# Intrusion Detection System Performing A Distributed Novel Hybrid Intrusion Detection Framework

**P.E. Elango, S. Subbaiah**

***Abstract--- Intrusion detection is the one of the major problem in network security, as the use of computer system and network increases, securing data is one of the important in order to achieve secure data transmission without hacking. Intrusions are the activities that violate the security policy of system. The process used to identify intrusions. Today network securities are used in various applications like protect vital information while still allowing access to those who need Trade secrets, Medical records etc. In this paper proposed to a novel hybrid intrusion detection framework methodology and its four various level of phases.***

***Keywords--- WSN, Intrusion, Malicious, Attacks, Hybrid, Storage Area.***

## I. INTRODUCTION

Wireless Sensor Networks have been applied to a range of applications, monitoring of space which includes environmental and habitat monitoring, indoor climate control, surveillance. Monitoring things example can be outlined as structural monitoring, condition-based equipment maintenance.

In addition, monitoring the interactions of things with each other and the surrounding space e.g., emergency response, disaster management, healthcare, energy sector. The majority of these applications may be split into two classifications: data collection and event detection. In various applications of WSNs, the node deployment always draws attention to cover the area of interest. Node deployment strategy is a fundamental issue of a WSN provisioning that is done based on the implementation scenario. The types, number, and locations of devices impact on many intrinsic properties of a WSN, such as coverage, connectivity, cost and lifetime.

Recently, the WSN's technology has widely been used in our daily life.

A typical WSN is shown in Figure 1. In Figure 1 an event is detected in the sensor field and the information is routed to the sinker or base station then to the user with several communication media.

Deployment can normally be categorized as either a dense deployment or a sparse deployment. A dense deployment has a relatively high number of sensor nodes in a given field of interest while a sparse deployment would have fewer nodes in the same field.

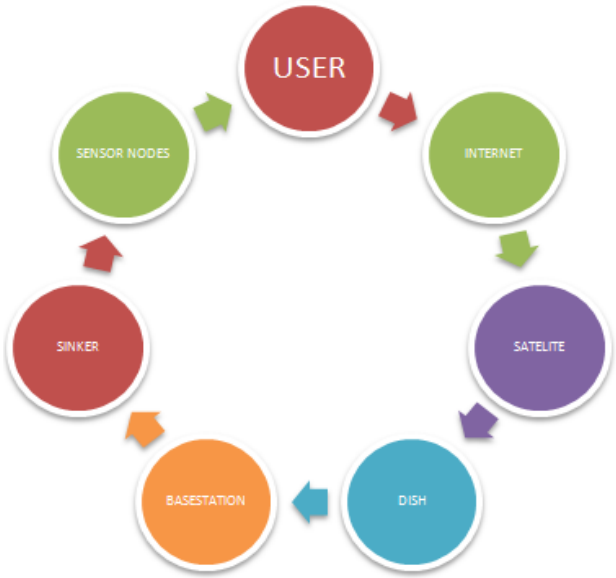


**Figure 1: Wireless Sensor Network Traditional Flow**

Deployment can normally be categorized as either a dense deployment or a sparse deployment. A dense deployment has a relatively high number of sensor nodes in a given field of interest while a sparse deployment would have fewer nodes in the same field. The dense deployment model is usually used in situations where intensive information is needed for every event or when it is important to have multiple sensors cover an area. Sparse deployments may be used when the cost of the sensors make a dense deployment prohibitive or when a WSN needs to achieve maximum coverage using the bare minimum number of sensors. For example, surveillance applications require different degrees of surveillance in different locations, in highly sensitive areas, dense deployment is needed. The limitations of wireless sensor networks are significant factors and must be addressed when designing and implementing a wireless sensor network for a specific application. Therefore, any security mechanism to extract meaningful and actionable information from WSNs becomes a challenge.

## II. METHODOLOGY

### *2.1 A Distributed Novel Hybrid Intrusion Detection Framework*

The agents that are hosted by the nodes are capable of sharing their partial views, agree on the identity of the source and expose it. By distributing the agents throughout the network and have they collaborate, we make the system scalable and adaptive.

