Packet Dropping Detection Through HLA Algorithm in Wireless Ad Hoc Networks

Mrs.P.Sundari[1] Mrs.S.Gowrimanohari[2]
[1]Assistant Professor [2]Research Scholar
[1,2]Department of Computer Science and Engineering
[1]Government Arts and Science College-Coimbatore
[2]LRG Government Arts College for Women-Tiruppur

Abstract— Security is one of the most important issues that have attracted a lot of research and development effort in past few years. In multi-hop wireless ad hoc network link error and malicious packet dropping are two sources for packet losses. Whether the losses are caused by link errors only, or by the combined effect of link errors and malicious drop are to be identified, can be known by observing a sequence of packet losses in the network. But 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. Conventional algorithms that are based on detecting the packet loss rate cannot achieve satisfactory detection accuracy because the packet dropping rate in this case is comparable to the channel error rate. Hence to improve the detection accuracy, the correlations between lost packets is identified. The technique called Homomorphic linear authenticator (HLA) based public auditing architecture is developed that allows the detector to verify the truthfulness of the packet loss information reported by nodes. This technique provides privacy preserving, collusion proof, and incurs low communication and storage overheads. A packet-block based mechanism is also proposed, to reduce the computation overhead of the baseline scheme, which allows one to trade detection accuracy for lower computation complexity.

Keywords: Packet Dropping, Secure Routing, Attack Detection, Homomorphic Linear Signature, Auditing

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 adversary 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.

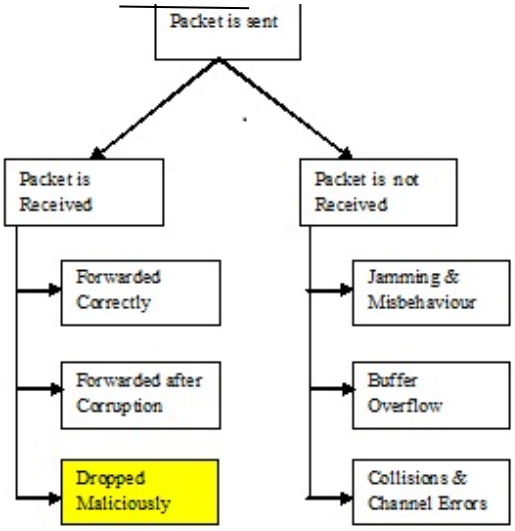


Fig. 1: Overview of Packet Loss

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 are deemed highly critical to the operation of the network. For example, in a frequency-hopping network, these could be the packets that convey frequency hopping sequences for network-wide frequency-hopping synchronization; insider attacker can cause significant damage to the network with low probability of being caught.

The main problem is to combat such an insider attack. In particularly, represents the problem of detecting the occurrence of selective packet drops and identifying the malicious node(s) responsible for these drops. Detecting selective packet dropping attacks is extremely challenging in a highly dynamic wireless environment. Due to the open nature of wireless medium, a packet drop in the network could be caused by harsh channel conditions (e.g., fading, noise, and interference, a.k.a., link errors), or by the insider attacker. In this case, just by observing the packet loss rate is not enough to accurately identify the exact cause of a packet loss.

An accurate algorithm for detecting selective packet drops made by insider attackers is developed. That 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

All rights reserved by *www.ijsrd.com*

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 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 this mechanism lies in how to guarantee that the packet-loss bitmaps reported by individual nodes along the route are truthful. Such truthfulness is essential for correct calculation of the correlation between lost packets.

Therefore, some auditing mechanism is needed to verify the truthfulness of the reported information.

