International seminar on Nanoscience and Technology- conference proceedings. Sept- 2016
Volume 3, Special Issue - Available Online: http://ijrset.in/Volume3,Special-Issue-II.php Pages: 51-61
ISSN 2394-739X
International Journal for Research in Science Engineering and Technology

INTRUSION DETECTION AND TOLERANCE OF ATTACKS IN STORAGE AREA NETWORKS AND WIRELESS ENVIRONMENTS

[1] P.E. Elango, [2] S. Subbaiah
[1] Ph.D Research Scholar, [2] Assistant Professor,
[1] PG & Research Dept of Computer Science, [2] Department of Computer Applications,
[1] Periyar University, [2] Vivekananda College of Arts & Science (Autonomous),
[1, 2] Salem, T.N, India.

ABSTRACT - Recently, the readying of 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. The detection of attacks against SAN systems requires the cooperation of different components located at the SVC and SDs connected to SAN. In this paper, we have presented a novel intrusion detection and tolerance system for SANs. This solution ensures dynamic intrusion detection based on a set of components and structures that are introduced in the SAN system at the SVC and each SD. The distribution of the SVC still among the limits of the proposed scheme since it is designed for a centralized SVC. We have presented an intrusion detection and tolerance scheme that may be integrated or added to traditional intrusion detection and tolerance systems in order to detect attacks related to traffic exchanged between transaction-based applications running in a wireless environment.

Keywords: [Storage area Network, Intrusion detection, Tolerance, Wireless.]

1. INTRODUCTION

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 to lerance answer for SANs environments. 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 postmortem investigation. In this section, we have a tendency to introduce the SAN thought and therefore the associated options that thought about once planning a security answer supported SAN systems. SAN means that to produce a network of storage resources to servers so as to extend storage utilization and flexibility whereas reducing total prices. This answer permits the separation of storage from individual platforms and at the same time the knowledge transfer among all nodes on the storage network. Because of recent technology advances regarding equipment's that are required to interconnect storage device (SDs) SANs don't seem to be solely deployed by massive enterprises however there are used in small-to-medium systems.

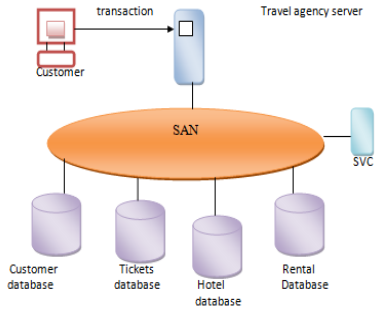

Figure 1: A SAN system for travel services.

A SAN may be a topology with 3 distinct features:
1) Storage isn't directly connected to network clients;
2) Storage isn't directly connected to the servers; and
3) SDs is interconnected.
SANs have found a good relevancy in these days business applications since they enable to share huge databases that are higher being placed on AN external Mount Rushmore State than on a server. The block virtualization approach is employed to handle a number of the complexities concerned in managing SAN systems by aggregating the storage into a common pool. Storage units are appointed to host systems (i.e., servers) from this common pool. These units are referred as logical disks.
As defined in, SAN systems are 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. Figure 2 shows the design of a SAN system. 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 (e.g. file,directory) to its physical allocation at the SD (e.g.FileallocationTable).

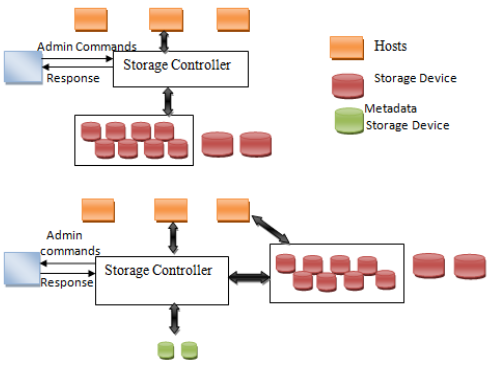

Figure 1: SAN system types

We can notice from these wants that a SVC is a crucial element in a very SAN system design which it oughtto have agood p erformance capability and additionally it oug ht to be extremely protected against malicious makes an attempt. The first want is fulfilled by the different on the market implementations like the IBM SVC that's engineered on a cluster of Pentium-based servers on the opposite hand security problems stay AN investigated topic seeking for solutions.