**Manuscript received September 16, 2019.**

**P.E. Elango,** Ph.D Research Scholar, PG & Research Dept of Computer Science, Periyar University, Salem, T.N, India.

**S. Subbaiah,** Assistant Professor, Department of Computer Applications, Vivekananda College of Arts & Science (Autonomous), T.N, India.

When a malicious node is found, an alarm message is broadcasted to the network. Each node then makes a final decision based on the detection reports from other nodes. To avoid drastic flooding over the network caused by broadcasting local detection results, the alarm messages are restricted to a region formed only by the alerted nodes.

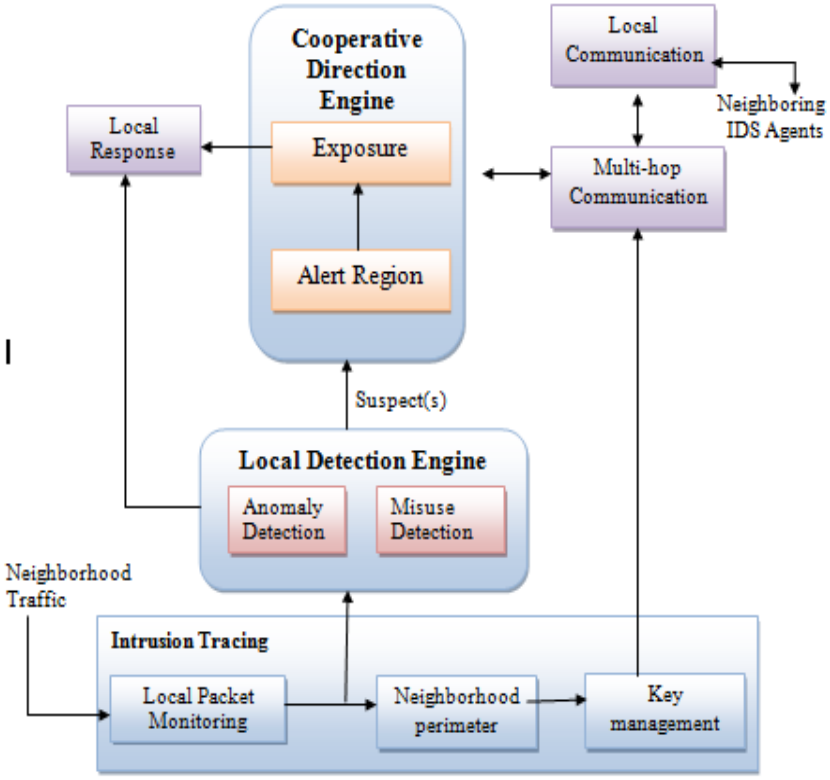


**Figure 2: Architecture of LIDeA IDS agent**

We build the architecture of the IDS agent based on the conceptual modules shown in Figure 2. Each module is responsible for a specific function, which we describe in the sections below. The IDS agents are identical in each node and they can broadcast messages for agents residing in neighboring nodes.

## III. PROPOSED WORK

### *3.1 Phase I- Malicious Code Injection Attacks On Software's*

Sensor networks hold the potential to significantly transform the way that computing affects life. In order to reach this potential, however, security must be achieved. This chapter in the thesis demonstrates how vulnerable, in terms of data confidentiality and network availability, sensor networks are. The best way to do that is to look into new threat models, how specific attacks can be realized in practice and study new methods from the attacker's point of view. This will set the scene for the development of the first instances of sophisticated attack tools capable of launching various kinds of attacks for compromising the network's functionality. We believe that by studying the after-effects of various exploits on the network itself. Moving on to the darker side, we will explore a new set of memory related vulnerabilities for sensor embedded devices that, if exploited, can lead to the execution of software-based attack. Software attacks are concerned with modifying the running code on a sensor platform or even injecting malicious one (malware). Malware is defined as a software designed to execute attacks on software systems and fulfill the harmful intents of an attacker. A well known example of this type of attack is the buffer overflow attack.We achieve this by exploiting a buffer overflow vulnerability to smash the call stack and intrude a remote node over the radio channel. By breaking the malware into multiple packets, the attacker can inject arbitrarily long malicious code to the node and completely take control of it. Then we proceed to show how the malware can be crafted to become a self-replicating worm that broadcasts itself and infects the network in a hop-by-hop manner. While this attack is extremely dangerous, there has been very little research in this area. To the best of our knowledge, this is the first instance of a self-propagating worm that provides a detailed analysis along with instructions in order to execute arbitrary malicious code. We expect that our work will be particularly useful in sensor network research for showing the destructive impacts of a sensor worm and highlighting the need to come up with efficient mechanisms to counter such attacks.