The solution to the above 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. To significantly reduce the computation overhead, a packet-block-based HLA 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

In the year 2003 S. Zhong, J. Chen, and Y. R. Yang proposed a paper titled "Sprite: A simple cheat-proof, credit-based system for mobile ad-hoc networks". Mobile ad hoc networking has been an active research area for several years. How to stimulate cooperation among selfish mobile nodes, however, is not well addressed yet. In this paper, a new technique is used for stimulating cooperation among selfish nodes in mobile ad hoc networks. This system provides incentive for mobile nodes to cooperate and report actions honestly. Compared with previous approaches, this system does not require any tamper-proof hardware at any node. Furthermore, they present a formal model of their system and prove its properties. Evaluations of a prototype implementation show that the overhead of our system is small. Simulations and analysis show that mobile nodes can cooperate and forward each other's messages, unless the resource of each node is extremely low.

In the year 2004 Q. He, D. Wu, and P. Khosla, proposed a paper titled "Sori: A secure and objective reputation-based incentive scheme for ad hoc networks,". In an ad-hoc network, intermediate nodes on a communication path are expected to forward packets of other nodes so that the mobile nodes can communicate beyond their wireless transmission range. However, because wireless mobile nodes are usually constrained by limited power and computation resources, a selfish node may be unwilling to spend its resources in forwarding packets which are not of its direct interest, even though it expects other nodes to forward its packets to the destination. It has been shown that the presence of such selfish nodes degrades the overall performance of a non-cooperative ad hoc network. To address this problem, we propose a secure and objective reputation-based incentive (SORI) scheme to encourage packet forwarding and discipline selfish behavior. Different from the existing schemes, under our approach, the reputation of a node is quantified by objective measures, and the propagation of reputation is efficiently secured by a one-way-hash-chain-based authentication scheme. Armed with the reputation-based mechanism, we design a punishment scheme to penalize selfish nodes. The experimental results show that the proposed scheme can successfully identify selfish nodes and punish them accordingly.

In the year 2004 Y. Xue and K. Nahrstedt, proposed a paper titled "Providing fault-tolerant ad-hoc routing Service in adversarial environments,". Most existing designs of ad hoc networks are based on the assumption of non-adversarial environments, where each node in the network is cooperative and well-behaved. When misbehaving nodes exist in the network, the performance of current routing protocols degrades significantly. Since ad hoc networks, consisting of autonomous nodes, are open and distributed in nature, maintaining a fault-free network environment is extremely difficult and expensive. In this paper, we propose a new routing service named best-effort fault-tolerant routing (BFTR). The design goal of BFTR is to provide packet routing service with high delivery ratio and low overhead in presence of misbehaving nodes. Instead of judging whether a path is good or bad, i.e., whether it contains any misbehaving node, BFTR evaluates the routing feasibility of a path by its end-to-end performance (e.g. packet delivery ratio and delay). By continuously observing the routing performance, BFTR dynamically routes packets via the most feasible path. BFTR provides an efficient and uniform solution for a broad range of node misbehaviors with very few security assumptions. The BFTR algorithm is evaluated through both analysis and extensive simulations. The results show that BFTR greatly improves the ad hoc routing performance in the presence of misbehaving nodes.

In the year 2009 W. Kozma Jr. and L. Lazos, proposed a paper titled "Dealing with liars: Misbehaviour identification via Renyi-Ulam games", this paper address the problem of identifying misbehaving nodes that refuse to forward packets in wireless multi-hop networks. They map the process of locating the misbehaving nodes to the classic R´enyi-Ulam game of 20 questions. Compared to previous methods, this mapping allows the evaluation of node behaviour on a per-packet basis, without the need for energy-expensive overhearing techniques or intensive acknowledgment schemes. Furthermore, it copes with colluding adversaries that coordinate their behavioural patterns to avoid identification and frame honest nodes. They show via simulations that this algorithms reduce the communication overhead for identifying misbehaving nodes by at least one order of magnitude compared to other methods, while increasing the identification delay logarithmically with the path size.