2. AN APPROACH TO INTRUSION DETECTION AND TOLERANCE FOR SANS

This section details the proposed intrusion detection and tolerance approach for SAN environments by giving the processing performed by SAN components and the decision made and the dynamic update process of available rules. In order to ensure efficient detection and tolerance, a secure communication scheme is proposed and detailed in this section in order to protect information exchanged between SAN components.

Dynamic scheme for detection and tolerance

The detection according to the proposed scheme is performed using compromise independent components that are located at the disk level. Detection is performed based on a set of rules that are generated initially then updated by cooperating detection components located at the SVC level (network, host and disk). In addition, the detection is performed in second step by the SD that processes the incoming request then attaches its decision to data sent back by the RH to the SCI, at the SVC side. The operation (*) models the fact that the message content is encrypted using the sender's private key. The negative value for a decision or means that the request is malicious. In addition, the operation <means in this context that the decision located at the left of the operator is negative (malicious). In addition to the processing of incoming requests by the SVC or SDs, the disk-side components perform a set of additional tasks in order to update the set of available dynamic and static rules. When the SVC receives the SCI forwards it to the VRM that decrypts it. If (<) then the RG updates the set of rules that are stored in the protected area. The detection scheme is illustrated on the basis of an elementary request that belongs to a task of the transaction to be processed. The SVC is responsible of the management of the overall tasks processing. The monitoring of requests and tasks processing continues until the end of the transaction treatment. The tolerance strategy, in both the SVC and SDs, is based on a set of rules that are initially defined in and the redirection file that defines the mapping between the disk blocks that are located in both the virtual contaminated and healthy areas. It is noteworthy that the tolerance strategy is not static since it is updated when the disk changes or new rules are added by the RG at both the SVC and SDs.

Secure communication scheme

Information exchanged between the SAN system components need to be secured in order to prevent unauthorized alteration of their content. In order to protect exchanged information, a certificate-based scheme is adopted where SDs and the SVC have their own digital certificates. Each entity (SVC, SDs) sending a message (e.g. an alert) through the network interface encrypts the message content using its private key. Entity receiving this encrypted message uses the sender's public key in order to decrypt it. Let denotes as the message (request, notification), j the encrypted message sent by and associated to, as the x's digital certificate, the x's private key and the x's public key associated to and extracted from. Message encryption and decryption are performed respectively as following: j= (,) and = (j ,). Both encryption and decryption are performed at

the disk level and certificates are stored in a protected area. At the implementation phase, this protection scheme introduces an additional delay that may be optimized by encrypting the hash instead of the message. In this case, a HMAC digest may be a solution replacing the encryption of the message.

3. PROCESSING REQUIREMENTS IN WIRELESS ENVIRONMENTS

The processing of transactions in a wireless environment is different compared to a wired one. Therefore, a set of structures and processing capabilities are needed to manage them in such environments and to render intrusion detection and tolerance possible, as described in. In this section, we give a description of the adopted communication model and the processing performed by the scheduler.

The transaction concept

Before describing the processing capabilities, a definition of the transaction concept is required and given in. A transaction represents a finite sequence of operations on database objects. The execution of a transaction is finalized with a commit or abort operation to indicate if the execution was successful or not, respectively. Transactions generally consist of a sequence of read and write operations. If several transactions concurrently read from and write to a database, the following problems detailed in, can arise: 1) Lost Update Problem; 2) Dirty Read; and 3) Phantom Updates. To avoid these problems transactions describe atomic operations. The provision of an atomic operation means that the effect of performing any operation on behalf of one client is free from interference with operations being performed on behalf of other concurrent clients; and either an operation must be completed successfully or it must have no effect at all. A mobile transaction is a set of relatively independent (component) transactions which can interleave in any way with other mobile transaction. A transaction is composed of a set of sub transactions that may be represented as follows: = 1, 2, ..,where is the number of sub transactions.

Communication model

Wireless nodes that participate in the processing of a transaction and their associated sub transactions exchange a set of messages with the central node, called the transaction scheduler. These messages are needed by the central node in order to collect sufficient information about the processing status related to the monitored transactions. In this way, a strategy of notification is applied by each node participant in the execution of a transaction. A node sends a notification if one of the following events occur: The transaction processing status has changed; the processing node faces a lack of resources (battery, memory, free disk space, etc.) that render the transaction processing impossible; A malicious activity is detected by the local detection components; a timeout expired before receiving from other nodes information needed as input for the processing of transactions ; and the processing node is not accessible by the remaining participant nodes or by the scheduler. Exchanged messages between processing nodes are protected by adopting a sym- metric encryption scheme that make use of a shared key that is distributed by the scheduler to all participant nodes in a secure manner.

