Privacy-Preserving and Truthful Detection of Packet Dropping Attacks in Wireless Ad Hoc Networks

M.Krishnamoorthi
Dept of Computer Science
K.S.Ranagasamy College of Arts and Science (Autonomous), Tiruchengode

Abstract- *Link error and Malicious Packet dropping are two sources for Packet Losses in multi-hop wireless Ad hoc network. In this Work while observing a sequence of packet losses in the network, it is interested in determining whether the losses are caused by link errors only, or by the combined effect of link errors and malicious drop. It is especially interested in the insider-attack case, whereby 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. Because the Packet dropping rate in this case is comparable to the channel error rate, Conventional algorithms that are based on detecting the Packet loss rate cannot achieve satisfactory detection accuracy. To improve the detection accuracy, this is proposed to exploit the correlations between lost packets. Furthermore, to ensure truthful calculation of these correlations, it develops a Homomorphic Linear Authenticator (HLA) based public auditing architecture that allows the detector to verify the truthfulness of the packet loss information reported by nodes. This construction is Privacy Preserving, Collusion Proof, and incurs low communication and storage overheads. To reduce the computation overhead of the baseline scheme, a Packet-Block based Mechanism is also proposed, which allows one to trade detection accuracy for lower computation complexity. Through extensive simulations, this is verify that the proposed mechanisms achieve significantly better detection accuracy than conventional methods such as a Maximum-likelihood based detection.*

Keywords- Link Error, Packet Drop, Homomorphic Linear Authenticator, Privacy Preserving, Packet Block

I. INTRODUCTION

An Ad-hoc network is a Local Area Network (LAN) that is built spontaneously as devices connect. Instead of relying on a base station to coordinate the flow of messages to each node in the network, the individual network nodes forward packets to and from each other. In Latin, ad hocliterally means "for this," meaning "for this special purpose" and also, by extension, improvised. In the Windows operating system, ad-hoc is a communication mode (setting) that allows computers to directly communicate with each other without a router.

A wireless Ad hoc Network (WANET) is a decentralized type of wireless network. The network is ad hoc because it does not rely on a pre existing infrastructure, such as routers in Wired Networks or access points in managed (infrastructure) wireless networks. Instead, each node participates in routing by forwarding data for other nodes, so the determination of which nodes forward data is made dynamically on the basis of network connectivity. In addition to the Classic Routing, Ad hoc networks can use flooding for forwarding data.

An ad hoc network typically refers to any set of networks where all devices have equal status on a network and are free to associate with any other ad hoc network device in link range. It also refers to a network device's ability to maintain link status information for any number of devices in a 1-link (aka "hop") range, and thus, this is most often a Layer 2 activity. Because this is only a Layer 2 activity, ad hoc networks alone may not support a routable IP network environment without additional Layer 2 or Layer 3 capabilities. The earliest wireless ad hoc networks were the "Packet Radio Networks" (PRNETs) from the 1970s, sponsored by DARPA after the ALOHA net project.

II. RELATED WORK

The related work can be classified into the following two categories. (i) High Malicious dropping rates (ii) The first category aims at high malicious dropping rates, where most (or all) lost packets are caused by malicious dropping. In this case, the impact of link errors is ignored. Most related work falls into this category. Based on the methodology used to identify the attacking nodes; these works can be further classified into four sub-categories.

i.Credit systems

ii. A credit system provides an incentive for cooperation. A node receives credit by relaying packets for others, and uses its credit to send its own packets. As a result, a maliciously

node that continuous to drop packets will eventually deplete its credit, and will not be able to send its own traffic.

i.Reputation systems

ii. A reputation system relies on neighbors to monitor and identify misbehaving nodes. A node with a high packet dropping rate is given a bad reputation by its neighbors. This reputation information is propagated periodically throughout the network and is used as an important metric in selecting routes. Consequently, a malicious node will be excluded from any route.

iii. Most of the related works assumes that malicious dropping is the only source of packet loss. For the credit-system-based method, a malicious node may still receive enough credits by forwarding most of the packets it receives from upstream nodes. In the reputation-based approach, the malicious node can maintain reasonably good reputation by forwarding most of the packets to the next hop.

