International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 06 | June 2020 www.irjet.net p-ISSN: 2395-0072

HASH BASED INTRUSION DETECTION SYSTEM FOR MANET

Jeniton S[1], Thulasi K[2]

[1-5]*Department of Electronics and Communication Engineering, RAAK Collegeof Engineering and Technology, Puducherry, India.*

Abstract - In recent years migration from wired network to wireless network has been a global trend. The quality of free movement and scalability made wireless network indispensable in all walks of life. Among all other networks mobile ad hoc network (MANET) has become very popular by providing communications without any fixed structure or features of a system. In this Project a new intrusion detection technique based on Hash function is specially designed for MANETs. In hash function Message Digest 5 (MD5) is used for intrusion detection system to reduce routing overhead and to improve the packet delivery ratio and throughput. MD5 is a node verification scheme for intrusion detection system. The implementation of MD5 scheme results in the efficient detection of malicious node in the network compared to contemporary approaches, Hashing demonstrates higher malicious behaviour detection rates in certain circumstances while does not greatly affect the network performances.

Key Words: **MANET, Ad hoc, NS-2.**

1. INTRODUCTION

The Mobile Ad hoc NETwork (MANET) is a collection of mobile nodes that is equipped with wireless transmitter and a receiver which is used for communicating with each other through bidirectional wireless links either directly or indirectly. The communication between the mobile nodes is limited by the range of transmitters. So the two mobile nodes cannot communicate with each other when the distance between the two nodes is beyond the range. MANET solves this problem by allowing intermediate nodes to relay data transmissions. The open medium and remote distribution of MANET make it vulnerable to many types of attacks. In particular, considering this fact that most routing protocols in MANETs assume that every node in the network behaves cooperatively with other nodes, attackers can easily attack MANETs by inserting malicious or non cooperative nodes into the network. Further because of MANET's distributed architecture and changing topology, a centralized monitoring technique is no longer feasible in MANETs. In such case, it is crucial to develop an intrusion-detection system (IDS) especially for MANETs.

2. INTRUSION DETECTION SYSTEM

An intrusion detection system is a device or software application that monitors network or system activities for malicious activities or policy violations and produces reports to a management station. Some systems may attempt to stop an intrusion attempt but this is neither required nor expected of a monitoring system. Intrusion detection and prevention systems (IDPS) are primarily focused on identifying possible incidents, logging information about them, and reporting attempts. In addition, organizations use IDPSes for other purposes, such as identifying problems with security policies, documenting existing threats and deterring individuals from violating security policies. IDPSes typically record information related to observed events notify security administrators of important observed events and produce reports. Many IDPSes can also respond to a detected threat by attempting to prevent it from succeeding.

2.1 Passive or Reactive Systems

In a passive system, the intrusion detection system (IDS) sensor detects a potential security breach, logs the information and signals an alert on the console or owner. In a reactive system, also known as an intrusion prevention system (IPS), the IPS auto-responds to the suspicious activity by resetting the connection or by reprogramming the firewall to block network traffic from the suspected malicious source. The term IDPS is commonly used where this can happen automatically or at the command of an operator; systems that both "detect (alert)" and "prevent".

3. WIDS ARCHITECTURE

A wireless IDS can be centralized or decentralized. A centralized wireless IDS is usually a combination of individual sensors which collect and forward all data to a central management system, where the wireless IDS data is stored and processed. Decentralized wireless intrusion detection usually includes one or more devices that perform both the data gathering and processing/reporting functions of the IDS. The decentralized method is best suited for smaller (WAP) due to cost and management issues. The cost of sensors with data processing capability can become prohibitive when many sensors are required. Also, management of multiple processing/reporting sensors can be more time intensive than in a centralized model.