### *3.2 Phase II- Stealthy Exploits Attacks On Wireless Sensor Networks*

Spy-Sense is based on an intelligent component-based system. The hosted components are capable of loading predefined exploit profiles, injecting them to the targeted network through a transparent transmission of a series of specially crafted messages, receiving and logging of all node replies that report back requested system information.

Its core functionality is based on four main conceptual modules. One of the key design goals of Spy-Sense is its wide applicability; it supports exploit injection attacks and compromise of a wide variety of sensor hardware and network protocols. It can exploit tall vulnerabilities and weaknesses arising from a specific platform despite the followed memory architecture (Von Neumann and Harvard) since subsequent code injection can be performed in either of them. Furthermore, while capturing and logging of all node replies is performed in real time, content analysis can be done either online or o²ine. We believe that o²ine analysis provides a better way of extracting information regarding network activities and information patterns. In what follows we give a more detailed description of the four basic system components. The exploit loader is responsible for initializing the software by importing all predefined exploit profiles that reside in the Spy-Sense root folder. Such profiles contain (i) the machine code instructions that will be injected into the host sensor node, and (ii) their symbolic representation written in assembly language. Exploit loading and registration can occur anytime during Spy-Sense operation; either upon system boot up or during normal operation by updating the contents of the corresponding storage folder. All exploit code instructions are contained in files and are loaded one at a time. This is the most convenient and platform-independent way for a user to define his/her own exploit profiles that need to be imported in Spy-Sense. Again, new additions can either be performed at boot up time or during system operation.

### *3.3 Phase III- Intrusion Detection and Tolerance of Attacks in Storage Area Networks and Wireless Environments*

As defined in, SAN systems ar composed of SDs, specialized networks for connecting knowledge SDs to servers and therefore the needed management layer for putting in place and maintaining connections between these servers and SDs. In such design, there are essentially 3 element classes: the servers, the SVC and SDs. The SVC is that the major element that manages the server and SDs interactions. It virtualizes SDs into a standard pool and allocates storage to host systems from that common pool. Servers are connected to SAN so as to serve shopper requests and to access knowledge that are situated at the SDs. Analyzing an interaction state of affairs between the Different system entities (client, server, SVC and SDs), the SVC ought to maintain a set of structures in order to serve incoming requests. There are primarily 3 structures that ought to be on the market and managed at every SVC: 1) a mapping table that holds for every entry the dealings and its originator; mapping table that holds for every entry the server and its associated SDs, that are logically hooked up to it; and 3) a structure that ensures the mapping of every high level resource.

### *3.4 Phase IV- TLS and DTLS*

We offer the mandatory background data and necessity material that area unit required to ascertain an understanding of the TLS and DTLS protocols. First, we offer background data concerning the TCP/IP protocol suite and de-scribe 3 elementary networking protocols: scientific discipline, communications protocol and UDP. Second, we have a tendency to introduce Transport Layer Security (TLS), describe however the TLS protocol is structured and discuss its modes of operation. We have a tendency to conjointly introduce Datagram Transport Layer Security (DTLS) and describe the differences between TLS and DTLS. We have a tendency to then gift thoroughly the construct of oracles and show however an attack are often in theory mounted against TLS employing oracle. Finally, we have a tendency to gift variety of attacks against the TLS and DTLS protocols, serving as a forerunner to our attacks on DTLS and TLS.