III. PROPOSED 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 (*a*1, . . . , *aM*), where *aj 2 f*0, 1*g* 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 vs. 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 verify this property, in Figure below we have simulated the auto-correlation functions of two packet loss processes, one caused by 10% link errors, and the other by 10% link errors plus 10% malicious uniformly-random packet dropping. It can be observed that significant gap exists between these two auto-correlation functions. Therefore, by comparing the auto-correlation function of the observed packet loss process with that of a normal wireless channel (i.e., *fc*(*i*)), one can accurately identify the cause of the packet drops.The benefit of exploiting the correlation of lost packets can be better illustrated by examining the insufficiency of the conventional method that relies only on the distribution of the number of lost packets. More specifically, under the conventional method, malicious-node detection is modeled as a binary hypothesis test, where *H*0 is the hypothesis that there is no malicious node in a given link (all packet losses are due to link errors) and *H*1 denotes there is a malicious node in the given link (packet losses are due to both link errors and malicious drops).

a) Public Verifiability:

After each detection, Ad is required to publish the information it received from involved nodes, i.e., bj, r(j), s(j), for j 2 PSD, so that a node can verify all calculation has been performed correctly. Note that no knowledge of the HLA secret key x is required in the verification process. At the same time, because Ad has no knowledge of x, there is no way for it to forge a valid HLA signature for r (j). In other words, Ad cannot claim a misbehaving node to be a normal one. Furthermore, the privacy-preserving property of the scheme ensures that publishing the auditing information will not compromise the confidentiality of the communication.

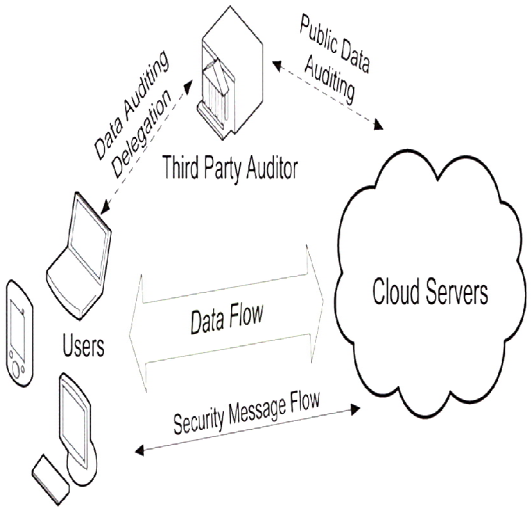

Figure 1.Proposed System Architecture

This 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, . . . , *nK* , *S*'s HLA public key information *pk* = (*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*. Recall that we assume the information sent by *S* and *D* is truthful, because detecting attacks is in their interest. *Ad* conducts the auditing process. Note that the above mechanism only guarantees that a node cannot of a packet that it actually did not receive. This mechanism cannot prevent a node from overly stating its packet loss by claiming that it did not receive a packet that it actually received.

IV. REDUCING COMPUTATION OVERHEAD: BLOCK-BASED HLA SIGNATURE GENERATION AND DETECTION

One major limitation of the proposed baseline HLA detection algorithm is the high computation overhead of the source node. In this section, we proposed a block-based solution that can reduce this overhead by multiple folds. The main idea is to make the HLA signature scalable: instead of generating per-packet HLA signatures, per-block HLA signatures will be generated, where a block consists of $L > 1$ packets. Accordingly, the detection will be extended to blocks, and each bit in the packet-loss bitmap represents a block of packets rather than a single packet. The details of this extension are elaborated as follows. In the Packet Transmission Phase, rather than generating HLA signatures for every packet, now the signatures are based on a block of packets. In particular, L consecutive packets are deemed as one block.

a) Scheme Details

(i) Setup phase

This phase takes place right after route PSD 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 key1. . . 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 to node nj on PSD, for j = 1, . . . ,K. Key distribution may be based on the public-key crypto-system such as RSA: S encrypts key using the public key of node nj and sends the cipher text to nj . nj decrypts the cipher text using its private key to obtain key . S also announces two hash functions, H1 and HMAC key, to all nodes in PSD. H1 is un keyed while HMAC key is a keyed hash function.

(ii) Audit Phase

This 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, . . . , nK , S's HLA public key information $pk = (v, g, u)$, 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. Recall that we assume the information sent by S and D is truthful, because detecting attacks is in their interest. *Ad* conducts the auditing process. Note that the above mechanism only guarantees that a node cannot of a packet that it actually did not receive. This mechanism cannot prevent a node from overly stating its packet loss by claiming that it did not receive a packet that it actually received. This latter case is prevented by another mechanism discussed in the detection phase.

(iii) Detection Phase

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. More specifically, Ad performs these tasks as follows. The auditor calculates the autocorrelation function.

The detection process applies to one end-to-end path. The detection for multiple paths can be performed as multiple independent detections, one for each path. Although the optimal error threshold that minimizes the detection error is still an open problem, our simulations show that through trial-and-error, one can easily find a good ϵth that provides a better detection accuracy than the optimal detection scheme that utilizes only the pdf of the number of lost packets.