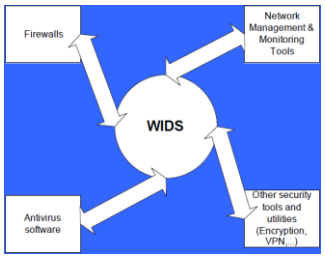

Figure – 1: Wireless Intrusion Detection Systems

3.1 Centralized Wireless IDS

For a centralized style Wireless IDS, there are several wireless 'sensors' that attempt to cover the entire area. The function of these sensors is to gather all wireless data traversing the network, and report it back to a central processing 'analyzer'. This analyzer is the brain of the setup, and carries out the task of scouring the data for patterns of malicious activity, abnormal activity, or activity that conforms to pre-written rules or 'signatures'. Assuming end-users can create their own signatures, or customize existing signatures, it would be possible to accomplish myriad things, like

- Check for unauthorized MAC addresses
- Check for rogue WAPs
- Look for high packet error rates
- Look for signal degradation
- Help triangulate an attacker's physical location
- Warn if a WAP's association table is getting filled up above a user-set threshold
- Check for unencrypted wireless transmission

WIDS signatures can take advantage of the fact that traditional wired Intrusion Detection Systems have an established library of signatures that identify a huge number of attacks. The analyzer will process this incoming data and monitor it for signs of malicious activity or activity that does not conform to operating policy. The analyzer will require substantial processing power in order to efficiently process data from large networks. Some of the essential components of a good analyzer would be a competent correlation engine, an excellent rule set, and common sense.

3.2 Distributed Wireless IDS

In the distributed scheme, there would be several sensors placed around the network, but there would be no central analyzer. Each sensor would be capable of the functions and capabilities of the analyzer described in the previous scheme. Each sensor would keep in touch with the other sensors to exchange information and alerts in order to function as a coherent setup.

4. ANOMALY AND SIGNATURE BASED IDS

Writing own IDS signatures

4.1 Signature Basics

A network IDS signature is a pattern that we want to look for in traffic. In order to give an idea of the variety of signatures, some examples and some of the methods that can be used to identify each one

- Connection attempt from a reserved IP address. This is easily identified by checking the source address field in an IP header.
- Packet with an illegal TCP flag combination. This can be found by comparing the flags set in a TCP header against known good or bad flag combinations.
- Email containing a particular virus. The IDS can compare the subject of each email to the subject associated with the virus-laden email, or it can look for an attachment with a particular name.
- DNS buffer overflow attempt contained in the payload of a query. By parsing the DNS fields and checking the length of each of them, the IDS can identify an attempt to perform a buffer overflow using a DNS field. A different method would be to look for exploit shell code sequences in the payload.
- Denial of service attack on a POP server caused by issuing the same command thousands of times. One signature for this attack would be to keep track of how many times the command is issued and to alert when that number exceeds a certain threshold.
- File access attack on an FTP server by issuing file and directory commands to it without first logging in. A state-tracking signature could be developed which would monitor FTP traffic for a successful login and would alert if certain commands were issued before the user had authenticated properly.

4.2 Functions of Signatures

The obvious is that want to be alerted when an intrusion attempt occurs. But take a moment to think about other reasons why we might want to write or modify a signature. Perhaps seeing some odd traffic on network and want to be alerted the next time it occurs. Noticed that it has unusual header characteristics and want to write a signature that will match this known pattern. Some signatures may tell which specific attack is occurring or what vulnerability the attacker is trying to exploit, while other signatures may just indicate that unusual behavior is occurring, without specifying a particular attack. It will often take significantly more time and resources to identify the tool that's causing malicious activity, but it will gives more information about why attack is made and what the intent of the attack.

4.3 Header Values

Some header values are clearly abnormal, so they make great candidates for signatures. A classic example of this is a TCP packet with the SYN and FIN flags set. This is a violation of RFC, and has been used in many tools in an attempt to circumvent firewalls, routers and intrusion detection systems. Many exploits include header values that purposely violate RFCs, because many operating systems and applications have been written on the assumption that the RFCs would not be violated and don't perform proper error handling of such traffic. Also, many tools either contain coding mistakes or are incomplete, so that crafted packets produced by them contain header values that violate RFCs. Both poorly written tools and various intrusion techniques provide distinguishing characteristics that can be used for signature purposes.