In the year 2010 C. Wang, Q. Wang, K. Ren, and W. Lou, proposed a paper titled "Privacy-preserving public auditing for data storage security in cloud computing," Cloud Computing is the long dreamed vision of computing as a utility, where users can remotely store their data into the cloud so as to enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources. By data outsourcing, users can be relieved from

All rights reserved by *www.ijsrd.com*

the burden of local data storage and maintenance. However, the fact that users no longer have physical possession of the possibly large size of outsourced data makes the data integrity protection in Cloud Computing a very challenging and potentially formidable task, especially for users with constrained computing resources and capabilities. Thus, enabling public auditability for cloud data storage security is of critical importance so that users can resort to an external audit party to check the integrity of outsourced data when needed. To securely introduce an effective third party auditor (TPA), the following two fundamental requirements have to be met: 1) TPA should be able to efficiently audit the cloud data storage without demanding the local copy of data, and introduce no additional on-line burden to the cloud user; 2) The third party auditing process should bring in no new vulnerabilities towards user data privacy.This paper, utilize and uniquely combine the public key based homomorphic authenticator with random masking to achieve the privacy-preserving public cloud data auditing system, which meets all above requirements. To support efficient handling of multiple auditing tasks, we further explore the technique of bilinear aggregate signature to extend our main result into a multi-user setting, where TPA can perform multiple auditing tasks simultaneously. Extensive security and performance analysis shows the proposed schemes are provably secure and highly efficient.

In the year 2003 R. Rao and G. Kesidis, proposed a paper titled "Detecting malicious packet dropping using statistically regular traffic patterns in multihop wireless networks that are not bandwidth limited,". Ad hoc networks are gaining presence with the proliferation of cheap wireless devices and the need to keep them connected. Individual applications and larger missions, such as those of tactical sensor networks, require secure data transmission among wireless devices. Security remains a major challenge for such networks. Current protocols employ encryption and authentication techniques for secure message exchange, but given the limitations and innately insecure nature of ad-hoc networks, such mechanisms may not suffice. A security breach can, for example, be a network-level denial-of-service (DoS) attack, passive eavesdropping, or physical layer jamming to degrade communication channels. In a multi-hop network, an intruder node can degrade communication quality by simply dropping packets that are meant to be relayed (forwarded). The network could then misinterpret the cause of packet loss as congestion instead of malicious activity. They suggest that traffic transmission patterns be selected to facilitate verification by a receiver. Such traffic patterns are used in concert with suboptimal MAC that preserves the statistical regularity from hop to hop. This general technique for intrusion detection is therefore suitable for networks that are not bandwidth limited but have strict security requirements, e.g., certain kinds of tactical sensor network

III. PROBLEM STATEMENT AND SYSTEM MODELS

A. Problem Definition:

By analyzing the existing approaches the following problems were arose to satisfy :

1) Identify the nodes on the path from source to destination that drop packets maliciously.
2) Detection should be performed by a public auditor that does not have knowledge of the secrets of the nodes.
3) When a malicious node is identified, the auditor should be able to construct a publicly verifiable proof of the misbehavior of that node.
4) The Detection mechanism should incur low communication and storage overhead.

B. System Models:

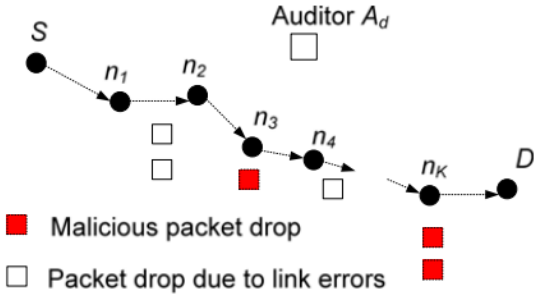


Fig. 2: System Model Diagram

1) Network Model:

Assume an arbitrary path PSD in a multi-hop wireless adhoc network. The source node S continuously sends packet to the destination node D through intermediate nodes n1……..nK, where ni is the upstream node of ni+1. Source S is aware of the route PSD either by Dynamic Source Routing or by trace route operation. The network topology and link characteristics remain unchanged for a long period. Example: wireless mesh network(WMNs).

2) Channel Model:

The wireless channel model for each hop along PSD is 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 channel model, it only require that the sequence of sojourn times for each state follows a stationary distribution and the autocorrelation function of the channel state. By observing whether the transmissions are successful or not, the receiver obtains a realization of the channel state.

