International Journal of Engineering Research in Computer Science and Engineering (IJERCSE) Vol 3, Issue 3, March 2016

Identification of Truthful Packet Dropping In Wireless Ad-Hoc Networks Using HLA Algorithm

[1] CH V Divya Teja [2] V Dinesh, [3] Prasad B
[1]II/IV, [2][3]Associate Professor
[1][2][3] Department of CSE, Marri Laxman Reddy Institute of Technology and Management (MLRITM) Hyderabad
[1] vdteja.chebrolu@gmail.com [2] dinesh.valluru15@yahoo.in [3]bprasad@gmail.com

Abstract:-Identification of truthful packet dropping in wireless ad-hoc networks using HLA algorithm's main purpose is to determine whether the is loss of packets are caused by link errors only or by the combined effect of link errors and malicious drop errors that are the two causes in multi-hop wireless ad-hoc networks. It is particularly considered for insider attack case, where by malicious nodes which are part of the route exploit cognition of communication context to selectively drop a small amount of packets that act critical to the network performance. Development of homo_morphic linear authenticator [HLA] is necessary to ensure truthful calculation of these correlations and to verify truthfulness of the packet loss information reported by nodes. This construction is privacy preserving, collusion proof and provokes low communication and storage overheads. To reduce computation overhead a packet block based mechanism is also proposed to trade detection accuracy for low computation complexity.

Keywords: **Packet Dropping, Wireless Ad-Hoc Networks, HLA Algorithm's, Link Errors And Malicious Drop Errors.**

I. INTRODUCTION

IN a multi-hop wireless network, nodes cooperate in relaying/routing traffic. An adversary can exploit this cooperative nature to launch attacks. For example, the adversary may first pretend to be a cooperative node in the route discovery process. Once being included in a route, the adver-sary starts dropping packets. In the most severe form, the malicious node simply stops forwarding every packet received from upstream nodes, completely disrupting the path between the source and the destination. Eventually, such a severe denial-of-service (DoS) attack can paralyze the network by partitioning its topology. In this paper, we develop an accurate algorithm for detecting selective packet drops made by insider attackers. Our algorithm also provides a truthful and publicly verifiable decision statistics as a proof to support the detection decision. The high detection accuracy is achieved by exploiting the correlations between the positions of lost packets, as calculated from the auto-correlation function (ACF) of the packet-loss bitmap—a bitmap describing the lost/received status of each packet in a sequence of consecutive packet transmissions. The basic idea behind this method is that even though malicious dropping may result in a packet loss rate that is comparable to normal channel losses, the stochastic processes that characterize the two phenomena exhibit different correlation structures (equivalently, different patterns of packet losses). Therefore, by detecting the correla-tions 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. Our 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. Our solution to the above public-auditing problem is constructed based on the homomorphic linear authenticator (HLA) cryptographic primitive which is basically a signature scheme widely used in cloud computing and storage server systems to provide a proof of storage from the server to entrusting clients [30]. However, direct application of HLA does not solve our problem well, mainly because in our problem setup, there can be more than one malicious node along the route. These nodes may collude (by exchanging information) during the attack and when being asked to submit their reports. For example, a packet and its associated HLA signature may be dropped at an upstream malicious node, so a downstream malicious node does not receive this packet and the HLA signature from the route. However, this downstream attacker can still open a back-channel to request this information from the upstream malicious node. When being audited, the downstream malicious node can still provide valid

proof for the reception of the packet. So packet dropping at the upstream malicious node is not detected. Such collusion is unique to our problem, because in the cloud computing/storage server scenario, a file is uniquely stored at a single server, so there are no other parties for the server to collude with. We show that our new HLA construction is collusion-proof. Our construction also provides the following new features. First, privacy-preserving: the public auditor should not be able to decern the content of a packet delivered on the route through the auditing information submitted by individual hops, no matter how many independent reports of the auditing information are submitted to the auditor. Second, our construction incurs low communication and storage overheads at intermediate nodes. This makes our mechanism applicable to a wide range of wireless devices, including low-cost wireless sensors that have very limited bandwidth and memory capacities. This is also in sharp contrast to the typical storage-server scenario, where bandwidth/storage is not considered an issue. Last, to significantly reduce the computation over-head of the baseline constructions so that they can be used in computation-constrained mobile devices, a packet-block-based algorithm is proposed to achieves scalable signature generation and detection. This mechanism allows one to trade detection accuracy for lower computation complexity.

