International Journal of Research
Available at https://edupediapublications.org/journals

p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 10
June 2016

Identifying Malicious Packet Dropping By Using Holomorphic Identification Protocols in Wireless Ad Hoc Networks

[1]K. Madhusudhan & [2]K. Muralidhar

[1]M.Tech Student, Dept. of CSE, ALITS Engineering College, Affiliated to JNTUA , Andhra Pradesh, India

[2]Assistant Professor in Dept. of CSE, , ALITS Engineering College, Affiliated to JNTUA, Andhra Pradesh, India

Abstract---Wireless ad hoc network is a network formed without any central infrastructure which consists of nodes that use a wireless interface to send packet data. Linkage error and malicious packet dropping are two sources for packet losses in wireless ad hoc network. A sequence of packet losses are present in the network, it determines whether the losses are caused by linkage errors only, or by the joint effect of linkage errors and malicious drop. In the interior-attack case, whereby malicious nodes that are part of the route utilize their knowledge of the communication framework to selectively drop a small amount of packets vital to the network performance. This is because the packet dropping rate is comparable to the channel error rate. Conventional algorithms are used to detect the packet loss rate that cannot reach acceptable detection accuracy. We proposed to improve the detection accuracy. So we developed the correlations between lost packets and to ensure truthful calculation of these correlations, the homomorphic linear authenticator (HLA) is used. HLA is based on public auditing architecture that allows the detector to verify the truthfulness of the packet loss information reported by nodes. This development is privacy protect, scam proof, and low communication and storage overheads. It reduce the computation overhead, a packet-block based method is also proposed, which allows one to trade detection truthfulness for lower computation complexity. The proposed mechanisms obtain much better detection accuracy than conventional methods.

Index Terms—*Wireless Adhoc Network; Public Auditing; Selective Dropping; Homomorphic Linear Authenticator*

I.INTRODUCTION

Wireless ad hoc networks are collections of wireless nodes, that communicate directly over common wireless channel. The nodes are equipped with wireless transceiver. They don't need any additional infrastructure, such as base station or wired access point, etc. Therefore, each node doesn't only plays the role of an end system, but also acts as a router, that sends packets to desired nodes. The ad hoc are expected to do assignments, which the infrastructure can't do. Ad hoc networks are mostly used by military, rescue mission team, taxi driver. Their works can't rely on a infrastructure's network. As an illustrative example, imagine fire fighters put out hazardous fire in a big forest. They have to communicate each other, but establishing a infrastructure or cabling in such area is impossible or too expensive. The main problems in ad hoc networks are routing and characteristic of wireless communication. In infrastructure's networks a node can communicate with all nodes in the same cell. In ad hoc a node can communicate only with nodes in its area, this node can communicate with other nodes, but a routing algorithm is necessary. Unlike wired communication, wireless networks have transmission problem with data transmission such as, possibility of asymmetric connections and higher interferences. The aim of this overview article is to provide informations on ad hoc networks and specially WANET, their structure, their applications on the current time, as well as their strong and weakness in comparison with infrastructure networks.

In the case of computer networks, the ad hoc networks mean wireless network without infrastructure, they can be called spontaneous network. One way to understand ad hoc networks is by comparing them with infrastructure based wireless networks, such as cellular network and WLAN. In the infrastructure based wireless networks a node can only send a packet to a destination node only via access point (in cellular network like GSM, it is called base station). The access point establishes an network area and only the nodes in this area can use access point's services. There are some unknown events, which cause access point's malfunction. The nodes lose their network and they are quasi not working. It is the biggest infrastructure's disadvantage. There are also some reasons to sacrifice or not to use access point's services. These can be cost factor, impossibility to install access point in short time, etc. In this case the nodes have to build its own network. This network is called wireless adhoc network. The wireless ad hoc networks only consist of nodes equipped with transceiver. The network are created to be independent from an infrastructure. Therefore, the nodes must be able to arrange their own networks. A node can now communicate only with other nodes in its transmission range. In the infrastructure based wireless network, the nodes can communicate with a node, which is located in another network area, by transmitting data to destination access point and this access point relay the data to the desired node. It seems like, that the ad hoc networks are not powerful enough. Each node has its own transmission range, if these small transmission areas are combined, they will form a much bigger transmission area. The nodes transmit their data with single or multiple hopping technique. Now a suitable routing algorithm must be implemented, so the process of transmitting data will be more effective.

The wireless networks can be categorized based on their system architecture into two basic versions. The one is Infrastructure and second is ad-hoc network. The biggest difference in them is infrastructure networks consist of access point and nodes, meanwhile the ad hoc networks are independent from access point. In the infrastructure version, a terminal can't communicate directly with other terminals in the same cell and other cell. A access point here perform control messages. Messages are sent to the access point and then the access point distributes the messages to the desired terminal. If a terminal want to communicate with a terminal, which is located in other cell, the access point will relay the message to other access point, which has control over desired

cell. The access points are normally wired connected. The problem in infrastructure, if the access point fails, all terminal in this cell can't perform any communication. Unlike the infrastructure, the ad hoc networks have a different method to distribute messages. In a given network, N1 want to communicate with N5. N5 is located outside N1 transmission range, so N1 must hop the message to N4-N2-N3-N5 or N2-N3-N5. Routing algorithm will decide, which route performs the best. There will be no problem if N4 leaves the network, because N1 still has a route to N5. Therefore ad hoc networks are robuster than infrastructure.

