Recognition of Malignant Nodes in Wireless Ad-hoc Network based on Auto-Correlation function

Ambika V[1],Rummana Firdaus[2]

[1]*Assistant professor, Department of Computer Science & Engineering,*
GSSSInstitude of Engineering and Technology for Women,
KRS Road, Metagalli,Mysuru,Karnataka, India
ambikav@gsss.edu.in

[2]*Assistant professor, Department of Computer Science & Engineering,*
GSSSInstitude of Engineering and Technology for Women,
KRS Road, Metagalli,Mysuru,Karnataka, India
rummana@gsss.edu.in

***Abstract*— Preserving privacy of data and securing access to data from unauthorized nodes in wireless ad hoc network have attracted a lot of research and development effort in past few years. In a multi-hop wireless network, nodes cooperate in relaying traffic. An adversary can exploit this cooperative nature to launch attacks. Malicious nodes that are part of the route exploit their knowledge of the communication context to selectively drop a small amount of packets critical to the network performance. The data can be replicated by the malignant nodes in network. The data is error prone if certainty is not assured while packets are dropped. The common reason for packet dropping in wireless ad hoc network is considered as the channel error. The packet loss may also be due to malicious nodes. Identifying if the packet loss is intentional or unintentional is needed before computing packet loss rate. The Auto-Correlation function based on position of packet loss is taken to account to detect attacker. The truthfulness of detection is verified by auditing mechanism based on Homomorphic Linear Authenticator(HLA) cryptographic primitive. So the privacy of data is maintained while auditing and ensures low transmission, storage overheads and reduces resource contention. To keep computation simple, packet-block based detection mechanism is considered. Thus the implementation is useful to avoid packet dropping attack in Wireless Ad hoc Network.**

Keywords: **Node misbehavior, schemes, routing path, HLA.**

1. INTRODUCTION

A wireless network works in either Infrastructure mode or Ad-hoc mode. In infrastructure mode, the nodes communicate via access points whereas in Ad-hoc, network the nodes communicate directly peer-to-peer without centralized administration. Since the nodes in Ad-hoc network rely on neighbor nodes to transmit the data, it poses a security challenge.

The vulnerable nature of the wireless network provides a scope for the malicious nodes to disrupt the routing of packets. Due to congestion in the network, there may be packet dropping. The packet dropping can be either intentional or unintentional. The harsh channel conditions such as link error, noise etc may be source for the unintentional packet dropping. In a multi-hop wireless network, malicious node can exploit the cooperative nature to launch attacks in the network intentionally. The malicious node may first pretend to be a cooperative node in the route discovery process. After it is included in a route, the malicious node starts dropping packets affecting the performance of the network. In some cases, the node may stop forwarding every packet it receives.[4] This is because the node is selfish to save it own credits and power and it can also paralyze the network by partitioning its topology. A malicious node that is part of the route can exploit its knowledge of the network protocol and the communication context to launch an insider attack—an attack that is intermittent, but can achieve the same performance degradation effect as a persistent attack at a much lower risk of being detected. Specifically, the malicious node may evaluate the importance of various packets, and then drop the small amount that is deemed highly critical to the operation of the network. Such misbehavior has been shown to have a severe impact on the network operability [1-4]. Therefore, the malicious node has to be identified and eliminated from the routing path.

2. LITERATURE SURVEY

Methods for addressing the misbehavior problem can be classified into, (a) credit-based systems[5-8], (b) reputation-based systems [10-14], and (c) acknowledgment-based systems [16-18].

2.1 Credit- based scheme

Credit systems provide incentives for cooperation. Buttyan et al. [5] proposed a system where nodes receive credit (nuglets) for packets they forward, and spend credit to transmit their own packets. Each node maintains a counter termed nuglet counter. The counter is decreased when the node sends packets of its own, but increased when it forwards packets for the other nodes. The counter should be positive before a node is allowed to send its packet. Therefore, the nodes are encouraged to continue to help other nodes. When the nuglet counter is 0 node is not allowed to send its own packets until it gains nuglets by forwarding the other packets thus increasing the credit. Tamper resistant hardware modules are used to keep nodes from increasing the nuglet counter illegally.

Zhong et al. [7] proposed Sprite, where nodes collect *receipts* for packets they forward. Nodes upload receipts are uploaded to a Credit Clearance Service (CCS) in return for credit to transmit their own packets.

Jakobsson et al. [8] proposed a scheme where cryptographic payment tokens are attached to packets and managed by a base station, i.e., a form of virtual bank.