II. RELATED WORK

Depending on how much weight a detection algorithm gives to link errors relative to malicious packet drops, the related work can be classified into the following two categories. 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. The first sub-category is based on credit systems . 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. The second sub-category is based on reputation systems . 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. The third sub-category of works relies on end-to-end or hop-to-hop acknowledgements to directly locate the hops where packets are lost . A hop of high packet loss rate will be excluded from the route. The fourth subcategory addresses the problem using cryptographic methods. For example, the work in utilizes Bloom filters to construct proofs for the forwarding of packets at each node. By examining the relayed packets at successive hops along a route, one can identify suspicious hops that exhibit high packet loss rates. Similarly, the method in traces the forwarding records of a particular packet at each intermediate node by formulating the tracing problem as a Renyi-Ulam game. The first hop where the packet is no longer forwarded is considered a suspect for misbehaving.

III. SYSTEM ANALYSIS

Existing System:

The related work can be classified into the following two categories. High malicious dropping rates. 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 subcategories. Credit systems 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. Reputation systems 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. o End-to end or hop-to-hop acknowledgements To directly locate the hops where packets are lost. A hop of high packet loss rate will be excluded from the route. Cryptographic methods Bloom filters used to construct proofs for the forwarding of packets at each node. By examining the relayed packets at successive hops along a route, one can identify suspicious hops that exhibit high packet loss rates. Number of maliciously dropped packets is significantly higher than that caused by link errors. The second category targets the scenario where the number of maliciously dropped packets is significantly higher than that caused by link errors, but the impact of link errors is non-negligible.

Limitation of Existing System:

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 a reasonably good reputation by forwarding most of the packets to the next hop. While the Bloom-filter scheme is able to provide a packet forwarding proof, the correctness of the proof is probabilistic and it may contain errors. As for the acknowledgement-based method and all the mechanisms in the second category, merely counting the number of lost packets does not give a sufficient ground to detect the real culprit that is causing packet losses.

Proposed System:

To develop an accurate algorithm for detecting selective packet drops made by insider attackers. This algorithm also provides a truthful and publicly verifiable decision statistics as a proof to support the detection decision. The high detection accuracy is achieved by exploiting the correlations between the positions of lost packets, as calculated from the auto-correlation function (ACF) of the packet-loss bitmap–a bitmap describing the lost/received status of each packet in a sequence of consecutive packet transmissions. 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 main challenge in our mechanism lies in how to guarantee that the packet-loss bitmaps reported by individual nodes along the route are truthful, i.e., reflect the actual status of each packet transmission. Such truthfulness is essential for correct calculation of the correlation between lost packets, this can be achieved by some auditing. Considering that a typical wireless device is resource-constrained, we also require that a user should be able to delegate the burden of auditing and detection to some public server to save its own resources. Public-auditing problem is constructed based on the homomorphic linear authenticator (HLA) cryptographic primitive, which is basically a signature scheme widely used in cloud computing and storage server systems to provide a proof of storage from the server to entrusting clients.

Advantages:

High detection accuracy Privacy-preserving: the public auditor should not be able to decern the content of a packet delivered on the route through the auditing information submitted by individual hops Incurs low communication and storage overheads at intermediate nodes.

Challenges:

The main challenge in our mechanism lies in how to guarantee that the packet-loss bitmaps reported by individual nodes along the route are truthful, i.e., reflect the actual status of each packet transmission. Such truthfulness is essential for correct calculation of the correlation between lost packets. This challenge is not trivial, because it is natural for an attacker to report false information to the detection algorithm to avoid being detected. For example, the malicious node may understate its packet-loss bitmap, i.e., some packets may have been dropped by the node but the node reports that these packets have been forwarded. Therefore, some auditing mechanism is needed to verify the truthfulness of the reported information. Considering that a typical wireless device is resource-constrained, we also require that a user should be able to delegate the burden of auditing and detection to some public server to save its own resources.