Scheduler Processing

The scheduler is the main component that is used in a transaction-based environment. In addition to the set of tasks performed by this component in order to manage the transaction processing, to implement the following functions, which are required by the intrusion detection scheme. Generation and distribution of encryption keys: for each new incoming transaction, the scheduler determines the wireless nodes that are

required to process sub transactions. After identifying these nodes, the scheduler initiates the generation of the key used during the notification phase. After generating this key, the scheduler sends it in an encrypted message using the public key extracted from each participant node certificate. Another schememay be used with nodes that have no certificates. In this case, the Diffie-Hellman key exchange protocol is used in order to share a secret key with each participant node to be used for notifications exchange.

Monitoring the processing of sub-transactions: the scheduler monitors the processing of sub transactions by updating the transactions' status table. The update process is performed based on the notifications received from the participant nodes. If notifications are not sent by some participant nodes, the scheduler checks the availability of these nodes and sends a status request to each node. The scheduler waits a period of time, for a response from such nodes. If notifications received, it updates the adequate entries in such table else it checks if there is another rnode that is able to process these subtransactions. Selection and connection negotiation with a new scheduler: when the current scheduler is unable to perform the management of transactions and their monitoring due to a lack of resources, it starts the selection phase of a new scheduler from available nodes then negotiates with the selected node a connection establishment in order to transfer available structures needed to monitor and manage running transactions. Such exchange is performed after establishing an encrypted tunnel between the two nodes. Managing commit and rollback operations: during the sub transactions processing a local commit or rollback may be performed. The decision about these two operations is made by the scheduler that orders one of the two operations based on the global transaction processing status and the detected alerts. In this case, one of the two messages is addressed by the scheduler to the participant node in order to perform a local commit or rollback. The participant nodes end a response to this request when the operation is terminated.

If a global commit is accepted by the scheduler, a request is addressed by it to all participant nodes in order to perform the set of local commits. The requests are addressed in an order that is coherent with the dependency relationship between participant nodes concerning the processing of the requested transaction and their sub transactions.

4. INTRUSION DETECTION AND TOLERANCE IN WSAN'S

Many intrusion detection and tolerance solutions have been proposed last years for wired and wireless networks. These techniques have shown their efficiency against several kinds of attacks but they are unable to detect attacks that make use of transaction-based traffic exchanged in a wireless environment since they limit their processing only to the requested operations without considering the set of constraints related to a wireless node such as the lack of resources, the unavailability of a node for a lap of time, etc. Some particularities related to the transaction-based traffic may be added to these constraints and that may be exploited by an intruder such as rollback and commit operations, in order to perform its attack by introducing damage to nodes participating in the running of transactions. For example, an intruder may define an attack scenario including a transaction that is composed of sub transactions. The execution of the malicious sub transactions may be delayed by the intruder by rendering the mobile nodes unavailable temporarily in order to defeat available intrusion detection systems. Moreover, the node performing detection maybe unavailable or compromised therefore the detection of attacks is impossible. A fake commit or rollback may be issued by the intruder when the scheduler is compromised. As a consequence to these malicious actions, critical data may be lost. To overcome such shortcomings, we propose

in the following a storage-based intrusion detection and tolerance scheme for transaction-based applications in wireless environment. A set of modules that are compromise independent are defined in order to detect and tolerate attacks for such environments in addition to the generation of The rest of this section is organized as follows. First, we present the system processing in wireless environments by detailing the proposed communication model and the processing performed by the scheduler. Next, we describe the proposed intrusion detection scheme for wireless environments. The following sub-section details the proposed scheme to tolerate transactions in wireless environments based on information collected from running transactionsand detection rules. The detection and tolerance capabilities of the proposed system are illustrated through a case study.

The proposed structures

The proposed intrusion detection and tolerance system performs detection based on a set of information collected about running transactions. To fulfill this need the following set of structures are defined and stored in a protected area as described in.

Transactions' status table: This table holds for each transaction information about their sub transactions. It is only located at the scheduler side. Each entry of this table includes the following fields: This table holds only information about transactions that are in processing. The decision n field may have one of the following two values: legitimate, malicious. Entries related to a transaction which its processing has been finished are retrieved from this table after a time period, called fixed by the scheduler at the initialization phase.