While credit-based systems motivate cooperation, malicious nodes have no incentive to collect credit and receive no punishment for non-cooperation. Furthermore, tamper proof hardware is currently too expensive to integrate in every network device [9]. Sprite removes this requirement, at the expense of a CCS. Lastly, credit systems lack a mechanism for identifying the misbehaving node(s) for revocation.

2.2 Reputation based scheme

Reputation system use neighborhood monitoring techniques to identify misbehaving nodes.
Marti et al. [14] proposed a scheme which relies on two modules, a *watchdog* and a *pathrater*.

The watchdog module monitors the behavior of neighboring nodes by operating its radio in promiscuous mode to verify packet forwarding, making accusations of misbehavior when packets are not forwarded.

Figure 2.2 illustrates how the watchdog works. Suppose there exists a path from node S to D through intermediate nodes A, B, and C. Node A cannot transmit all the way to node C, but it can listen in on node B's traffic. Thus, when A transmits a packet for B to forward to C, A can often tell if B transmits the packet and can also tell if B has tampered with the payload or the header.
Nodes operate in a promiscuous mode wherein; the watchdog module overhears the medium to check whether the next-hop node faithfully forwards the packet. At the same time, it maintains a buffer of recently sent packets. A data packet is cleared from the buffer when the watchdog overhears the same packet being forwarded by the next-hop node over the medium. If a data packet remains in the buffer for too long, the watchdog module accuses the next-hop neighbour to be misbehaving.

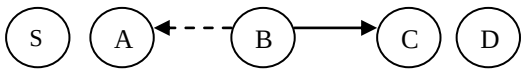

Fig 2.2: When B forwards a packet from S toward D through C, A can overhear B's transmission as it is in transmission range of B and so it can verify that B has attempted to pass the packet to C. The solid line represents the intended direction of the packet sent by B to C, while the dashed line indicates that A is within transmission range of B and can overhear the packet transfer.

For a watchdog to work properly, it must have information as in where the packet should be among the two hops, which is provided by the routing protocol DSR. So, watchdog works best when it is on top of source routing protocol.

The pathrater module uses the watchdog's accusations to select paths that will most likely avoid misbehavior.
Buchegger et al. [10,11] proposed the *CONFIDANT* scheme, utilizing the watchdog/pathrater model, where detected misbehavior is broadcast using alarm messages. He et al. [13] proposed SORI, which propagates monitored behavior, thus relying on first and second-hand information.

Neighborhood monitoring becomes complex in cases of multi-channel networks or nodes equipped with directional antennas. Neighboring nodes may be engaged in parallel transmissions in orthogonal channels or different sectors, thus unable to monitor their peers. Moreover, operating in promiscuous mode requires up to 0.5 times the energy for transmitting a message [15], thus making message over-hearing an energy-expensive operation.

Finally, reputation-based systems are proactive in nature, requiring the constant
monitoring of nearby nodes for the building of reputation metrics. Hence, overhead is incurred on all nodes regardless of whether a misbehaving node exists.

2.3 Acknowledgment-Based Systems

Acknowledgment systems rely on acknowledgments to verify that packets were forwarded.
Liu etal. [16] proposed the 2ACK scheme, where nodes explicitly send 2-hop acknowledgments in the reverse direction
to verify the cooperation of the intermediate node in packet forwarding. A value is assigned to the quantity/frequency of un-verified packets to determine misbehavior. Padmanabhan et al. [16] proposed a method based on traceroute inwhich the source probes the path with pilot packets indistinguishable from data packets. Xue et al. [17] proposed Best effort Fault Tolerant Routing, relying on end-to-end ACK's to monitor packet delivery ratio and select routing paths that minimize misbehavior.

Acknowledgment-based schemes are proactive, and hence incur message overhead regardless of the presence of misbehavior. 2ACK provides a method to reduce message overhead by acknowledging only a fraction of the packets, with the tradeoff increased delay in misbehavior detection.

3. MODELS AND PROBLEM STATEMENT

Network Model: We assume a multi-hop ad hoc network consisting of N nodes. Each node is responsible for relaying messages from source S to destination D. We assume S is aware of nodes in path P_{SD} (Path to Source and Destination), as in Dynamic Source Routing (DSR) [13]. If DSR is not used, the source can identify the nodes in P_{SD} by performing a traceroute operation. For simplicity, we number the nodes in $P_{SD} = \{n1 \ldots\ldots n_k\}$in ascending order with k = |PSD|. Node n_i is upstream of nj if i < j and is downstream of n_j if i > j.