Although illegal header values are certainly a fundamental component of signatures, legal but suspicious header values are at least as important. Alerting on connections to suspicious port may provide a quick way of

identifying Trojan activity. Unfortunately, some normal, benign traffic may happen to use the same port numbers. Without using a more detailed signature that includes other characteristics of the traffic, won't be able to determine the true nature of this traffic. Suspicious but legal values such as a port number are best used in combination with other values.

4.4 Choosing a Signature

A simple signature would be packets with only the SYN and FIN flags set. Although this would certainly be a good indicator of likely malicious activity, it doesn't give us any idea why this activity occurred. Signature development is always tradeoffs between efficiency and accuracy. In many cases, simpler signatures are more prone to false positives than more complex signatures, because simpler signatures are much more general. But more complex signatures may be more prone to false negatives than simpler signatures, because one of the characteristics of a tool or methodology may change over time.

4.5 Anomaly detection

The anomaly detection technique centres on the concept of a baseline for network behaviour. This baseline is a description of accepted network behaviour, which is learned or specified by the network administrators, or both. Events in an anomaly detection engine are caused by any behaviour that fall outside the predefined or accepted model of behaviour.

An integral part of base lining network behaviour is the engine's ability to dissect protocols at all layers. This protocol "dissection" is initially computationally expensive, but it allows the engine to scale as the rule set grows and alert with fewer false positives when variances from the accepted behaviours are detected. Another pitfall of anomaly detection is that malicious activity that falls within normal usage patterns is not detected

However, anomaly detection has an advantage over signature-based engines in that a new attack for which a signature does not exist can be detected if it falls out of the normal traffic patterns.

5. COMPARISON WITH FIREWALLS

Firewalls limit access between networks to prevent intrusion and do not signal an attack from inside the network. An IDS evaluates a suspected intrusion once it has taken place and signals an alarm. An IDS also watches for attacks that originate from within a system. This is traditionally achieved by examining network communications, identifying heuristics and patterns of common computer attacks, and taking action to alert operators. A system that terminates connections is called an intrusion prevention system, and is another form of an application layer firewall.

6. MOBILE AD HOC NETWORK

A mobile ad hoc network (MANET) is a self-configuring infrastructureless network of mobile devices connected by wireless. Each device in a MANET is free to move independently in any direction, and will therefore change its links to other devices frequently. Each must forward traffic unrelated to its own use, and therefore be a router. MANETs are a kind of wireless ad hoc network that usually has a routable networking environment on top of a Link Layer ad hoc network.

6.1 Security of MANETs

A lot of research was done in the past but the most significant contributions were the PGP (Pretty Good Privacy) and the trust based security but none of the protocols made a decent tradeoff between security and performance. In an attempt to enhance security in MANETs many researchers have suggested and implemented new improvements to the protocols and some of them have suggested new protocols.

6.2 Threats to MANETs

- The standard encryption method, Wired Equivalent Privacy (WEP) is weak.
- Hackers can also attack a MANET and gather sensitive data by introducing a rogue WAP into the MANET coverage area.
- The rogue WAP can be configured to look like a legitimate WAP and, since many wireless clients simply connect to the WAP with the best signal strength, users can be "tricked" into inadvertently associating with the rogue WAP.
- Low cost and easy implementation coupled with the flexibility of wireless network communications makes MANETs highly desirable to users.
- By installing a WAP on an established MANET, a user can create a backdoor into the network , subverting all the hard-wired security solutions and leaving the network open to hackers.

6.3 Attacks on MANETs

- Application Layer: Malicious code, Repudiation.
- Transport Layer: Session hijacking, flooding.
- Network Layer: Flooding, Black Hole.
- Data Link/MAC: Malicious Behaviour, Selfish Behaviour.
- Physical: Interference, Traffic Jamming.