## IV. EXPERIMENTAL RESULTS

*Phase I*

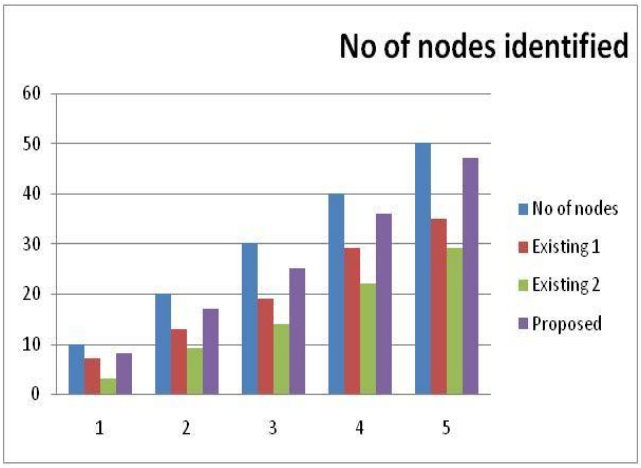


**Figure 3: Nodes Identification**

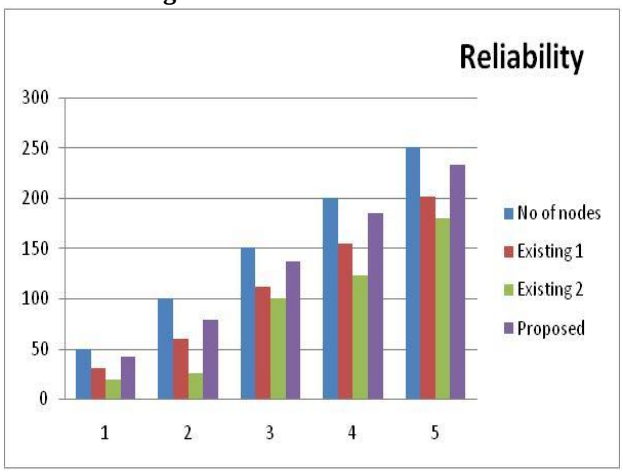


**Figure 4: Reliability**

Even though research in worms against several types of networks has increased significantly over the last years, existing literature in sensor networks is quite limited. However, as we described in the previous sections, their effects on the network itself can be destructive. Indeed, as we will demonstrate an adversary can use malicious code injection techniques for injecting various spyware exploits in the sensor nodes; another severe threat that is often overlooked in the design of secure sensor network applications.

Figure 3 is represented into no of nodes identified values in graphs. External attacks find the existing values are low but their Malicious code injection values are detect amongnodes in the external attacks. Figure 4 is represented into reliability values in graphs. External attacks find the existing values are low but their Malicious code injection values are detect amongnodes in the external attacks.

