyn-Jones, D. (2010). A Survey of Reputation Based Schemes for MANET. (Online) Available: http://www.cms.livjm.ac.uk/pgnet2010/MakeCD/Papers/2010046.pdf (Jan. 30, 2015).

[2] Balakrishnan K., Deng, J., Liu, K. and Varshney, P. K. "An acknowledgment-based approach for the detection of routing misbehaviour in MANETs," IEEE Trans. Mobile Comput., vol. 6, no. 5, pp. 536–550, May 2007.

[3] Balakrishnan, K., Deng, J., and Varshney, P.K." TWOACK: Preventing selfishness in Mobile Ad Hoc Networks," Proc. IEEE Wireless Comm. and Networking Conf. WCNC 2005, March 2005. (Jan. 24, 2015).

[4] Buchegger S., Le Boudec, J. (2002). "Performance analysis of the CONFIDANT protocol (Cooperation of nodes dairness in dynamic ad-

hoc network)", in Proceeding 3rd ACM (MobileHoc'02). pp 226-336. (Jan. 26, 2015).

[5] Kachirski, O., Guha, R. (2003). "Effective Intrustion Detection Using Multiple Sensors in Wireless Ad Hoc Networks", in Proceeding IEEE, (HICSS'03), pp 55.1. (Jan. 24, 2015).

[6] Kargl, F., Klenk, A., Schlott, S., and Weber, M. Advanced Detection of Selfish or Malicious Nodes in Ad Hoc Networks. In *Proceedings of 1st European Workshop on Security in Ad-Hoc and Sensor Networks (ESAS 2004)* (Sept. 2004), Springer Verlag, pp. 152–165.

[7] Karthik, M., John, J. (2013). "A Survey of Techniques Used To Detect Selfish Nodes in MANET". *International Journal for Scientific Research & Development.* (Online). 1(4), pp. 1029-1032. Available: http://www.ijsrd.com/articles/IJSRDV1I4052.pdf (Jan. 6, 2015).

[8] Koroupi, F., Kuchaki, M., and Movaghar, A. (2008). Investigating Intrusion Detection Systems in MANET and Comparing IDSs for Detecting Misbehaving Nodes. World Academy of Science, Engineering and Technology, 44, pp. 351-355. Available: http://waset.org/publications/6483/investigating- intrusion-detection-systems-in-manet-and-comparing-idss-for-detecting- misbehaving-nodes (Jan.27, 2015).

[9] Michiardi P., Molva, R. (2002). "Core: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks", in International Conference on (CMS'02). (Jan. 26, 2015).