7. NS-2 OVERVIEW

NS2 is a discrete event simulator targeted at Networking Research &Educational Institutions. NS2 provides substantial support for simulation of TCP /IP, UDP, Routing, Multi casting protocol over wired and wireless network.NS-2 is an event driven packet level network simulator developed as part of the VINT project (Virtual Internet Test bed). This was a collaboration of many institutes including UC Berkeley, AT&T, XEROX PARC and

ETH. Version 1 of NS was developed in 1995 and with version 2 released in 1996. Version 2 included a scripting language called Object oriented Tcl (OTcl).
It is an open source software package available for both windows and Linux platforms. NS-2 has many and expanding uses including
*To evaluate the performance of existing network protocols.
*To evaluate new network protocols before use.
*To run large scale experiments not possible in real experiments.
* To simulate a variety of IP networks
* Another important tool NAM which is used for Network Animation, X graph tool is used for plotting the NAM log file.
* A discrete event simulator
* Simple model
* focused on modelling network protocols
* Wired, wireless, satellite
* TCP, UDP, multicast, unicast
* Web, Telnet, FTP
* Adhoc routing, sensor networks

7.1 DISCRETE EVENT SIMULATOR

* Model world as events
* Simulator has list of events
* Process: take next one, run it, until done
* Each event happens in an instant of virtual (simulated) time, but takes an Arbitrary amount of real time
* Ns uses simple model: single thread of control => no locking or race Conditions to worry about (very easy)
* NS is an Object-oriented Tcl(Otcl) script interpreter that has a simulation event scheduler and network component object libraries, and network set-up (plumbing) module libraries.
* Object-oriented (C++, OTCL)
* Modular approach
* Fine-grained object composition
* Reusability
* Maintenance
* Performance (speed and memory)
* Careful planning of modularity

8. PROPOSED SYSTEM

For Intrusion detection hash function is used in our project. Message Digest 5 (MD5) algorithm is used for generation of a hash value. This algorithm is used to reduce the routing overhead and to improve the packet delivery ratio and throughput.

8.1 HASHING ALGORITHM

A cryptographic hash function is a hash function that takes an arbitrary block of data and returns a fixed-size bit string, the cryptographic hash value, such that any (accidental or intentional) change to the data will (with very high probability) change the hash value. The data to be encoded are often called the message, and the hash values are sometimes called the message digest or simply digest.

The ideal cryptographic hash function has four main properties

- It is easy to compute the hash value for any given message.
- It is infeasible to generate a message that has a given hash.
- It is infeasible to modify a message without changing the hash.
- It is infeasible to find two different messages with the same hash.

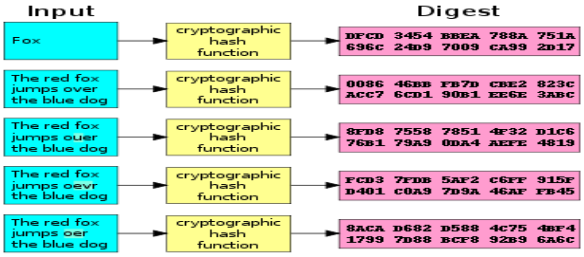

Figure – 2: Cryptographic hash functions

8.2 MD5

MD5 is a node verification technique for intrusion detection system. The implementation of MD5 techniques results in the efficient detection of malicious node from the network. The MD5 technique is an generation of hash id to mobile nodes H(n) = public key/identity. If any node in the network wants to verify neighbour node or any other node , the particular node X request a neighbour node Y to generate a hash id using hash function. The Y node generates hash id using a public key of X and identity of Y. in the same X node also generates a hash id using public key of Y and identity of X. if both the hash id are equal then the nodes are authenticated and not a malicious node. If Y node hash id is not equal to X node hash id then the corresponding Y node is malicious node. Then the detected malicious node is eliminated from the network. In this way MD5 technique detects and eliminate malicious node.