V.PERFORMANCE ANALYSIS AND RESULTS

a) Simulation Setup

The detection accuracy which can be achieved by the Conventional algorithm with the optimal maximum likelihood algorithm that utilizes the distribution of the number of lost packets. For given packet-loss bitmaps, the detection on different hops is conducted separately. So, only need to simulate the detection of one hop to evaluate the performance of a given algorithm. It assume packets are transmitted continuously over this hop, i.e., a saturated traffic environment and assume channel fluctuations for this hop follow the Gilbert-Elliot model, with the transition probabilities from good to bad and from bad to good given respectively. The two types of malicious packet dropping: random dropping and selective dropping. In the random dropping attack, a packet is dropped at the malicious node with probability. In the selective dropping attack, the adversary drops packets of certain sequence numbers.

(i) Packet Drop

The detection error as a function of the number of maliciously dropped packets. Similar performance trends can be observed to the case of the random packet dropping. Fewer

detection errors are made by both algorithms when more packets are maliciously dropped. In all the simulated cases, the proposed algorithm can detect the actual cause of the packet drop more accurately than the ML scheme, especially when the number of maliciously dropped packets is small. When the number of maliciously dropped packets is significantly higher than that caused by link errors (greater than 4 packets in our simulation), the two algorithms achieve comparable detection accuracy. In this scenario, it may be wise to use the conventional ML scheme due to its simplicity (e.g., no need to enforce truthful reports from intermediate nodes, etc).

Packet lost = Number of packet send – Number of packet received.

The lower value of the packet lost means the better performance of the protocol.

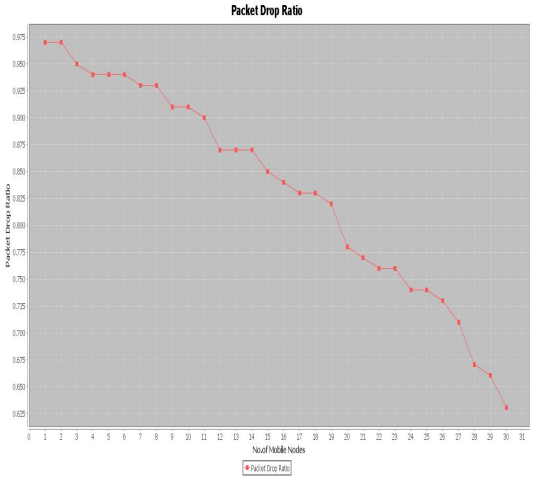

Figure 2 .Packet Drop Ratio

(ii) Throughput

The Average rate of successful Packet Delivery over a communication channel called Throughput. The Throughput is usually measured in bit/s or data packets/sec. It is the ratio of the total amount of data that reaches a receiver from a sender to the time it takes for the receiver to get the last packet.

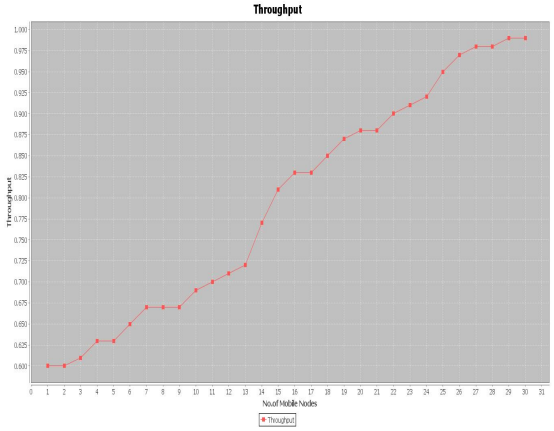

Figure 3. Throughput

(iii) Packet Delivery Ratio

Packet Delivery Ratio (PDR) is the ratio between the number of packets transmitted by a traffic source and the number of packets received by a traffic sink. It measures the loss rate as seen by transport protocols and as such, it characterizes both the correctness and efficiency of ad hoc routing protocols. A high is desired in any network. The ratio of the Originated applications' packets of each protocol which was able to deliver at varying time.

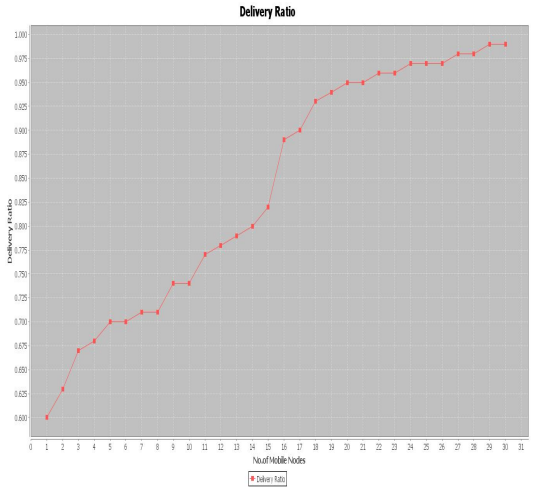

Figure 4.Packet Delivery Ratio

(iv) False Packet Ratio

The false packet in measurement of how much of packets dropped and except remaining packets are false packets.