IV. SYSTEM DESIGN

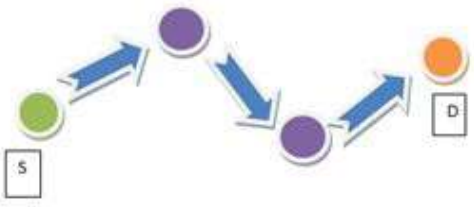

Fig1: Intermediate nodes with source and destination.

System Modules:

The system contains four modules. 1. Network modeling. 2. Independent auditing. 3. Setup phase. 4. Packet dropping detection

A. Network modeling

The wireless channel is modeled of each hop along PSD (Path to Source and Destination) as a random process that alternates between good and bad states. Packets transmitted during the good state are successful, and packets transmitted during the bad state are lost. It is assumed quasi-static networks, whereby the path PSD remains unchanged for a relatively long time. Detecting malicious packet drops may not be a concern for highly mobile networks, because the fast-changing topology of such networks makes route disruption the dominant cause for packet losses. In this case, maintaining

stable connectivity between nodes is a greater concern than detecting malicious nodes. A sequence of M packets is transmitted consecutively over the channel.

B. Independent auditor

There is an independent auditor Ad in the network. Ad is independent in the sense that it is not associated with any node Proceedings . The auditor is responsible for detecting malicious nodes on demand. Specifically, it is assumed S receives feedback from D when D suspects that the route is under attack. Once the destination click on verify, the action takes places to identify the packet loss. To facilitate its investigation, Ad needs to collect certain information from the nodes on route PSD.

C. 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 encrypt the packets and sent through the route to destination. Destination after receiving packets can verify the packet and after verification it can decrypt the packets.

D. Packet drop detection

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. Under different packet dropping conditions, packet loss is identified.

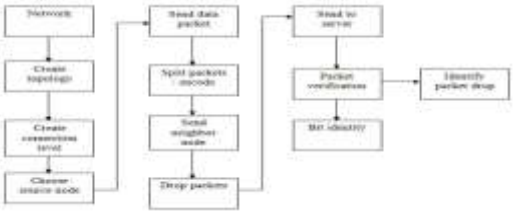

Fig 2: System architecture

V. IMPLEMENTATION

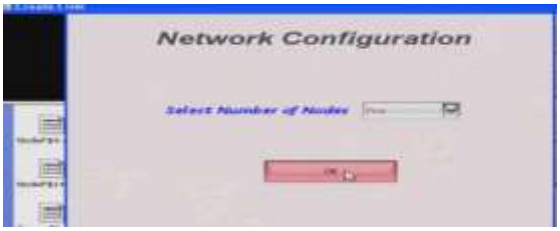

Fig 3: Selection of Nodes

Fig 4: Input

Fig 5 : Node Configuration

As shown ,fig 5 screen shots represents the configuration of nodes by giving required input.

Fig 6: Server

As shown in fig 6 Server node represents the central node for providing required packets i.e; through which data can be transferred.