Auditor

There is an independent auditor A_d in the network. A_d is independent in the sense that it is not associated with any node in P_{SD} 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.

S receives feedback from D when D suspects that the route is under attack. Once being notified of possible attacks, S submits an attack-detection request (ADR) to A_d. To facilitate its investigation, A_d needs to collect certain information (elaborated on in the next sec-tion) from the nodes on route P_{SD}. Each node must reply to A_d's inquiry, otherwise the node will be considered as misbehaving. Normal nodes will reply with truthful information, but malicious nodes may cheat. At the same time, for privacy reasons, A_d cannot determine the content of the normal packets delivered over P_{SD} from the information collected during the auditing.

3) Adversarial Model:

The goal of the adversary is to degrade the network's performance by maliciously dropping packets while remaining undetected. The malicious node has knowledge of

All rights reserved by *www.ijsrd.com*

the wireless channel, and is aware of the algorithm used for misbehavior detection. It has the freedom to choose what packets to drop. For example, in the random-drop mode, the malicious node may drop any packet with a small probability p_d. In the selective-mode, the malicious node only drops packets of certain types.

The following form of collusion between malicious nodes is considered: A covert communication channel may exist between any two malicious nodes, in addition to the path connecting them on P_{SD}. As a result, malicious nodes can exchange any information without being detected by A_d or any other nodes in P_{SD}. Malicious nodes can take advantage of this covert channel to hide their misbehavior and reduce the chance of being detected.

IV. SYSTEM ARCHITECTURE

Initially the network is configured with calling the Node configure function with number of nodes. And then Link create 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 Add bit 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 cannot be decrypted.

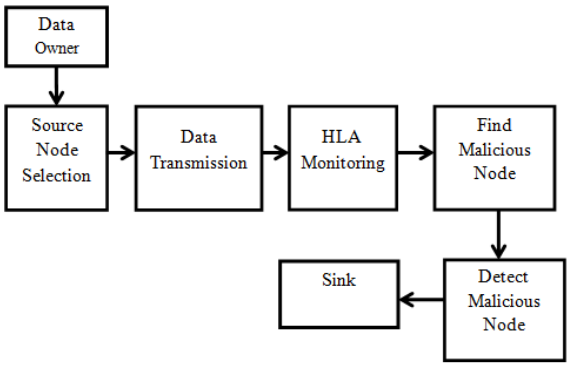


Fig. 3: System Architecture

V. SYSTEM PHASE

A. Setup Phase:

The network path between source and destination is established in this phase. Every network path has a network controller. All the nodes are sensed with this network controller. In this phase, Source node decides symmetric key crypto-system(encryptionkey, decryptionkey) and symmetric keys. Source securely distributes decryptionkey and a symmetric key to all the nodes in the network path. The key distribution based on the public-key crypto system. Source also announces two hash functions to all nodes in the path that will be used for message authentication purpose. Besides symmetric key distribution Source also needs to setup its HLA keys.

B. Packet Transmission Phase:

In this phase, Source transmits packets along the path according to the following steps.

Before sending out a packet, Source computes hash message(H1) and generates the HLA signatures of hash message for the node next to the Source. These signatures are then sent together with the packet, to the route by using a one-way chained encryption that prevents upstream node from deciphering the signatures intended for downstream nodes. The Message Authentication Code(MAC)in each stage is computed according to the MAC hash function(H2). After calculating the Tag t Source concatenate packet and tag together and sends it to the next node.

When the node receives the packet from source, it extracts packet and ŝ(encrypted signature)and MAC key(ŝ) from the packet. Then the node verifies the integrity of ŝ. If the test is true, then the node decrypts ŝ to extract signature and the tag from the decrypted text. It stores the signature and the calculated hash value(H1) in a database. This process is repeated at every intermediate node.

