Audit Based Misbehaviour Detection Of Nodes In Wireless Adhoc Network

Ms. Vrushali M. Bodhankar
M.E [C.S.E] ,
Computer Science and Engg,Department
DIEMS, Aurangabad

Ms. Ashwini S. Gaikwad
Assistant Professor
Computer Science and Engg, Department
DIEMS, Aurangabad.

ABSTRACT

In the network there are the packet losses. But packet losses are done only by link error. The attacks too causes because of malicious nodes are a component of the path use its data of communication to packet loss. Small number of packets risky to system performance and intend to make use of the correlation among missing packets. Packets dropping rates throughout this is associated to the channel error rate standard algorithms is predicated on the packets loss rate that can't do well adequate detection accuracy. The computation overhead of the existing system decreased by using a packet -block-based mechanism designed which allows to deal with revealing accurateness for lower computation difficulty. In addition to make sure truthful computation of these correlations use a homomorphic linear authenticator (HLA) based audit mechanism that allow the detector to prove the regularity of the packet loss information given by nodes. This formation is privacy preserve collusion verification and recovers low communication and storage overheads. The packet dropping rate during this case is resulting to the channel error rate standard algorithms that are supported the packet loss rate cannot achieve satisfactory detection accuracy. So the proposed method this paper recover the limits of existing system detection truthfulness and improve the limitations of Existing system Detection Accuracy and Computation overhead are improved. For that DES algorithm And Bayes model used

Keywords/ Index Term — Packet dropping, secure routing, attack detection, homomorphic linear authenticator.

1. INTRODUCTION

In a multi-hop wireless network, nodes cooperate in relaying/routing traffic someone will exploit this cooperative nature to launch attacks. Eventually, such a severe denial-of-service (DoS) attack can paralyze the network by partitioning its topology. In the most severe kind, the malicious node simply stops forwarding every packet received from upstream nodes, completely disrupting the path between the source and the destination. For example, the adversary may first pretend to be a cooperative node in a route, the adversary starts dropping packets. The continuous presence of extremely high packet loss rate at the malicious nodes makes this type of attack easy to be detected [2] this is the first case. In Second case, once being detected, these attacks are weak. For example, in case the attack is detected but the malicious nodes are not identified, one can use the randomized multi-path routing algorithms [2], [3] to circumvent the black holes generated by the attack, probabilistically eliminating the attacker's threat. Even though constant packet dropping will effectively degrade the performance of the network, from the attacker's point of view such an "always-on" attack has its disadvantages. If the malicious nodes are also identified, their threats can be completely eliminated by simply deleting these nodes from the network's routing table. A malicious node that's a part of the route will exploit its information of the network protocol and therefore the communication context to launch corporate executive attack—an attack that's irregular but can achieve the same performance degradation effect as a persistent attack at a much lower risk of being detected. Specifically, the malicious node might measure the importance of varied packets, so drop the tiny quantity that area unit deemed extremely essential to the operation of the network. In this paper, we are interested in combating such an insider attack. In particular, we are interested in the problem of detecting the occurrence of selective packet drops and identifying the malicious node(s) responsible for these drops. In a frequency-hopping network, these could be the packets that carry frequency hopping sequences for network-wide frequency-hopping system; in an ad hoc cognitive radio network, they could be the packets that carry the idle channel lists that are used to establish a network-wide control channel. By targeting these highly critical packets, the authors in [1], [4], [5] have shown that an intermittent insider attacker can cause significant damage to the network with low probability of being caught. Specifically, due to the open environment of wireless medium, a packet drop in the network could be caused by harden channel conditions (e.g., fading, noise, and interference, link errors),or by the insider attacker. Detecting selective packet-dropping attacks is very difficult during a extremely dynamic wireless atmosphere. The difficulty comes from the need that we'd like to not solely observe the place (or hop) wherever the packet is born, but also identify whether

the drop is intentional or unintentional. So, the business executive assailants will camouflage beneath the background of harsh channel conditions. In this case, just by observing the packet loss rate is not enough to accurately identify the exact reason of a packet loss. In associate open wireless atmosphere, link errors are quite significant, and may not be significantly smaller than the packet dropping rate of the insider attacker. On the opposite hand, for the little variety 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 over link errors, in order to achieve an acceptable detection accuracy. The above problem has not been well addressed in the literature. As discussed in Section 2, 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's no ought to account for the impact of link errors. 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 standing of every packet during a sequence of consecutive packet transmissions. The basic plan behind this methodology is that even if malicious dropping could end in a packet loss rate that's equivalent to traditional channel losses, the stochastic processes that characterize the 2 phenomena exhibit completely different correlation structures (equivalently, completely different patterns of packet losses).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.