False packet = total no of packets sent– total no dropped packets = no of remain Packets

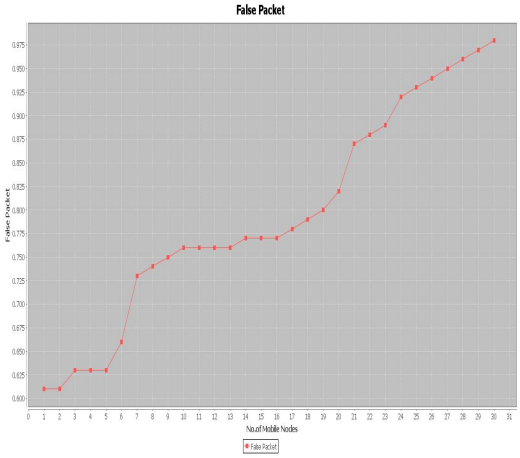

Figure 5. False Packet Ratio

VI. CONCLUSION

Therefore, by detecting the correlations between lost packets, one can decide whether the packet loss is purely due to regular link errors, or is a combined effect of link error and malicious drop. The algorithm takes into account the cross statistics between lost packets to make a more informative decision, and thus is in sharp contrast to the conventional methods that rely only on the distribution of the number of lost packets. It is 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.

REFERENCES

[1] Aishwarya Sagar Anand Ukey, Meenu Chawla Department Of Computer Science Engineering, Maulana Azad National Institute of Technology," Detection of Packet Dropping Attack Using Improved Acknowledgement Based Scheme in MANET. They Published Their Research Paper in "International Journal of Computer Science Issues" On July 2012.

[2] Sagar Patolia and Harmandeep Singh, Department of Computer Engineering, Lovely Professional University, Phagwara, Punjab, India" Review of Isolate and Prevent Selective Packet Drop Attack in MANET" They Published Their Research Paper International Journal of Innovative Research in Science" On December 2014.

[3] Priya Malhotra Student of Master of Technology Department Of Computer Science and Engineering Shobhit University, Meerut, Uttar Pradesh, India" Detecting Packet-Dropping Faults in Mobile Ad-Hoc Wireless Networks. They Published Their Research Paper in "International Journal OF Advanced Research In Computer Science And Software Engineering" On February 2015.

[4] Rajendra Aaseri, Pankaj Choudhary, Nirmal Roberts Abv-indian institute of information technology, gwalior, India." trust value algorithm: a secure approach against packet drop attack in wireless ad-hoc networks" The Published Their Research Paper in International Journal of Security and Its Applications" On May 2013.

[5] P.R. Jasmine Jeni, A. Vimala Julie and A. Messiah Bose SRM University, Chennai, Tamilnadu, India" an enhanced route failure recovery model for mobile ad hoc networks" they published their research work in "Journal Of Computer Science" On 2014.

[6] Meenakshi Devi, Dr. Pardeep Kumar Mittal Assistant Professor DCSA, KU, Haryana, India DCSA," Local Route Repair in MANET for on Demand Routing Protocol: A Review" They Published Their Research Work in International Journal Of Advanced Research In Computer Science And Software Engineering" On May 2014

[7] Xin Ming Zhang, Member, IEEE, En Bo Wang, Jing Jing Xia, and Dan Keun Sung, Senior Member, IEEE, A Neighbor Coverage-Based Probabilistic Rebroadcast for Reducing Routing Overhead in Mobile Ad Hoc Networks, IEEE transactions on mobile computing, vol. 12, no. 3, march 2013.

[8] Yoav Sasson David Cavin Andre Schiper ,Probabilistic Broadcast for Flooding in Wireless Mobile Ad hoc Networks, Swiss Federal Institute of Technology (EPFL)1015 Lausanne, Switzerland

[9] M. A. Gafur, N. Upadhayaya, S. A. Sattar, " An Efficient Approach For Local Repairing In Mobile Ad hoc Network", Canadian Journal on Network and Information Security Vol. 3 No. 1, August 2012..

[10] V. P. Patil et al. "Performance Enhancement of Reactive on Demand Routing Protocol in Wireless Ad Hoc Network", International Journal of Smart Sensors and Ad Hoc Networks (IJSSAN) ISSN No. 2248-9738 Volume-1, Issue-4, 2012.

[11] K. S. Rao, L. Shrivastava, "Efficient Local Route Repair Method in AODV to Reduce Congestion in MANET", Corona Journal of Science and Technology, Vol. 1, No. 1, October 2012.

[12] A. Akbari, M. Soruri and A. Khosrozadeh, " A New AODV Routing Protocol in Mobile Adhoc Networks" , World Applied Sciences Journal 19 (4): 478-485, 2012.

[13] P. Priya Naidu and M. Chawla ,"Extended Ad hoc On Demand Distance Vector Local Repair For MANET" International Journal of Wireless and Mobile Networks(IJWMN), vol. 4, No. 2, April 2012.