Adversarial Model: We assume the existence of multiple independently misbehaving nodes in P_{SD}. Any node in P_{SD} may be misbehaving, except the source and the destination which are assumed to be trusted. The goal of misbehaving nodes is to degrade throughput while remaining
undetected. Misbehaving nodes are assumed to be aware of the mechanisms used for misbehavior detection. The case of multiple colluding nodes is left as future work.

Problem Statement: Under the system and adversary models defined above, we address the problem of identifying the nodes on P_{SD} that
drop packets maliciously. We require the detection to be performed by a public auditor that does not have knowledge of the secrets held by the nodes on PSD. When a malicious node is identified, the auditor should be able to construct a publicly verifiable proof of the misbehavior of that node. The construction of such a proof should be privacy preserving, i.e., it does not reveal the original information that is transmitted on PSD. In addition, the detection mechanism should incur low communication and storage overheads, so that it can be applied to a wide variety of wireless networks.

4. PROPOSED DETECTION SCHEME

The proposed mechanism is based on detecting the correlations between the lost packets over each hop of the path. The basic idea is to model the packet loss process of a hop as a random process alternating between 0 (loss) and 1 (no
loss). Specifically, consider that a sequence of M packets that are transmitted consecutively over a wireless channel. By observing whether the transmissions are successful or not, the receiver of the hop obtains a bitmap (a1... a_M)
where $a_j \epsilon \{0, 1\}$ for packets j= 1 . . .M. The correlation of the lost packet is calculated as the auto-correlation function of this bitmap.
Under different packet dropping conditions, i.e., link-error versus malicious dropping, the
instantiations of the packet-loss random process should present distinct dropping patterns (represented by the correlation of the instance). This is true even when the packet loss rate is similar in each instantiation.
To correctly calculate the correlation between lost packets, it is critical to enforce a truthful packet-loss bitmap report by each node. HLA cryptographic primitive is used for this purpose. The basic idea is as follows. An HLA scheme allows the source, which has knowledge of
the HLA secret key, to generate HLA signatures $s1 \ldots\ldots s_M$ for M independent messages $r1 \ldots. r_M$, respectively. The source sends out the r_i's and s_i's along the route. The HLA signatures are made in such a way that they can be used as
the basis to construct a valid HLA signature for any arbitrary linear combination of the messages.
Detection architecture consists of four phases: setup, packet transmission, audit, and detection.

Set up phase takes place right after route P_{SD} is established, but before any data packets are transmitted over the route.In this phase, S decides on a symmetric-key crypto-system ($encrypt_{key}$,$decrypt_{key}$) and K symmetric keys $key_1 \ldots. key_K$, where $encrypt_{key}$ and $decrypt_{key}$ are the keyed encryption and decryption functions, respectively. S securely distributes $decrypt_{key}$ and a symmetric key key_j to node n_j on P_{SD}, for j= 1.... . .K. Key distribution may be based on the publickey crypto-system such as RSA: S encrypts key_j using the public key of node nj and sends the cipher text to n_j. n_j decrypts the cipher text using its private key to obtain key_j. S also announces two hash functions, H_1 and H^{MAC}_{key} , to all nodes in P_{SD}. H_1 is unkeyed while HMAC key is a keyed hash function that will be used for message authentication purposes later on. Besides symmetric key distribution, S also needs to set up its HLA keys. Let $e : GxG \rightarrow G_T$ be a computable bilinear map with multiplicative cyclic group G and support Z_p,where p is the prime order of G, i.e., for all $\alpha,\beta \epsilon G$ and q1, $q_2 \epsilon Z_p$, $e(\alpha^{q1},\beta^{q2}) = e(\alpha, \beta)^{q1q2}$.
After completing the setup phase, S enters the packet transmission phase. S transmits packets to PSD. Before sending out a packet P_i, where i is a sequence number that uniquely identifies P_i, S computes $r_i = H_1(P_i)$ and generates the HLA signatures of r_i. node n_j
Audit Phase is triggered when the public auditor Ad receives an ADR message from S. The ADR message includes the id of the nodes on PSD, ordered in the downstream direction, i.e., n1 ... n_K, S's HLA public key information p_k (v, g, u) the sequence numbers of the most recent M packets sent by S, and the sequence numbers of the subset of these M packets that were received by D.
The public auditor Ad enters the detection phase after receiving and auditing the reply to its challenge from all nodes on PSD. The main tasks of Ad in this phase include the following: detecting any overstatement of packet loss at each node, constructing a packet-loss bitmap for each hop, calculating the autocorrelation function for the packet loss on each hop, and deciding whether malicious behavior is present