8.3 Advantages

The speed of the cryptographic process can be increased and routing overhead can be decreased. Any node can verify any other at any time, by this intrusion detection system all nodes in the network can be verified and we can form a securable network.

9. SIMULATION

9.1 SIMULATION METHODOLOGY

The performance of MD5 is measured under packet dropping attack. This packet dropping attack makes malicious nodes to drop all the packets that they receive. The purpose of this scenario is to test the performance of IDS.

9.2 SIMULATION CONFIGURATION

The simulation is conducted with the Network Simulator (NS) 2.34 and cygwin. The system is running on a laptop with Core i5 processor and 4-GB RAM. We adopt the

default scenario settings in NS 2.34. The configuration specifies 36 nodes in a flat space with a size of 170×210m. The maximum hops allowed are four. Both physical layer and 802.11 MAC layer are included in the wireless extension of NS2. The moving speed of mobile node is limited to 30m/s. User datagram protocol with constant bit rate with a packet size of 512 B.

9.3 SIMULATION PARAMETERS

Number of nodes	36
Channel type	Wireless
Size of simulation area	170*210
Traffic at application	Cbr
Cbr packet size	512
Traffic time	30s
MAC	Mac/802.11
Routing	AODV/ DSR
Propagation Model	Free space – two ray ground propagation
Maximum packet in queue	100

Table - 1 Simulation Parameter

PERFORMANCE METRICS

In order to measure and compare the performance of MD5 we adopt three performance metrics.

1) Packet delivery ratio.
2) Routing overhead.
3) Throughput.

1) **Packet Deliver Ratio**

The ratio of packets that are successfully delivered to a destination compared to the number of packets that have been sent.

Packet delivery ratio = ∑ Number of packet receive / ∑ Number of packet sent

2) **Routing overhead**

Resource consumed or lost in completing a process that does not contribute directly to the data transmission.

3) Throughput

Throughput or network throughput is the rate of successful message delivery ratio over a communication channel.

Throughput = (Number of Packets Received) / (Number of packets Sent)

9.4 SIMULATION

Figure 3 shows that Nodes used for simulation is generated in the network animator.

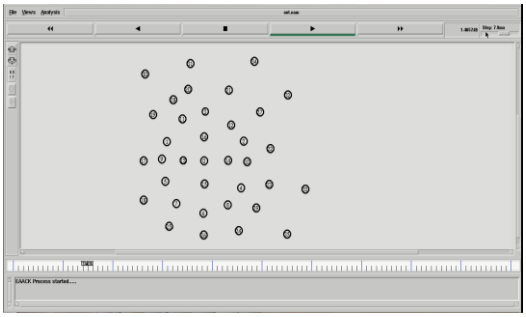

Figure – 3: Generation of nodes

Figure 4 shows the Transmission of data from source to destination through intermediate nodes.

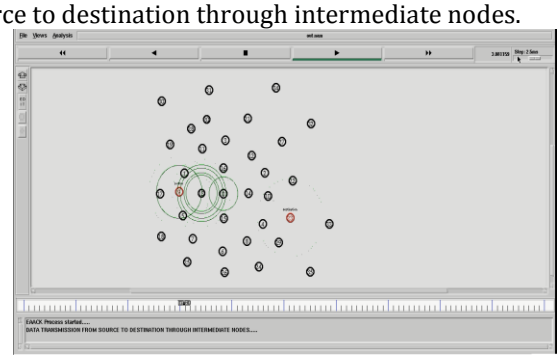

Figure – 4: Data transmission

Figure 5 Shows the Intrusion detection system using hash function is done. Here node 8 requests a neighbour node 4 to generate a hash id using hash function. The node 4 generates hash id using a public key of 8 and identity of 4. In the same 8 node also generates a hash id using public key of