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. 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. 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. 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. 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.

2. LITERATURE SURVEY

The first class aims at high malicious dropping rates, wherever most (or all) lost packets area unit caused by malicious dropping. In this case, the impact of link errors is neglected. Most related work falls into this category. 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 base of the second sub category depend on reputation systems. This name information is propagated from time to time throughout the network and is working as a very important measure for choosing routes. Therefore, a malicious node will be excluded from any route end-to-end or hop-to-hop is the base of third sub category in which acknowledgements to directly establish the hops where packets are lost. Based on the method used to recognize the attacking nodes, these can be further classified into four subcategories. The first sub-category is based on credit systems [4], [8], [9]. A credit system provides an reason for support. A node receives credit by relaying packets for others, and uses its credit to send its own packets. Similarly, the method in [6], [3] traces the forwarding records of a particular packet at each intermediate node by formulating the tracing problem . The first hop where the packet is no longer forwarded is considered for misbehaving. 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 [7] utilizes Bloom filters to construct proofs for the forwarding of packets at each node. By examining the relayed packets at serial hops on a route, one can identify suspicious hops that exhibit high packet loss rates. All strategies mentioned on top of don't perform well once malicious packet dropping is extremely selective. More specifically, for the credit system-based method, a malicious node may still receive enough credits by forwarding the majority of the packets it receives from upstream nodes. The works in [3] and [7] proposed to detect malicious packet dropping by counting the number of lost packets. If the quantity of lost packets is significantly larger than the expected packet loss rate created by link errors, then with high probability a malicious node is contributing to packet losses. Certain data of the wireless channel is critical during this case. The authors in [6] proposed to shape the traffic at the MAC layer of the source node according to a certain statistical distribution, so that intermediate nodes are able to guess the rate of received traffic by sampling the packet coming times. By comparing the source traffic rate with the expected received rate, the detection algorithm decides whether the differences in rates, is within a reasonable range such that the difference can be considered as being caused by normal channel loss or caused by malicious dropping, In the reputation-based method, the malicious node can maintain a convincingly good status by forwarding most of the packets to the next

hop. While the Bloom-filter theme is in a position to produce a packet forwarding proof, the correctness of the proof is probabilistic and it may contain errors. For highly selectively attacks (low packet-dropping rate), the intrinsic error rate of Bloom filter significantly undermines its detection accuracy. As for the acknowledgement-based technique and every one the mechanisms within the second class, simply numeration the quantity of lost packets doesn't provides a sufficient ground to sight the real culprit that is caused packet losses. The effort in the literature on this problem has been quite low level, and there is a few related works. Note that the cryptographic methods proposed in [4] to counter selective packet jamming target a different issue than the detection problem studied in this paper

2.1 Network and Channel Models

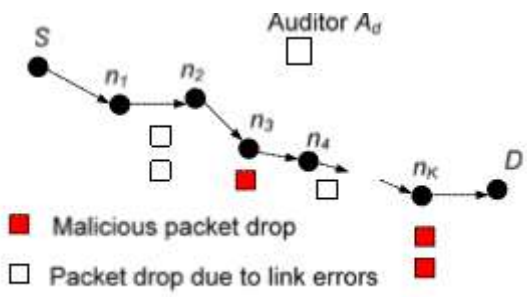

Fig 1 :- Network and Attack Model

Fig 1 Shows Network and Attack model .We model the wireless channel of each hop along PSD as a random process that alternates between good and bad states. Transmission of packets in good state are successful and in bad state are unsuccessful., 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 $f_e(i)$, where i is the time lag in packets, is also stationary. We concentrate 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., $f_e(i)$ is stationary). For highly mobile networks checking malicious packet drops may not be a concern for highly mobile, because the fast-changing topology of such networks makes route interruption is the main cause for packet losses. In this case, maintaining stable connectivity between nodes is a greater point than detecting malicious node. The function $f_e(i)$ can be calculated using the probing approach in [1]. Sequences of M packets are transmitted repeatedly over the channel. whether the transmissions are successful or not, the receiver obtains understanding of the channel In this sequence"1"