Alerts file: alerts generated by the violation rules monitor for the transaction scheduleror participant nodes are stored in the alerts file. Each entry in this file includes the following set of information: transactionid, sub transactionid, violation details. This file has a XML format that is selected since it is adequate to represent the parent/child relationship between a transaction and their sub transactions. The scheduler collects alerts from participant nodes only if there is atleast a subtransaction that is malicious.

Intrusion detection rules: these rules are hold in a database located in a protected area. These rules have a format that is different compared to known detection rules that include generally a set of fields that are related to the performed elementary action on the system.

detection rules suitable for the transactions structure and processing in wireless environments that better enabling detection of attacks targeting transactions and attempting to delay the processing or replaying some subtransactions.

The proposed architecture

In order to perform intrusion detection and tolerance in a transaction-based wireless environment, a set of modules are introduced, as illustrated by Figure 3.

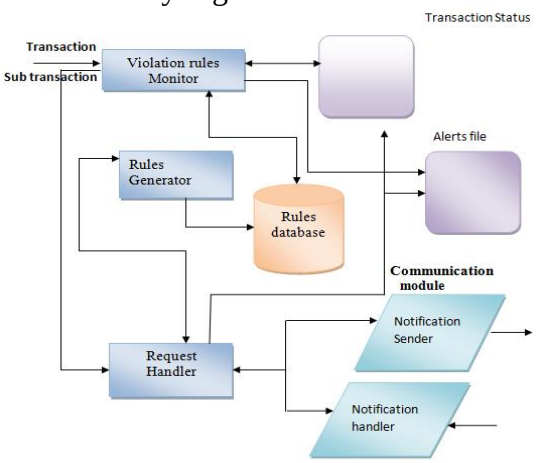

Figure 3: Intrusion detection and tolerance architecture.

The features of the proposed scheme are implemented by the following set of components: Violation rules monitor (VRM): it checks incoming traffic against available detection rules. If a rule matches the monitored traffic, an alert is generated and then stored in the alert file. The detection at the scheduler and participant nodes is performed in the same way button different

detection rules and structures. In addition, this module checks the transactions' status table in order to verify if there are notifications received from the participant nodes and that signal a new detected violation related to processed subtransactions. When a violation is detected, the VRM orders the Request handler to initiate the intrusion tolerance process and to inform the participant nodes. At each participant node, the VRM performs intrusion detection based on a set of detection rules that are related to elementary operations on available resources for each node and based on notifications sent by the scheduler.

Rules generator module: based on a previous work described in, the rule database is composed of both dynamic and static rules. Static rules are generated initially for each wireless node. However, dynamic rules are generated when a new malicious activity is detected. For transaction-based applications, the generated rules has a format that is different to traditional intrusion detection systems such as snort rules for example since it is needed to include in each rule additional information that represent the relation between a transaction and their subtransactions. Moreover, this module updates detection rules based on the notifications received from the scheduler or the participant nodes depending on the node where this module is implemented (scheduler or participant node).

Request handler (RH): this component has been defined previously in. It handles incoming transactions when the wireless node is a scheduler. In this case, it determines nodes that are needed to process sub transactions and address each sub transaction to the adequate participant node. For the remaining nodes, this module handles incoming sub transactions by checking disk space management structures in order to access the requested resources needed to process there quested actions on them.

Rule Generation Process

The proposed scheme generates rules in a manner that takes into consideration the transaction particularities. The proposed format reduces considerably the number of rules and also optimizes processing for detection components. Rules are generated by following a set of steps that starts by analyzing the processed transaction in order to generate a dependency graph. This graph includes all sub transactions and the relationship between them. Nodes in this graph are the sub transactions that constitute the transaction. Links between nodes represent input/output dependency between nodes. This graph is represented using two operators that are: +, =. The former defines the relation between sub transactions. The latter illustrates the dependency input/output relationship between sub transactions. Figure 4 illustrates the dependency graph and the generated rules for a sample transaction.