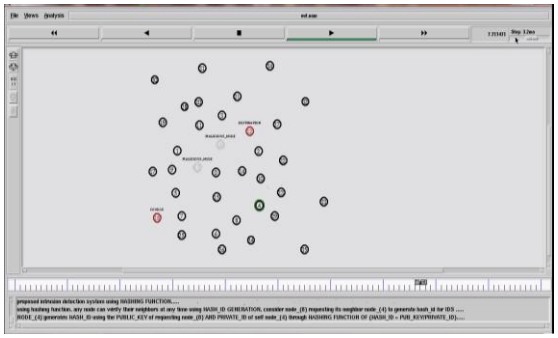

Figure – 5: Hashing function

Figure 6 shows that hash id of node 8 is not equal to hash id of node 4. So node 4 is marked as malicious.

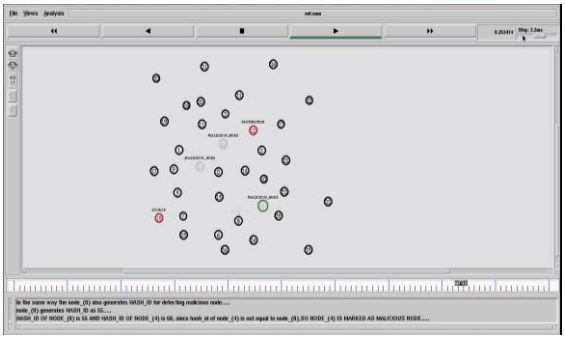

Figure - 6: Detection of malicious node

Figure 7 shows that node 30 requests node 31 to generate hash id for verification. Here the hash id of two nodes is equal so the node 31 is not a malicious node.

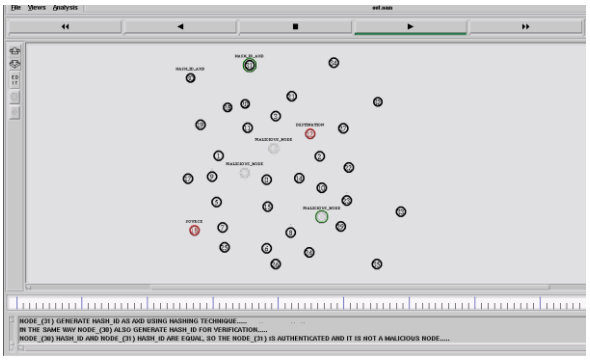

Figure – 7: Detection of authenticated node

10. RESULTS

Figure 8 shows the performance analysis of packet delivery ratio. Here hashing function gives better performance compared to EAACK.

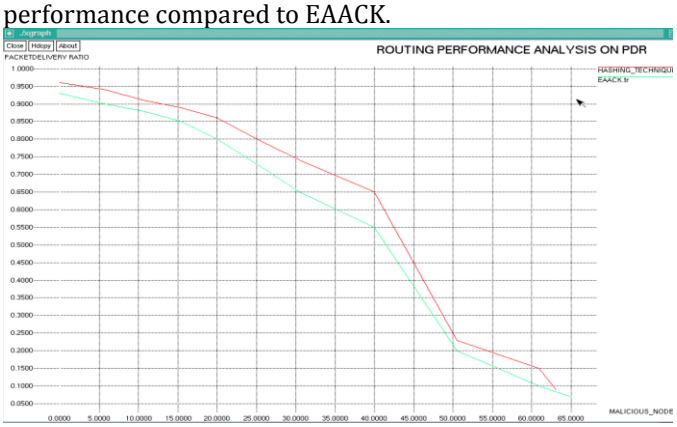

Figure – 8: Packet delivery ratio

Figure 9 shows the performance analysis of throughput here hashing based IDS gives better performance compared to the existing IDS.