$$(a_{1,,,,,,} a_M), where\ a_j \in \{0,1\} for\ j = 1,,,,,,, M$$

denotes the packet was successfully received, and "0" denotes the packet was dropped. $f_e(i)$ is derived by computing the autocorrelation function of this sample sequence: , where the expectation is calculated $f_e(i)^{def} = E\{a_j, a_{j+1}\} for\ i = 0,,,,, M$ 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 f_c is guaranteed by the WSS assumption of the wireless channel. The measurement of $f_c(i)$ can take place online or offline. A detailed discussion on how $f_c(i)$ 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. 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 from the nodes on route PSD. We assume that each such node must reply to Ad's analysis, otherwise the node will be considered as misbehaving. We suppose that normal nodes will reply with truthful information, but malicious nodes may take advantage of. For confidentiality reasons, we require that Ad cannot determine the content of the normal packets delivered over PSD from the information collected during the auditing.

System Architecture :-

Overview

The most important challenge in our method is how to guarantee the packet loss bitmaps reported by individual nodes along the route are truthful. This can be achieved by using the HLA scheme for detecting selective packet dropping attack made by malicious node. The high detection accuracy is achieved by using entropy method to detect malicious activities.

In Fig 2 shows source, destination and three intermediate nodes where packet is transmitted and malicious node drops selective packets along with link error because of which some packets are dropped. Each intermediate node sends an recognition to the source after receiving the key during key transmission stage. Each intermediate node provides a bitmap describing the lost or received status of each packet in a sequence of

consecutive packet transmission to the auditor

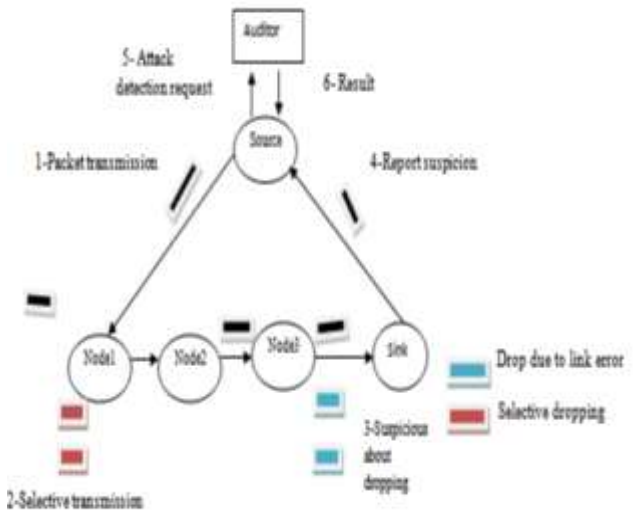

Fig 2: Packet Transmission From Source To Destination

Modules 1. Setup phase

The source selects the route by Ad hoc On-demand Distance Vector (AODV) Routing Protocol. In this phase, S decides on a symmetric-key crypto-system (encryptkey; decryptkey) and K symmetric keys key1; . . . ; keyK, where encryptkey and decryptkey are the keyed encryption and decryption functions, respectively. S securely distributes decryptkey and a symmetric key keyj to node nj on PSD, for j = 1; . . .;K.