CONCLUSION

In this paper, we have presented the transmission of packets from source to destination along the path P_{SD}. A malicious node maybe present in this path which may either drop or modify the packets it receives from its upstream node. We showed that compared with conventional detection algorithms that utilize only the distribution of the number of lost packets, exploiting the correlation between lost packets significantly improves the accuracy in detecting malicious packet drops. Such improvement is especially visible when the number of maliciously dropped packets is comparable with those caused by link errors. To correctly calculate the correlation between lost packets, it is critical to acquire truthful packet-loss information at individual nodes. We developed an HLA-based public auditing architecture that ensures truthful packet-loss reporting by individual nodes.

REFERENCES

[1] S. Buchegger and J.-Y. L. Boudec. Performance analysis of the con¯dant protocol (cooperation of nodes: Fairness in dynamic ad-hoc networks). In *MobiHOC 2002*, June 2002.

[2] K. Liu, J. Deng, P. Varshney, and K. Balakrishnan.An acknowledgment-based approach for the detection of routing misbehavior in manets. *IEEE Transactions on Mobile Computing*, 6(5):536{550, May 2006.

[3] Y. Liu and Y. R. Yang. Reputation propagation and agreement in mobile ad-hoc networks. In *WCNC 2003*, pages 1510{1515, March 2003.

[4] S. Marti, T. Giuli, K. Lai, and M. Baker. Mitigating routing misbehavior in mobile ad hoc networks. In *MobiCom 2000*, pages 255{265, 2000.

[5] L. Buttyan and J.-P. Hubaux. Stimulating cooperation in self-organizing mobile ad hoc networks. *ACM/Kluwer Mobile Networks and Applications*, 8(5), 2003.

[6] M. Jakobsson, J.-P. Hubaux, and L. Buttyan. A micropayment scheme encouraging collaboration in multi-hop cellular networks. In *Financial Crypto*, 2003.

[7] S. Zhong, J. Chen, and Y. R. Yang. Sprite: A simple cheat-proof, credit-based system for mobile ad-hoc networks. In *INFOCOM 2003*, pages 1987{1997.

[8] M. Jakobsson, J.-P. Hubaux, and L. Buttyan. A micropayment scheme encouraging collaboration in multi-hop cellular networks. In *Financial Crypto*, 2003.

[9] V. Gligor. Handling new adversaries in secure mobile ad-hoc networks. In *ESNS 2007*, 2007.

[10] S. Buchegger and J.-Y. L. Boudec. Performance analysis of the con¯dant protocol (cooperation of nodes: Fairness in dynamic ad-hoc networks). In *MobiHOC 2002*, June 2002.

[11] S. Buchegger and J.-Y. L. Boudec. Self-policing mobile ad-hoc networks by reputation systems. *IEEE Communications Magazine*, pages 101{107, 2005.

[12] L. Buttyan and J.-P. Hubaux. Stimulating cooperation in self-organizing mobile ad hoc networks. *ACM/Kluwer Mobile Networks and Applications*, 8(5), 2003.

[13] Q. He, D. Wu, and P. Khosla. Sori: A secure and objective reputation-based incentive scheme for ad hoc networks. In *WCNC 2004*, 2004.

[14] S. Marti, T. Giuli, K. Lai, and M. Baker. Mitigating routing misbehavior in mobile ad hoc networks. In *MobiCom 2000*, pages 255 265, 2000.

[15] L. M. Feeney and M. Nilsson. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In *INFOCOM 2001*.

[16] K. Liu, J. Deng, P. Varshney, and K. Balakrishnan. An acknowledgment-based approach for the detection of routing misbehavior in manets. *IEEE Transactionson Mobile Computing*, 6(5):536{550, May 2006.

[17] V.-N. Padmanabhan and D.-R. Simon. Secure traceroute to detect faulty or malicious routing. *SIGCOMM Computer Communication*, 33(1), 2003.

[18] Y. Xue and K. Nahrstedt. Providing fault-tolerant ad-hoc routing service in adversarial environments. *Wireless Personal Communications, Special Issue on Security for Next Generation Communications*, 29(3{4):367{388, 2004}