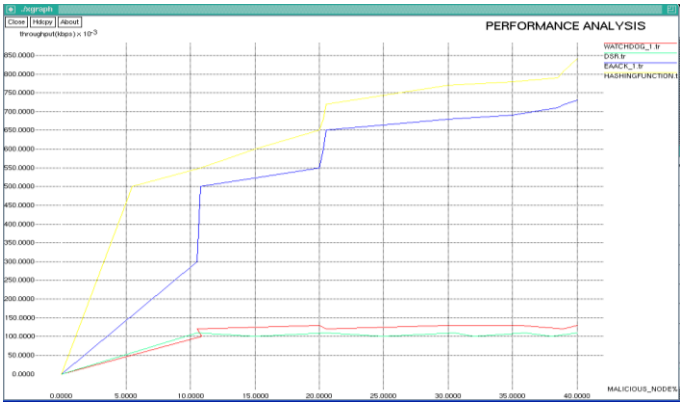

Figure – 9: Throughput

Here the Figure 10 gives the performance analysis of routing overhead for 7 malicious nodes. The hashing produces less routing overhead.

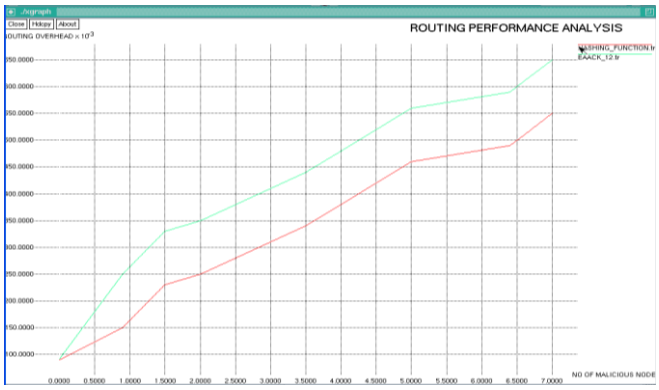

Figure – 10: Routing overhead for 7 malicious nodes

Figure 11 gives the performance analysis of routing overhead for 13 malicious nodes.

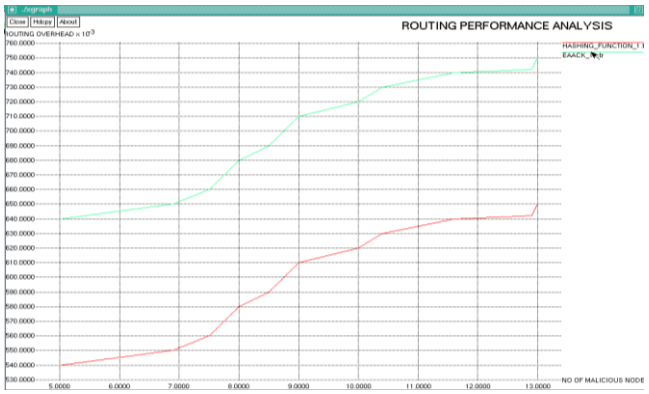

Figure – 11: Routing overhead for 13 malicious nodes

Figure 12 gives the performance analysis of routing overhead for 19 malicious nodes.

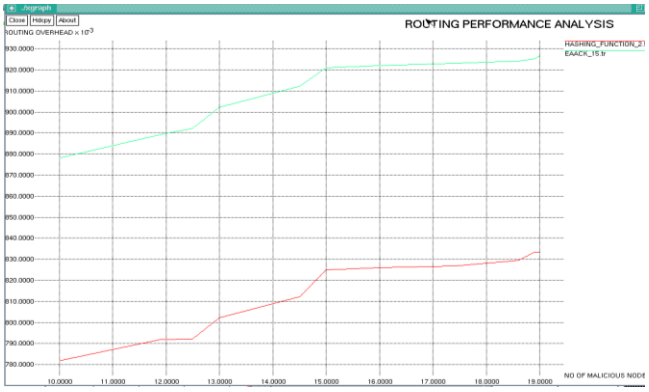

Figure – 12: Routing overhead for 19 malicious nodes

10.1 ANALYSIS

After analyzing all the graphs shown above, it is clear that Hashing performs better than the existing system. The table 3 shows the comparison of results.