A. Public auditing

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. At last, to significantly reduce the computation overhead of the baseline constructions so that they can be used in computation-constrained mobile devices, a conventional-based algorithm to achieves scalable signature generation and detection. This mechanism allows one to trade detection accuracy for lower computation complexity. A malicious node that is occurred on the route can exploit its information of the network protocol and the communication context to launch an *insider attack*–an attack that is intermittent, but can achieve the network performance degradation. Detecting selective packet-dropping attacks is extremely challenging in a highly dynamic wireless environment. The difficulty comes from the requirement that we need to not only detect the place (or hop) where the packet is dropped, but also identify whether the drop is intentional or unintentional. (e.g., fading, noise, and interference, a.k.a., link errors), or by the insider attacker. The above problem has not been well addressed in the literature. The most of the related works preclude the ambiguity of the environment by assuming that malicious dropping is the only source of packet loss, so that there is no need to account for the impact of link errors. On the other hand, for the small number of works that differentiate between link errors and malicious packet drops, their detection algorithms usually require the number of maliciously-dropped packets to be significantly higher than link errors, in order to achieve an acceptable detection accuracy.

III. DESCRIPTION OF THE SYSTEM

A. System Mode

Consider an arbitrary path PSD in a multi-hop wireless ad hoc network, as shown in Figure 1. The source node S continuously sends packets to the destination node D through intermediate nodes $n1, . . . , nK$, where ni is the upstream node of $ni+1$, for 1 i K 1. We assume that S is aware of the route PSD, as in Dynamic Source Routing (DSR) .If DSR is not used, S can identify the nodes in PSD by performing a trace route operation. Here we mainly focus on static or quasi-static wireless ad hoc networks, i.e., we assume that the network topology and link characteristics remain unchanged for a relatively long period of time Example networks include wireless mesh networks (WMNs) and ad hoc networks formed in nomadic computing. Extension to a highly mobile environment is out of our scope and will be considered in the future work We model the wireless channel of each hop along PSD 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. In contrast to the classical Gilbert-Ellioit (GE) channel model, here we do not assume any Markovian property on the channel behavior. We only require that the sequence of sojourn times for each state follows a stationary distribution, and the autocorrelation function of the channel state, say $fc(i)$, where i is the time lag in packets, is also stationary. Here we limit our study to quasi-static networks, whereby the path PSD remains unchanged for a relatively long time, so that the link error statistics of the wireless channel is a wide-sense stationary (WSS) random process (i.e., $fc(i)$ is stationary). 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. In brief, a sequence of M packets are transmitted consecutively over the channel. By observing whether the transmissions are successful or not, the receiver obtains a realization of the channel state ($a1, . . . , aM$), where aj 2 $f0, 1g$ for $j = 1, . . . , M$. In this sequence, “1” denotes the packet was successfully received, and “0” denotes the packet was dropped. $fc(i)$ is derived by computing the autocorrelation function of this def sample sequence: $fc(i) = Efajaj+i g$ for $i = 0, . . . , M$, where the expectation is calculated over all transmitted packets $j = 1, . . . , M$.

This autocorrelation function describes the correlation between packet transmissions (successful/lost) at different times, as a function of the time lag. The time invariant nature of fc is guaranteed by the WSS assumption of the wireless channel. The measurement of $fc(i)$ can take place online or offline. A detailed discussion on how $fc(i)$ is derived is out of the scope of this paper, and we simply assume that this information is given as input to our detection algorithm.

There is an independent auditor Ad in the network. Ad is independent in the sense that it is not associated with any node in PSD and does not have any knowledge of the secrets (e.g., cryptographic keys) held by various nodes. The auditor is responsible for detecting malicious nodes on demand. Specifically, we assume S receives feedback from D when D suspects that the route is under attack. Such a suspicion may be triggered by observing any abnormal events, e.g., a significant performance drop, the loss of multiple packets of a certain type, etc. We assume that the integrity and authenticity of the feedback from D to S can be verified by S using resource-efficient cryptographic methods such as the Elliptic Curve Digital Signature Algorithm (ECDSA). Once being notified of possible attacks, S submits an *attack-detection request* (ADR) to Ad. To facilitate its investigation, *Ad* needs to collect certain information (elaborated on in the next section) from the nodes on route *PSD*. We assume that each such node must reply to *Ad*'s inquiry, otherwise the node will be considered as misbehaving. We assume that normal nodes will reply with truthful information, but malicious nodes may cheat. At the same time, for privacy reasons, we require that *Ad* cannot determine the content of the normal packets delivered over *PSD* from the information collected during the auditing.