*Phase II*

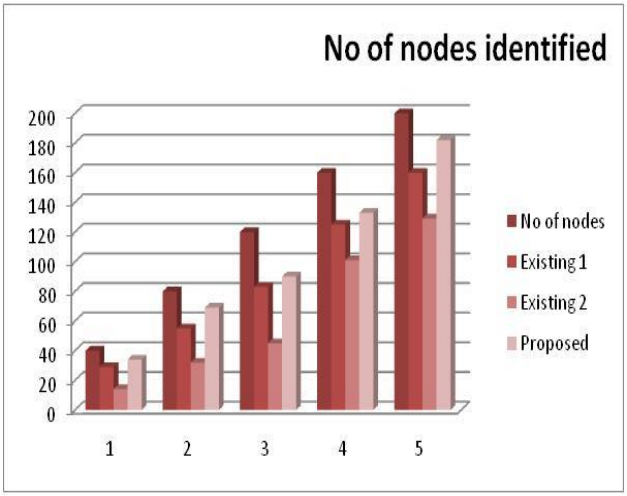


**Figure 5: Nodes Identification**

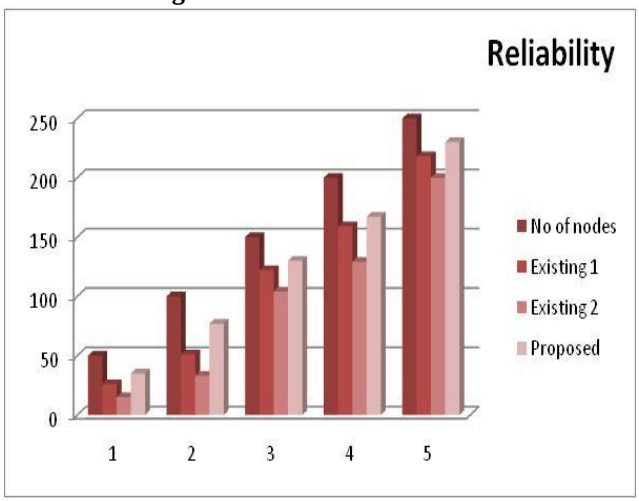


**Figure 6: Reliability**

Spy-Sense exploits will reside in a continuous memory region in the host sensor platform. They can operate in stealth mode as they are programmed to change and restore the flow of the system's control in such a way so that they don't let the underlying micro-controller go into an unstable state. These exploits make use of the existence of an empty, unused and unchecked memory region reserved to be used as the heap for dynamic memory allocation. This works as an umbrella of all the exploits masquerading their existence and reliably evading detection. Furthermore, it results in a permanent exploit injection; the micro-controller's main logic does not perform any actions on the heap region, and thus, the only way of erasing heap contents is by physically capturing a node and forcing it to hard reset itself. Figure 5 is represented into no of nodes identified values in graphs. External attacks find the existing values are high but their SPY-Sense values are detect the lower than among the nodes in the external attacks. Figure 6 is represented into reliability values in graphs. External attacks find the existing values are high but their SPY-Sense values are detect the lower than among the nodes in the external attacks.

*Phase III*

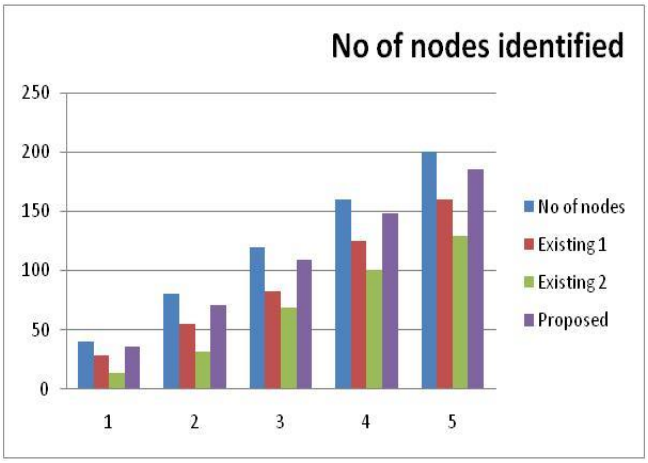


**Figure 7: Nodes Identification**

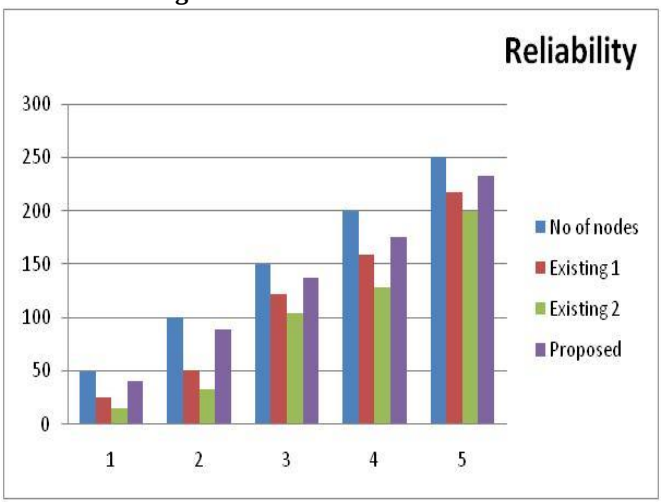


**Figure 8: Reliability**

Storage area Network (SAN) systems has enhanced over the globe because of the necessity of distributed storage and therefore the nice volumes of information handled by business applications. In such frameworks, all hosts hook up with storage through a network. There's additional security risk than ancient storage system. on the market intrusion detection systems don't apply efficiently to SAN environments because of the utilization of static rules and therefore the lack of cooperation between detection modules. Moreover, the detection elements could also be compromised if the interloper gains access to the system. Moreover, detection is performed for the foremost planned solutions at the system and network levels. to beat these limitations, some works have studied the practicability of police investigation attacks at the disk level and providing protection against intrusions. Moreover, the cooperation of the detection elements situated at an equivalent host or network is among solutions that are adopted to boost detection capabilities. During this context, we have a tendency to propose AN efficient intrusion detection and tolerance answer for SANs environments.