Packet Delivery Ratio					
	Malicious nodes 0%	Malicious nodes 10%	Malicious nodes 20%	Malicious nodes 30%	Malicious nodes 40%
EAACK	1	0.88	0.8	0.64	0.55
Hashing	1	0.92	0.86	0.75	0.65
Routing Overhead					
	Malicious nodes 0%	Malicious nodes 10%	Malicious nodes 20%	Malicious nodes 30%	Malicious nodes 40%
EAACK	0	0.44	0.65	0.74	0.92
Hashing	0	0.34	0.55	0.63	0.82
Throughput					
	Malicious nodes 0%	Malicious nodes 10%	Malicious nodes 20%	Malicious nodes 30%	Malicious nodes 40%
EAACK	0	0.29	0.55	0.68	0.73
Hashing	0	0.54	0.65	0.77	0.84

Table 3: Result comparison

11. CONCLUSIONS

In MANETs major threat is packet dropping attack. In this project, Intrusion detection system for reducing packet dropping attacks is designed for MANETs using hash function and compared with other mechanisms. The results obtained gives better performance of hash based schemes than other schemes.

Furthermore in an effect to prevent attackers, a new algorithm can be designed to improve the packet delivery ratio and to decrease routing overhead. In future this hashing algorithm can be implemented in real network environment instead of simulation.

REFERENCES

[1] Akbani R, Korkmaz T, and Raju G. V. S, (2012), "Mobile Ad hoc Network Security," in Lecture Notes in Electrical Engineering, Springer-Verlag, vol. 127, pp. 659–666.

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

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

[4] Dondi D, Bertacchini A, Brunelli D, Larcher L, and Benini L,(2008), "Modeling 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.

[5] Elhadi M. Shakshuki, Nan Kang, and Tarek R. Sheltami,(2013), "EAACK—A Secure Intrusion-Detection System for MANETs" IEEE Trans. Ind. Electron., VOL. 60, no. 3, pp. 1089-1096.

[6] Gungor V. C and Hancke G. P,(2009), "Industrial wireless sensor networks: Challenges, design principles, and technical approach," IEEE Trans. Ind. Electron., vol. 56, no. 10, pp. 4258–4265.

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

[8] Noman Mohammed, Hadi Otrok, Lingyu Wang, Mourad Debbabi, and Prabir Bhattacharya,(2011), "Mechanism Design-Based Secure Leader Election Model for Intrusion Detection in MANET", IEEE transactions on dependable and secure computing. pp. 285–294.

[9] Patwardhan A, Parker J, Joshi A, Iorga M, and Karygiannis T, (2005), "Secure routing and intrusion detection in ad hoc networks," in Proc. 3rd Int. Conf. Pervasive Comput. Commun., pp. 191–199.

[10] Parker J, Undercoffer J, Pinkston J, and Joshi A, (2004), "On intrusion detection and response for mobile ad hoc networks," in Proc. IEEE Int. Conf. Perform., Comput., Commun., pp. 747–752.

[11] Patcha A and Mishra A, (2003), "Collaborative security architecture for black hole attack prevention in mobile ad hoc networks," in Proc. Radio Wireless Conf., pp. 75–78.

[12] Puttini, R, Percher, JM, Camp, O de Sousa R, (2003), "A Modular Architecture for a Distributed IDS for Mobile Ad Hoc Networks". Lecture Notes on Computer Science vol. 2669, Springer-Verlag, pp. 91-113.

Authors

S. Jeniton, M.E.,
Department of Electronics and Communication Engineering,
RAAK College OF Engineering and Technology, Puducherry, India.

K. Thulasi
Department of Electronics and Communication Engineering,
RAAK College OF Engineering and Technology, Puducherry, India.