B. 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 ($a1, \ldots, aM$), where *aj* 2 *f*0, 1*g* for packets $j = 1, \ldots, M$. 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 2 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). Let *z* be the observed number of lost packets on the link during some interval *t*. Then,

$$z = \begin{cases} x, \text{ under } H0 & \text{(no malicious nodes)} \\ x + y, \text{ under } H1 & \text{(there is a malicious node)} \end{cases}$$

where *x* and *y* are the numbers of lost packets caused by link errors and by malicious drops, respectively. Both *x* and *y* are random variables.

C. 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., $n1, \ldots, nK$, *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 understate its packet loss, i.e., it cannot claim the reception 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.

D. 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.

IV.PERFORMANCE EVALUATION

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 Selective Packet Dropping.
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).

Dropping of Control Packets The simulations so far have not made any application- semantic (use case) assumption on the dropped packets. In reality, however, because these packets are usually used for control purposes, the loss of these packets may generate significant impacts on the transmission of other (i.e., data) packets. In this series of simulations, to evaluate how the correlation between the control and data packets affects the performance of the proposed scheme. In particular, consider a multi-hop cognitive radio network, where control packets are exchanged over an end-to-end path to maintain channel synchronization between consecutive hops.

Block-Based Detection In this series of simulations, the detection accuracy of block-based algorithms as a function of block size. In general, it shows that for both cases the detection error increases with the block size. This is expected, as a larger block size hides more details of packet losses, and therefore makes the actual correlation of lost packets more difficult to calculate.

Meanwhile, the benefits of blocked-based algorithm is also observed. It is able to trade computation complexity for better detection accuracy.

CONCLUSION

In this Research 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.

REFERENCES

[1] K. Al Agha, M.-H. Bertin, T. Dang, A. Guitton, P. Minet, T. Val, and J.-B. Viollet, “Which wireless technology for industrial wireless sensor networks? The development of OCARI technol,” *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 4266–4278, Oct. 2009.

[2] R. Akbani, T. Korkmaz, and G. V. S. Raju, “Mobile Ad hoc Net-work Security,” in *Lecture Notes in Electrical Engineering*, vol. 127. New York: Springer-Verlag, 2012, pp. 659–666.

[3] R. H. Akbani, S. Patel, and D. C. Jinwala, “DoS attacks in mobile ad hoc networks: A survey,” in *Proc. 2nd Int. Meeting ACCT* , Rohtak, Haryana,India, 2012, pp. 535–541.

[4] T. Anantvalee and J. Wu, “A Survey on Intrusion Detection in Mobile Ad Hoc Networks,” in *Wireless/Mobile Security*. New York: Springer- Verlag, 2008.

[5] L. Buttyan and J. P. Hubaux, *Security and Cooperation in Wireless Networks*. Cambridge, U.K.: Cambridge Univ. Press, Aug. 2007.

[6] D. Dondi, A. Bertacchini, D. Brunelli, L. Larcher, and L. Benini, “Model- ing and optimization of a solar energy harvester system for self-powered wireless sensor networks,” *IEEE Trans. Ind. Electron.*, vol. 55, no. 7, pp. 2759–2766, Jul. 2008.

[7] V. C. Gungor and G. P. Hancke, “Industrial wireless sensor networks: Challenges, design principles, and technical approach,” *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 4258–4265, Oct. 2009.

[8] Y. Hu, D. Johnson, and A. Perrig, “SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks,” in *Proc. 4th IEEE Workshop Mobile Comput. Syst. Appl.*, 2002, pp. 3–13.

[9] Y. Hu, A. Perrig, and D. Johnson, “ARIADNE: A secure on-demand rout- ing protocol for ad hoc networks,” in *Proc. 8th ACM Int. Conf. MobiCom*, Atlanta, GA, 2002, pp. 12–23.

[10] G. Jayakumar and G. Gopinath, “*Ad hoc* mobile wireless networks rout-ing protocol—A review,” *J. Comput. Sci.*, vol. 3, no. 8, pp. 574–582,2007.

[11] D. Johnson and D. Maltz, “Dynamic Source Routing in *ad hoc* wireless networks,” in *Mobile Computing*. Norwell, MA: Kluwer, 1996, ch. 5, pp. 153–181.

[12] N. Kang, E. Shakshuki, and T. Sheltami, “Detecting misbehaving nodes in MANETs,” in *Proc. 12th Int. Conf. iiWAS*, Paris, France, Nov. 8–10,2010, pp. 216–222.

[13] N. Kang, E. Shakshuki, and T. Sheltami, “Detecting forged acknowl- edgements in MANETs,” in *Proc. IEEE 25th Int. Conf. AINA*, Biopolis, Singapore, Mar. 22–25, 2011, pp. 488–494.

[14] K. Kuladinith, A. S. Timm-Giel, and C. Görg, “Mobile *ad-hoc* commu- nications in AEC industry,” *J. Inf. Technol. Const.*, vol. 9, pp. 313–323,2004.

[15] J.-S. Lee, “A Petri net design of command filters for semiautonomous mobile sensor networks,” *IEEE Trans. Ind. Electron.*, vol. 55, no. 4, pp. 1835–1841, Apr. 2008.