Detection capabilities are protected against interloper activities since they're performed by compromise freelance elements. Knowledge command by this technique is very protected by dividing the disk into 2 spaces and granting the management of the protected area to solely the disk aspect elements. Detection is increased by cooperating 3 levels of collected knowledge (network, host and storage levels) and dynamically change detection rules all told the SAN system. Additionally, the planned system tolerates attacks so as to gather information about malicious activity for post-mortem investigation. Figure 7 is represented into no of nodes identified values in graphs. External attacks find the existing values are low but their Storage Area Network IDS values are detect the among the nodes in the external attacks. Figure 8 is represented into reliability values in graphs. External attacks find the existing values are low but their Storage Area Network IDS values are detect the among the nodes in the external attacks.

*Phase IV*

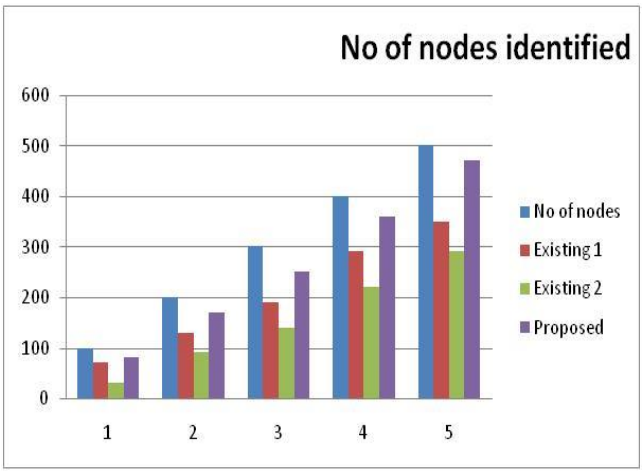


**Figure 9: Nodes Identification**

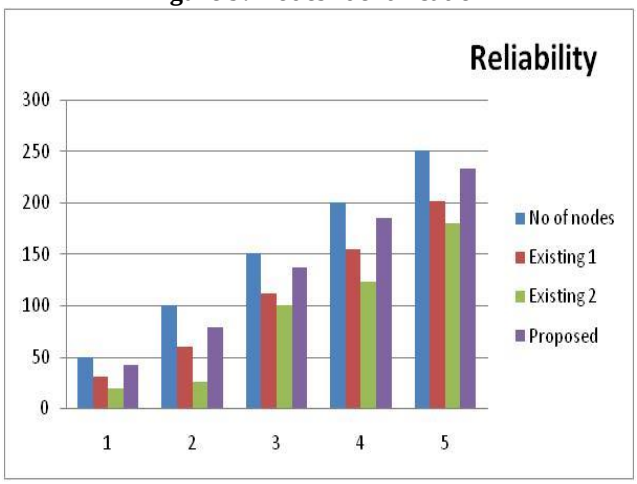


**Figure 10: Reliability**

We offer the mandatory background data and necessity material that area unit required to certain an understanding of the TLS and DTLS protocols. First, we offer background data concerning the TCP/IP protocol suite and de-scribe 3 elementary networking protocols: scientific discipline, communications protocol and UDP. Second, we have a tendency to introduce Transport Layer Security (TLS), describe however the TLS protocol is structured and discuss its modes of operation. We have a tendency to conjointly introduce Datagram Transport Layer Security (DTLS) and describe the differences between TLS and DTLS. we have a tendency to then gift thoroughly the construct of oracles and show however an attack are often in theory mounted against TLS employing fact oracle. Finally, we have a tendency to gift variety of attacks against the TLS and DTLS protocols, serving as a forerunner to our attacks on DTLS and TLS. Figure 9 is represented into no of nodes identified in external attack values in graphs. External attacks find the existing values are low but their TLS/DTLS values are detect the among the nodes in the external attacks. Figure 10 is represented into reliability in external attack values in graphs. External attacks find the existing values are low but their TLS/DTLS values are detect the among the nodes in the external attacks.