C. Audit Phase:

This phase is triggered when the public auditor receives an ADR message from Source. The ADR message includes the id of the nodes on the network path, ordered in the downstream direction, Source's public key information, the sequence number of most recent packets sent by Source and the sequence numbers of the subset of these packets that were received by Destination. It is assumed that the information sent by Source and Destination is truthful.
Auditor conducting the auditing process as follows:

Auditor submits a random challenge vector to the node Based on the information in the database of each node, the node generates a packet-reception bitmap (b, b =1 if packet received, b=0 otherwise). Based on these information node submits bitmap, hash value(H1) and signature as a proof of the packets its received. Auditor checks the validity of the hash value and signature. If the validation is true, then Auditor accepts that node received packets as reflected in bitmap. Otherwise, the Auditor rejects the bitmap and judges that not all packets claimed in the bitmap are actually received by the node, so that node is malicious node.

D. Detection Phase:

The public auditor enter into this phase, after receiving and auditing the reply to its challenge from all nodes on the network path. The main tasks of the auditor 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 auto correlation function for the packet loss on each hop, and deciding whether malicious behavior is present. Once being detected, the malicious node will be marked and excluded from the route to overcome its damage. This detection process applies to one end-to-end path.

All rights reserved by *www.ijsrd.com*

Public verifiability : After each detection, Auditor is required to publish the information it received from involved nodes, so that the node can verify all calculation has been performed correctly.

Privacy Preserving : This property ensures that publishing the auditing information will not compromise the confidentiality of the communication.

VI. EXPERIMENTAL RESULTS

The performance of the system is explained here with appropriate screenshots.

As a mobile based application it requires to launch MIDlet to setup a network connection.

Screen.1 Launching MIDlet

Each node in the network path need an identification (Node ID) to explore themselves in the path. It is depicted as a node configuration

Screen.2 Node Configuration

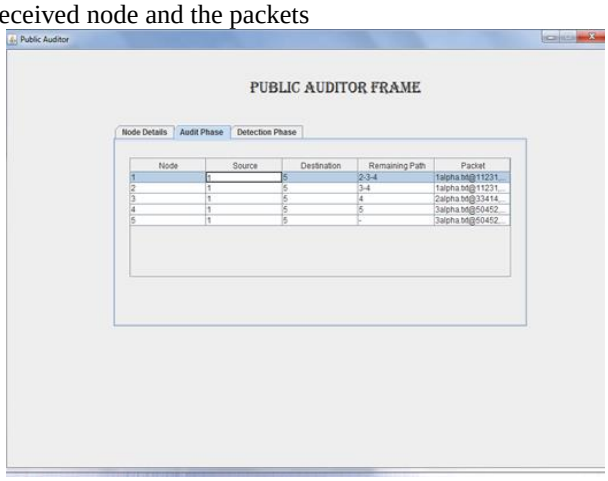
In the following screen represents, the file encryption process, where file name, content of the file and number of blocks details is required

Screen.3 Packet Generation

The following screen reveals, the encrypted data, the destination node ID and the path (i.e nodes between source to destination) .

Screen.4 Encrypted Data

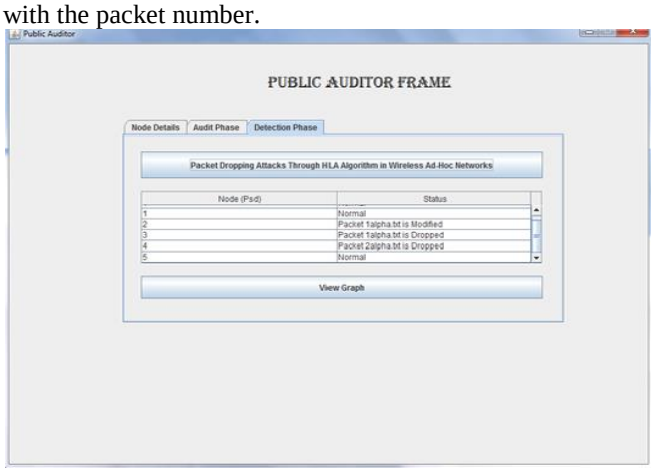
The following screen depicts the Auditing details such as Source Node, Sink Node, Node on which the packet is to be, Remaining Path to reach the Sink node from the Received node and the packets