2. Packet Transmission Phase

When the setup phase completes, S enters the packet transmission phase. It generates the HLA signatures for each packet. S transmits packets to PSD according to the following steps. Before transfer a packet Pi, where i is a sequence number that uniquely identifies Pi, S computes r_i = H1(Pi) and generates the HLA signatures of ri for node nj , as follows:

I.. $\tilde{S}_{ki} = encrypt_{keyk}(S_{ki})$,

II. $Ƭ_{ki}=\tilde{S}_{ki} \| MAC_{keyk}(\tilde{S}_{ki})$,

III. $\tilde{S}_{k-1i} = encrypt_{keyk}(S_{k-1i} \| Ƭ_{ki})$,

IV. $Ƭ_{k-1i}= \tilde{S}_{k-1i} \| MAC_{keyk-1}(\tilde{S}_{k-1i})$,

:

V. $\tilde{S}_{ji} = encrypt_{keyj}(S_{ji} \| Ƭ_{j+1i})$,

VI $Ƭ_{ji}=\tilde{S}_{ji} \| MAC_{keyj}(\tilde{S}_{ji})$,

:

:

VI $\tilde{S}_{1i} = encrypt_{key1}(S_{1i} \| Ƭ_{2i})$,

VIII $Ƭ_{1i}=\tilde{S}_{1i} \| MAC_{key1}(\tilde{S}_{1i})$,

IX $S_{ij} = [H2(i\|j)u_{ri}]x$, for j = 1, . . . ,K

(1) These signatures are then sent together with Packet to the route by using a one-way chained encryption that prevents an upstream node from deciphering the signatures intended for downstream nodes.

For encryption of keys RSA Algorithm is used

RSA Algorithm:

It is most popular and asymmetric key cryptographic algorithm. It may used to provide both secrecy and digital signature. It uses the prime no. to generate the public and private key based on mathematical fact and multiplying large numbers together. It uses the block size data in which plaintext and cipher text are integers between 0and n1 for some n values. Size of n is considered 1024 bits or 309 decimal digits. In this two different keys are used for encryption and decryption purpose. As sender knows encryption key and receiver knows decryption key [4].

Following steps are followed in RSA to generate the public and private keys [8, 10]:

I. Choose large prime numbers p and q such that p~=q

II Compute n=p*q

III Compute φ (pq) = (p-1)*(q-1)

IV Choose the public key e such that gcd (φ (n), e) =1; 1<e< φ (n)

V Select the private key d such that d*e mod φ (n) =1

So in RSA algorithm [10] encryption and decryption are performed as

I . Encryption
Calculate cipher text C from plaintext message M such that
C = M ^e mod n

II . Decryption
M = C^d mod n = M^ed mod n

3. 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, 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. Ad submits a random challenge vector $\vec{c}_j = (c_{j1}, . . . , c_{jM})$ to node n_j , j = 1, . . . ,K, node nj generates a packet-reception bitmap $\vec{b}_j = (b_{j1}, . . .b_{jM})$,

4. Detecting 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 entropy value for the packet loss on each hop, and deciding whether malicious behavior is present or not. The auditor calculates packet loss per-hop bitmap $\vec{m}_j = (m_{j1}, \ldots, m_{jM})$ where j=1,2…….K where K is the total number of intermediate nodes. Then auditor calculates entropy method [10] for each sequence

$\vec{m}_j = (m_{j1}, \ldots, m_{jM})$, j = 1, . . . ,K, as follows:

$$\gamma_j(i) = \frac{\sum_{k=1}^{M-i} m_{jk} m_{jk+i}}{M-i}, \quad \text{for } i = 0, \ldots, M-1; j = 1, \ldots, K.$$

Auditor then calculates the relative difference between γ_j And the ACF of wireless channel f_c as

$$\epsilon_j = \sum_{i=0}^{M-1} \frac{|\gamma_j(i) - f_c(i)|}{f_c(i)}.$$

3. PROPOSED METHODOLOGY

Detection accuracy and computation overhead are the limitations of existing system.To Overcome this problem proposed methodology is used.In Proposed system DES algorithm is used for encryption is used .Encryption is used Transmission phase and Bayes model used for getting result in Detection phase.

1. DES ALGORITHM

[1] DES [10] takes an input of 64-bit long plaintext and 56-bit key (8 bits of parity) and generates output of 64 bit block.

[2] The plaintext block is subject to an shift the bits around.

[3] The 8 parity bits are removed from the key by subjecting the key to its Key Permutation.