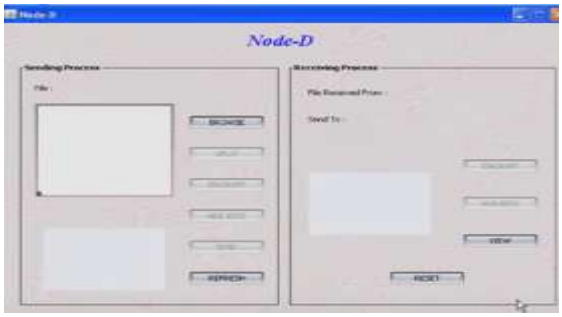

Fig 7: Node representation

Fig 8: Node representing selected packets

Fig 9: Received Packets in Node A

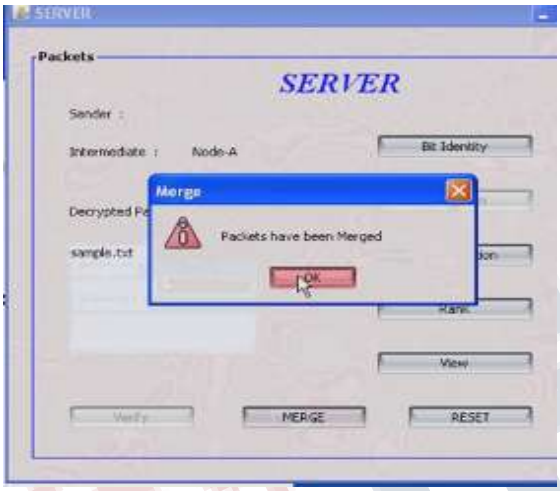

Fig 10 : Merging of packets in servers

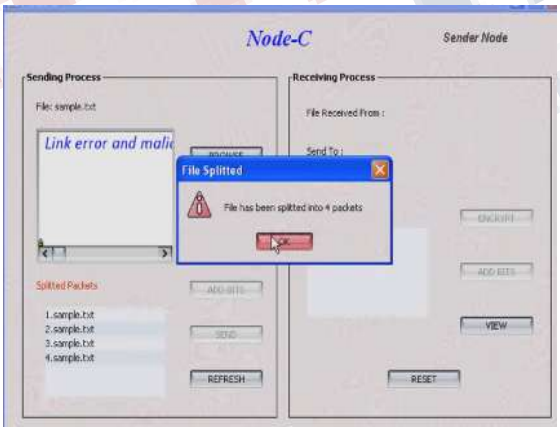

Fig : 11 Splitting of packets

Each packet is splitted into four sub packets as shown in fig 11.

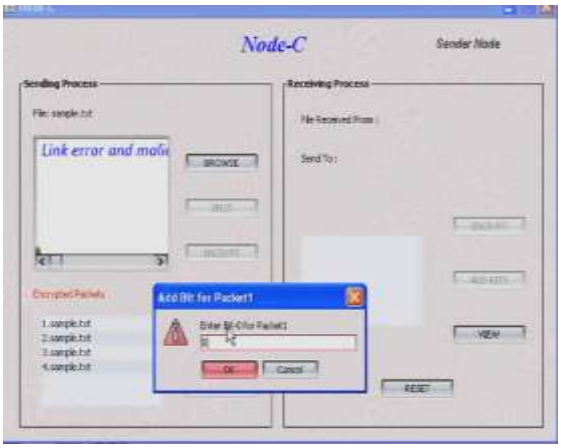

Fig 12 : Adding bit function to spiltted packets

Bits are added to the splitted packets by using bit functions as shown in fig 12.

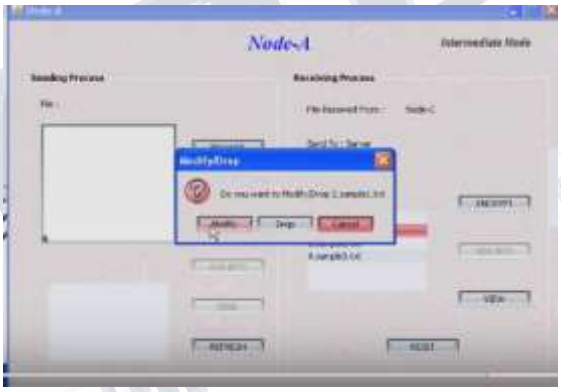

Fig 13 : Modification of packets

Fig14 : Packet drop

Packet drop is dropped due to malicious attack in nodes as shown in fig 14.

Fig 15 : Decryption of packet received

The packet received is decrypted and data which is sent by sender is received.

VI. RESULT ANALYSIS

The system is implemented. The initially the network is configured with calling the Nodeconfigure function with number of nodes. And then Linkcreate will create link, while creating link we need to specify the levels with which the node is associated. Once the network is configured we take up server as the destination and any of the nodes as the sender. Once the network is set we browse for the file we need to send. In the source we split the entire file in to number of packets these packets will be encrypted and Addbit function will help in adding bits to identify the change in number of packets and packet will be forwarded further. The packet will be received by the intermediated node in normal transition packet will be encrypted and forwarded whereas in attacker mode packet will be dropped or modified or both will be done and forwarded.Once the packet reach destination in normal node packet will be verified, bit identified, decrypted and finally merged. In attacker mode when packet is verified the packet dropped is identified, bit identification will let us know about packet modification. On modification or dropped packet cannote decrypted.We also have option for categorization which gives the number of packet received properly and number of packet modified. Also provide ranking about the node which help in routing in further packet transfer. The expected results are if any packets are modified or dropped our system has to identify the attack and the node in which the attack has occurred. This has been achieved thus gives us the proper results for the system implemented. In the system implement we can verify the packet received and rate about the nodes participated in transmission of the packets to destination.