## V. CONCLUSION

The attack can be used to add malicious functionalities to sensor nodes or simply shut down the entire network. The research have demonstrated the disastrous effects of such malware to the host network by building Spy-Sense, the first instance of a spyware tool capable of compromising a sensor network's confidentiality and functionality. Spy-Sense is undetectable, hard to recognize and get rid of, and once activated; it runs in a discrete background operation without interfering or disrupting normal network operation. The TLS and DTLS protocols are flexible by design; new protocol extensions can be easily introduced some-times in an ad hoc fashion. Although this flexibility is an advantage, it could potentially introduce an amount of confusion to implementer who need to be familiar with the different versions of TLS and their various extensions in order to implement and maintain the protocols properly, taking into consideration that all versions of (D)TLS are already in deployment. The detection of attacks against SAN systems requires the cooperation of different components located at the SVC and SDs connected to SAN.

## REFERENCES

1. Ashok Kumar, D., and Venugopalan, S.R., 2016, December. A Novel algorithm for Network Anomaly Detection using Adaptive Machine Learning.In Advanced Computing and Intelligent Technologies (ICACIE), 2016 First International Conference on. Springer
2. Singh, S.P. (2010) Data Clustering Using K-Mean Algorithm For Network Intrusion Detection, Thesis, Lovely Professional University, Jalandhar.
3. Deepthy K. Denatious, and John, A. (2012) 'Survey on data mining techniques to enhance intrusion detection', International Conference on Computer Communication and Informatics, ICCI-2012, Coimbatore, India.
4. Sunil Gupta, Authentication Framework against "Malicious Attack in Mobile Wireless Sensor Networks", Vol II, IMECS 2017, March 15 - 17, 2017
5. Chaudhari H.C. and KadamL.U, "Wireless Sensor Networks: Security, Attacks and Challenges International Journal of Networking", Volume 1, Issue 1, 2011, pp-04-16.

7. Hu, Perrig, and Johnson, "Malicious Node Detection in Wireless Sensor Networks" Waldir Ribeiro Pires J´unior Thiago H. de Paula Figueiredo Hao Chi Wong Antonio A.F. Loureiro
8. Deepmala Verma, Gajendra Singh, Kailash Patidar, Detection of Vampire Attack in Wireless Sensor Networks , Vol. 6 (4) , 2015, 3313-3317
9. Dr. AdwanYasin ,KefayaSabaneh ,"Enhancing Wireless Sensor Network Security using Artificial Neural Network based Trust Model" , Vol. 7, No. 9, 2016
10. H. Gorine, M. Ramadan Elmezughi, "Security Threats on Wireless Sensor Network Protocols," 18-19 August 2016
11. Soram Rakesh Singh, Narendra Babu C R, Improving the "Performance of Energy Attack Detection in Wireless Sensor Networks by Secure forward mechanism", Volume 4, Issue 7, July 2014
12. Andriy Stetsko, Lukas Folkman, Vashek Matyas, Neighbor-based" Intrusion Detection for Wireless Sensor Networks", 6th International Conference on Wireless and Mobile Communications (ICWMC), 2010, pp. 420-425
13. W. Wang and B. Bhargava, Visualization of wormholes in sensor networksin WiSe 2004: Proceedings of the 3rd ACM Workshop on Wireless Security,pp. 51-60, ACM New York, NY, USA, Oct. 2004.
14. L. Qian, N. Song, and X. Li, Detection of wormhole attacks in multi-pathrouted wireless ad hoc networks: a statistical analysis approach Journal OfNetwork and Computer Applications, vol. 30, pp. 308{330, Jan. 2007.
15. M. A. Gorlatova, M. Kelly, R. Liscano, and P. C. Mason, Enhancingfrequency-based wormhole attack detection with novel jitter waveforms in IEEE Secure comm: Third International Conference on Security and Work-shops, pp. 304{309, Sept. 2007.
16. P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Vien-not, Optimized link state routing protocol for ad hoc networks in INMIC 2001: IEEE International Multi Topic Conference - Technology for the 21st Century, pp. 62{68, Dec. 2001.
17. J. Daemen and V. Rijmen, The design of Rijndael: AES-the advanced encryption standard. Springer Verlag, 2002.