[4] The plaintext and key are processed in of:

a. The key is split into two 28 bit halves
b. Each half of the key is shifted (rotated) by one or more two bits depending on the round
c. The halves are recombined and subject to a compression permutation to reduce the key from 56 bits to 48 bits. This compressed key is used to encrypt this round's plaintext block.
d. The rotated key halves from step 2 are used in next round
e. The data block is split into two 32-bit halves.
f. One half is subject to an Expansion Permutation to increase its size to 48 bits.
g. Output of step 6 is exclusive-OR'ed with the 48-it compressed key from step 3.
h. Output of step 7 is fed into an S-box, which substitutes key bits and reduces the 48-bit block back down to 32-bits.
i. Output of step 8 is subject to a P-box to permute the bits. j. The output from the P-box is exclusive OR'ed with other half of the data block. k. The two data halves are swapped and become the next round's input.

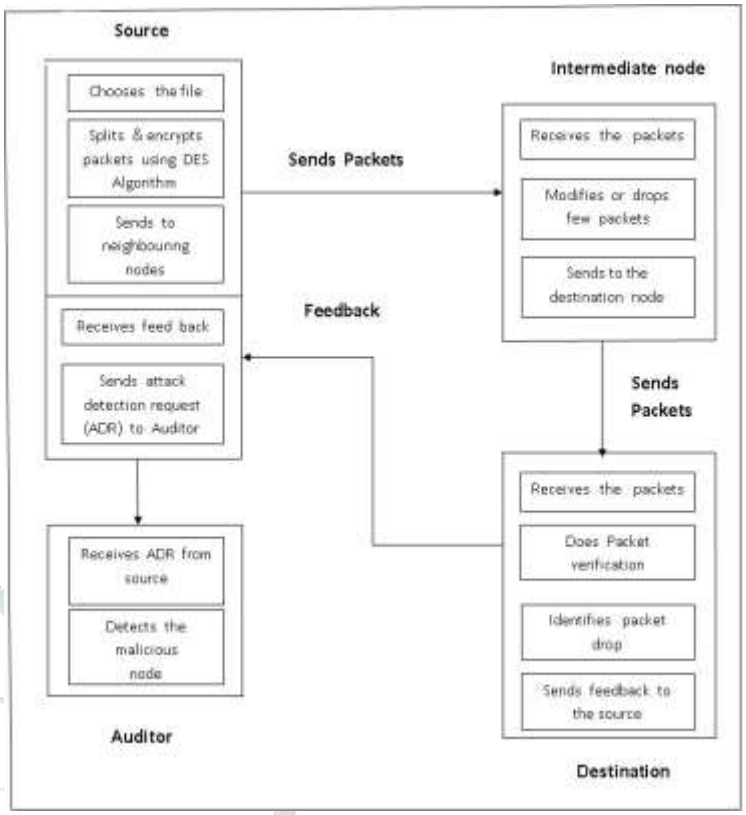

Fig 3 :-Working of Proposed model

2. Bayesian model

To make any important decision an entity takes an advice from other entities who have expertise in the field or knowledge. These experts also give their advice based on accumulated knowledge, experience and other information .The automation systems that take such decision are called expert systems. Probabilistic model can also be used to implement an expert system in which we can consider the uncertain expert knowledge to take a decision. Probabilistic model can use either classical approach in which based on repeated trials probable outcome can be find out, or Bayesian model [11] [12] which uses degree of persons belief that an event is occurred based on past experiences Bayesian model is widely used to calculate trust value of a mobile node from collecting evidence and past experiences This model is based on Bayes' rule that is used to calculate conditional probability of b given a from conditional probability of a given b.

P (b|a) = (p (a|b) * p (b)) /p (a)

From Beta distribution, trust can be calculated as

Depending on the Bayesian model proposed formula for the relative difference relative γ_j and the ACF of wireless channel f_c as

$$\varepsilon_j = \Pr(\gamma_j(i) \mid f_c(i))$$

4. RESULT ANALYSIS

To improve the detection accuracy and computation overhead we use this proposed method. In which we use DES and Bayes model . Three parameters Delay, Throughput, and Admin count are improved in Proposed methodology. Delay and Throughput covered for detection accuracy and Admin count are covered for computation overhead. Network delay is an important design and performance characteristic of a computer or telecommunication network . The delay of a network specifies how long it takes for a bit of data to travel across the network from one communication endpoint to another. It is typically measured in multiples or fractions of seconds. Delay may differ slightly, depending on the location of the specific pair of communicating endpoints. Throughput refers to the performance of tasks by a computing service or device over a specific period. It measures the amount of completed work against time consumed and may be used to measure the performance of a processor, memory and/or network communications. Network throughput refers to the average data rate of successful data or message delivery over a specific communications link. Network throughput is measured in bits per second (bps). A common misconception on measuring network throughput is that measuring the time it takes to upload or download a large file is the maximum throughput of a network. This method does not take into account communications overhead such as Network receiver window size, machine limitations or network latency