VII. CONCLUSION

In this paper, 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. This architecture is collusion proof, requires relatively high computational capacity at the source node, but incurs low communication and storage overheads over the route. To reduce the computation overhead of the baseline construction, a packet-block-based mechanism was also proposed, which allows one to trade detection accuracy for lower computation complexity. Some open issues remain to be explored in our future work. First, the proposed mechanisms are limited to static or quasi-static wireless ad hoc networks. Frequent changes on topology and link characteristics have not been considered. Extension to highly mobile environment will be studied in our future work. In addition, in this paper we have assumed that source and destination are truthful in following the established protocol because delivering packets end-to-end is in their interest. Misbehaving source and destination will be pursued in our future research. Moreover, in this paper, as a proof of concept, we mainly focused on showing the feasibility of the proposed cypto-primitives and how secondorder statistics of packet loss can be utilized to improve detection accuracy. As a first step in this direction, our analysis mainly emphasize the fundamental features of the problem, such as the untruthfulness nature of the attackers, the public verifiability of proofs, the privacy-preserving requirement for the auditing process, and the randomness of wireless channels and packet losses, but ignore the particular behavior of various protocols that may be used at different layers of the protocol stack. The implementation and optimization of the proposed mechanism under various particular protocols will be considered in our future studies.

REFERENCES

[1] J. N. Arauz, “802.11 Markov channel modeling,” Ph.D. dissertation, School Inform. Sci., Univ. Pittsburgh, Pittsburgh, PA, USA, 2004.

[2] C. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in Proc. ACM Conf. Comput. and Commun. Secur., Oct. 2007, pp. 598–610.

[3] G. Ateniese, S. Kamara, and J. Katz, "Proofs of storage from homomorphic identification protocols," in Proc. Int. Conf. Theory Appl. Cryptol. Inf. Security, 2009, pp. 319–333.

[4] B. Awerbuch, R. Curtmola, D. Holmer, C. Nita-Rotaru, and H. Rubens, "ODSBR: An on-demand secure byzantine resilient routing protocol for wireless ad hoc networks," ACM Trans. Inform. Syst. Security, vol. 10, no. 4, pp. 1–35, 2008.

[5] B. Awerbuch, R. Curtmola, D. Holmer, C. Nita-Rotaru, and H. Rubens, "ODSBR: An on-demand secure byzantine resilient routing protocol for wireless ad hoc networks," ACM Trans. Inf. Syst. Secur., vol. 10, no. 4, pp. 11–35, 2008.

[6] K. Balakrishnan, J. Deng, and P. K. Varshney, "TWOACK: Preventing selfishness in mobile ad hoc networks," in Proc. IEEE Wireless Commun. Netw. Conf., 2005, pp. 2137–2142.

[7] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," J. Cryptol., vol. 17, no. 4, pp. 297–319, Sep. 2004.

[8] S. Buchegger and J. Y. L. Boudec, "Performance analysis of the confidant protocol (cooperation of nodes: Fairness in dynamic adhoc networks)," in Proc. 3rd ACM Int. Symp. Mobile Ad Hoc Netw. Comput. Conf., 2002, pp. 226–236.

[9] L. Buttyan and J. P. Hubaux, "Stimulating cooperation in selforganizing mobile ad hoc networks," ACM/Kluwer Mobile Netw. Appl., vol. 8, no. 5, pp. 579–592, Oct. 2003.

[10] J. Crowcroft, R. Gibbens, F. Kelly, and S. Ostring, "Modelling incentives for collaboration in mobile ad hoc networks," presented at the First Workshop Modeling Optimization Mobile, Ad Hoc Wireless Netw., Sophia Antipolis, France, 2003.