Screen.5 Audit Phase Details

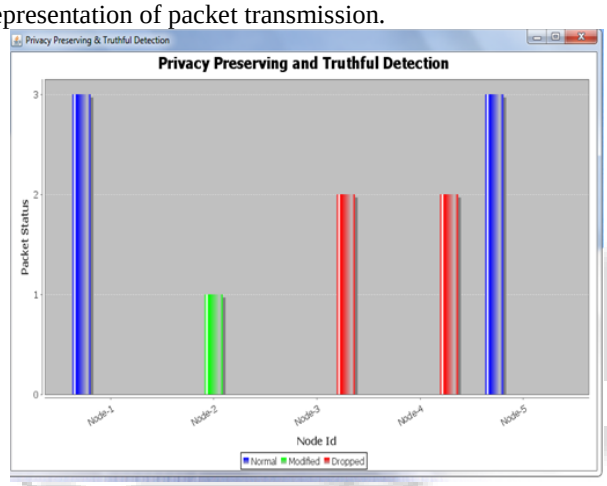
All rights reserved by *www.ijsrd.com*

In the following screen, the detection details is showed, where the nodeID's in the path and the packet status either normal or modified or dropped is displayed with the packet number.



Screen.6 Detection Phase

The following graph represent the pictorial representation of packet transmission.



Screen.7 Graphical Representation

ACKNOWLEDGMENT

I am very grateful and would like to thank my guide and teacher Prof. P.Sundari for her advice and continued support without which it would not have been possible for me to complete this report. I am also very grateful and also like to thank Head of The Department and the entire Computer Science Department, faculty and staff, for helping me in every possible manner during my course of study for this subject.

VII. CONCLUSION

Detecting malicious packet dropping is a critical issue in networks. The conventional detection algorithms face several challenges that is fully overcame by the proposed approach. Exploiting the correlation between the lost packets improves the accuracy in detecting malicious packet drops. The truthfulness of the bitmap reported by each node is ensured by the cryptographic primitive which enables a public auditing architecture which is developed by HLA. This approach provides high detection accuracy. The randomized dispersive routes on detecting the malicious nodes effectively overcome the packet drop. Experimental results showed that proposed method performs well.

Even though some open issues are to be considered in the future work. Changes in topology and link-characteristics are to be considered. In this paper we have assumed the source and destination are truthful, but malicious source and destination is a possibility which needs to be considered.

REFERENCES

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

[2] S. Zhong, J. Chen, and Y. R. Yang, "Sprite: A simple cheat-proof, credit-based system for mobile ad-hoc networks," in Proc. IEEE INFOCOM Conf., 2003, pp. 1987–1997.

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

[4] W. Kozma Jr. and L. Lazos, "Dealing with liars: Misbehavior identification via Renyi-Ulam games," presented at the Int. ICST Conf. Security Privacy in Commun. Networks, Athens,Greece, 2009.

[5] Q. He, D. Wu, and P. Khosla, "Sori: A secure and objective reputation-based incentive scheme for ad hoc networks," in Proc. IEEE Wireless Commun. Netw. Conf., 2004, pp. 825–830.

[6] R. Rao and G. Kesidis, "Detecting malicious packet droppingusing statistically regular traffic patterns in multihop wireless networks that are not bandwidth limited," in Proc. IEEE GLOBECOM Conf., 2003, pp. 2957–2961.

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

[8] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in Proc. IEEE INFOCOM Conf., Mar. 2010, pp. 1–9.

[9] Y. Xue and K. Nahrstedt, "Providing fault-tolerant ad-hoc routing service in adversarial environments," Wireless Pers. Commun., Special Issue Secur. Next Generation Commun., vol. 29, no. 3, pp. 367– 388, 2004.

[10] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in Proc. ACM MobiHoc Conf., 2005, pp. 46–57.

All rights reserved by *www.ijsrd.com*