In proposed system Admin Count is also improved Admin count is the number of malicious nodes. Using the proposed methodology we improve the detection of malicious nodes

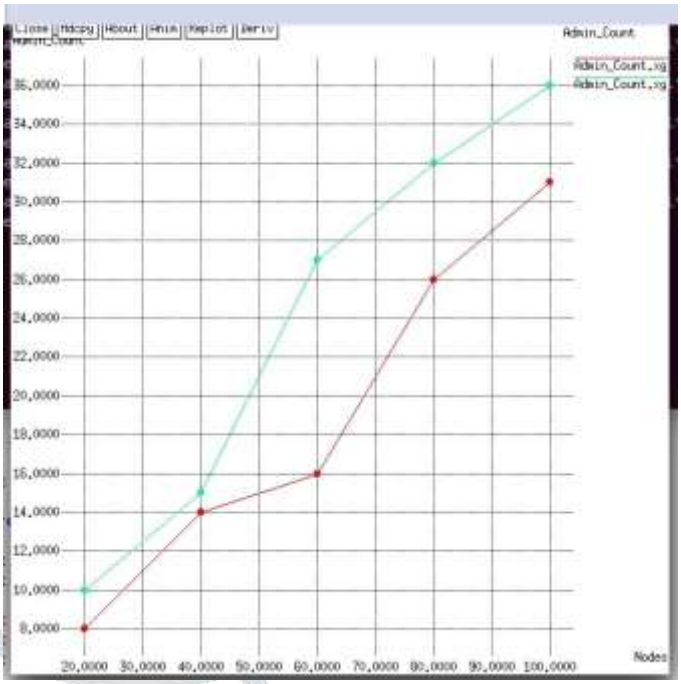

Fig 4 Admin_Count Existing and Proposed System Graph

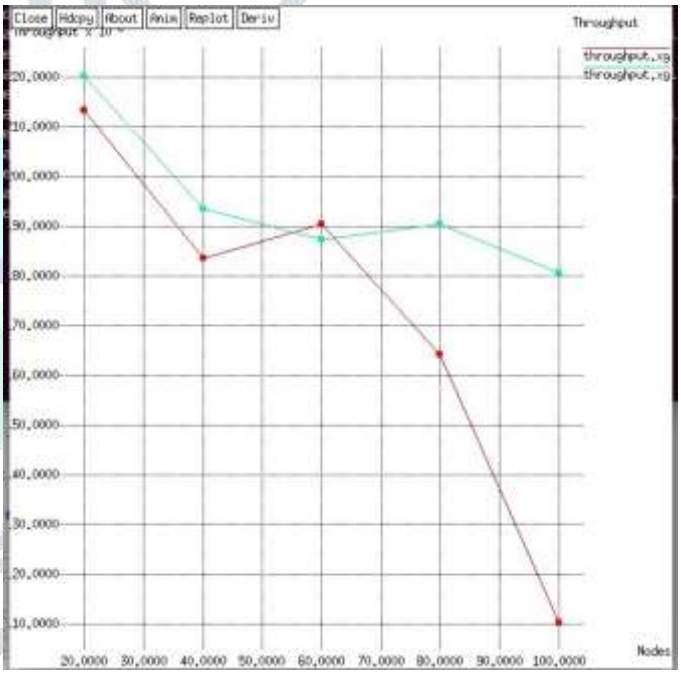

Fig 5 Throughput Existing and Proposed System Graph

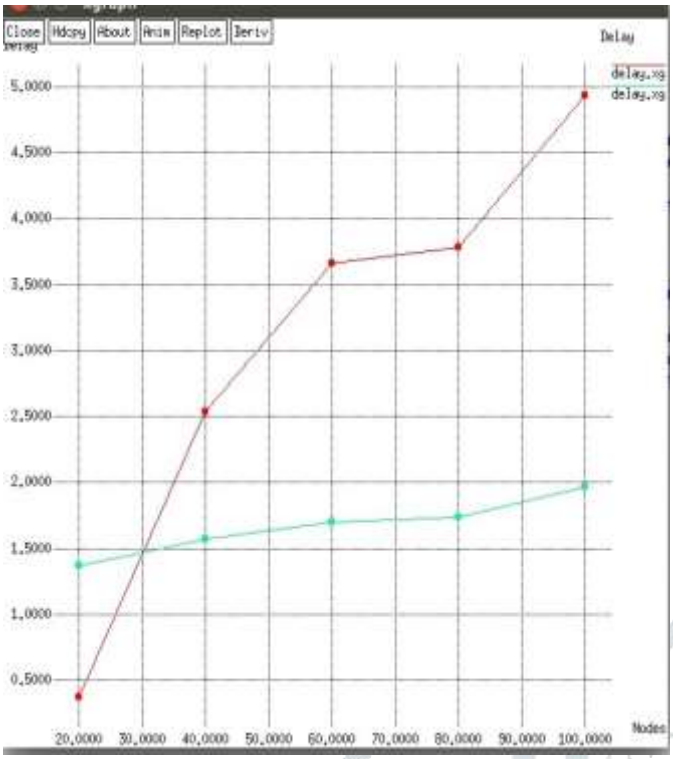

Fig6 Delay Existing and Proposed System Graph

Table 1: Admin_Count

Nodes	20	40	60	80	100
Existing Algorithm	8	14	16	26	31
Proposed Algorithm	10	15	27	31	36

Table 2: Throughput in bps(Bits per second) or pps(Packets per second)

Nodes	20	40	60	80	100
Existing Algorithm	212	182	190	162	110
Proposed Algorithm	220	192	188	190	180

Table 3 : Delay in ms (mili second)

Nodes	20	40	60	80	100
Existing Algorithm	0.4	2.5	3.6	3.7	5
Proposed Algorithm	1.4	1.6	1.7	1.8	2.0

Fig 3,4,5 shows Admin count,Throughput and Delay respectively.Table 1,2,3 shows Numerical values for Existing and Proposed system. Limitations Occur in Existing System are overcome in proposed system

CONCLUSION

The packets dropping rates during this is corresponding to the channel error rate, standard algorithms is predicated on sleuthing the packets loss rate that can't succeed satisfactory detection accuracy. To reduce the computation overhead of the baseline schema, a packet - block-based mechanism is additionally planned, which allows one to trade detection accuracy for lower computation complexity. Furthermore, to ensure