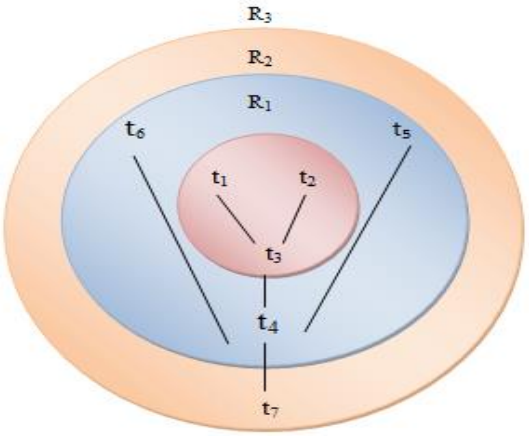

Figure 4: Transaction's dependency graph.

An example illustrating this case is the following: 1) 1:performing a SQL injection attack against a user database; 2) 2: open a session on 1 using a network service such as ssh using valid user authentication parameters; and 3) 3: establish a connection from1 to2 without asking for authentication parameters by exploiting the rhosts mechanism that allows one system to trust another system. According to this

example, the intrusion detection system ensuring the monitoring of 1 and 2 using three separated rules considers 2and 3 as legitimate, however if there is a composite rule that includes the three actions, the intrusion detection system matches this rule and it considers such actions as malicious.

5. EXPERIMENTAL RESULTS

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 informationaboutmaliciousactivityforp ostmorteminvestigation.

No of Nodes	Existing 1	Existing 2	Proposed
40	29	14	36
80	55	32	71
120	83	69	109
160	125	101	148
200	160	129	185

Table 1: No of Nodes Identified

Table 1 represented into no of nodes identified in external attack values. Storage Area Network IDS is proposed into this phase. Proposed Malicious code injection is detect the more than external attacks in this phase. So it is better proposed concept of this phase.

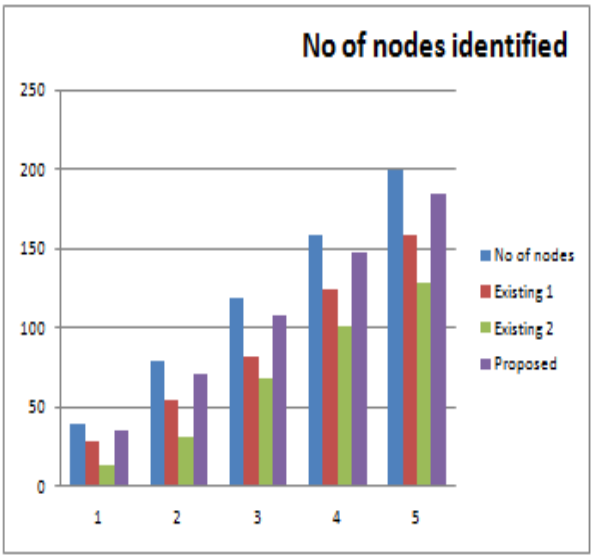

Figure 5: No of Nodes Identified

Figure 5 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.

No of Nodes	Existing 1	Existing 2	Proposed
50	26	15	41
100	51	33	89
150	122	104	137
200	159	129	176
250	218	200	233

Table 2: Reliability

Table 2 represented into reliability values in external attack values. Storage Area Network IDS is proposed into this phase. Proposed Storage Area Network IDS is detect the more than external attacks in this phase. So it is better proposed concept of this phase.

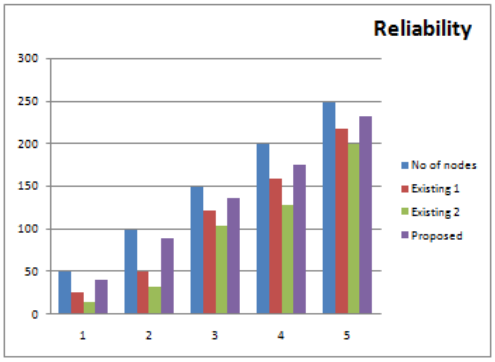

Figure 6: Reliability

Figure 6 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.

No of Nodes	Existing 1	Existing 2	Proposed
100	69	47	81
200	156	142	179
300	267	233	282
400	354	329	368
500	451	430	477

Table 3: Consistency

Table 3 represented into consistency in external attack values. Storage Area Network IDS is proposed into this phase. Proposed Storage Area Network IDS is detect the more than external attacks in this phase. So it is better proposed concept of this phase.