truthful calculation of these correlations, we develop 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. So this paper improve the limitations of Existing system Detection Accuracy and Computation overhead are improved. For that DES algorithm And Bayes model used.To improve encryption results and detection results proposed system used.In this way we get the better results.

REFERENCES

[1] 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.

[2] 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.

[3] 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.

[4] 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.

[5] P. Papadimitratos and Z. Haas, "Secure message transmission in mobile ad hoc networks," Ad Hoc Netw., vol. 1, no. 1, pp. 193–209, 2003.

[6] T. Shu, S. Liu, and M. Krunz, "Secure data collection in wireless sensor networks using randomized dispersive routes," in Proc. IEEE INFOCOM Conf., 2009, pp. 2846–2850.

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

[8] 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.

[9] 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.

[10] Aman kumar,Dr.Sudesh Jakhar,Sunil Makkar "Comparitive Analysis between DES and RSA Algorithms" vol2,Issue7,July 2012,ISSN:2277 128X

[11] Jaya Soni, Deepak Xaxa "An Improved Naïve Bayes Classifier for Intrusion Detection System" Vol 5,Issue 6,June 2016,ISSN 2347-8616

[12] Pranav Kulkarni "Design of Hierarchical Intrusion Detection Unit for AD-HOC Networks Based on Baysian Networks"