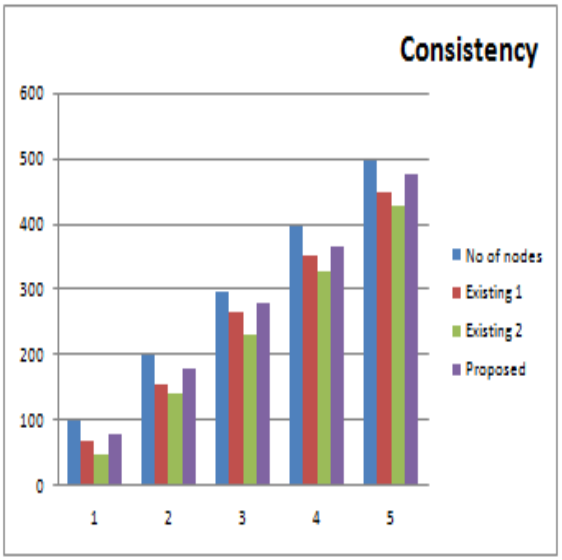

Figure 7: Consistency

Figure 7 is represented into consistency 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.

No of Nodes	Existing 1	Existing 2	Proposed
25	17	22	9
50	36	41	25
75	60	69	47
100	79	95	62
125	99	111	89

Table 4: Error Reporting

Table 4 represented into error reporting in external attack values. Storage Area Network IDS is proposed into this phase. Proposed Storage Area Network IDS is detect the more than external attacks in this phase. So it is better proposed concept of this phase.

Figure 8: Error Reporting

Figure 8 is represented into error reporting 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.

No of nodes	Existing 1	Existing 2	Proposed
30	17	26	9
60	38	49	22
90	66	81	58
120	96	113	83
150	135	140	111

Table 5: Traffic Latency

Table 5 represented into traffic latency in external attack values. Storage Area Network IDS is proposed into this phase. Storage Area Network IDS is detect the more than external attacks in this phase. So it is better proposed concept of this phase.

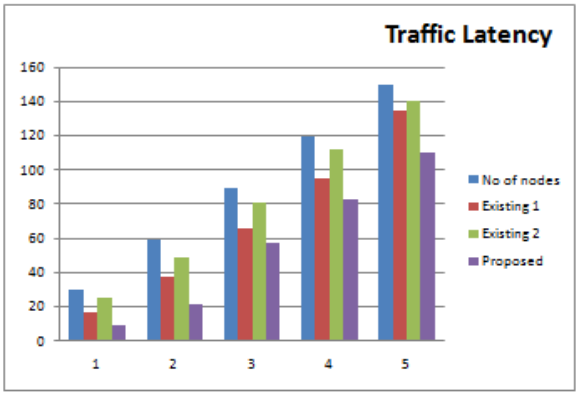

Figure 9: Traffic Latency

Figure 9 is represented into traffic latency values in graphs. External attacks find the existing values are high but their Storage Area Network IDS values are lower than detect the among the nodes in the external attacks.

No of nodes	Existing 1	Existing 2	Proposed
100	**61**	**39**	**85**
250	**146**	**101**	**197**
300	**201**	**177**	**269**
450	**333**	**316**	**399**
500	**409**	**388**	**454**

Table 6: IDS Throughput

Table 6 represented into IDS throughput in external attack values. Storage Area Network IDS is proposed into this phase. Proposed Storage Area Network IDS is detect the more than external attacks in this phase. So it is better proposed concept of this phase.

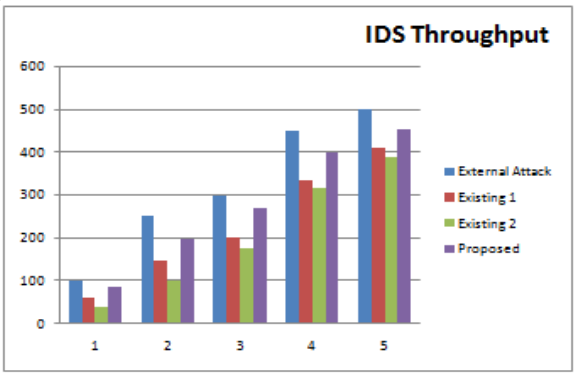

Figure 10: IDS Throughput

Figure 10 is represented into IDS throughput 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.

CONCLUSION

The detection of attacks against SAN systems requires the cooperation of different components located at the SVC and SDs connected to SAN. In the first part of this

paper, we have presented a novel intrusion detection and tolerance system for SANs. This solution ensures dynamic intrusion detection based on a set of components and structures that are introduced in the SAN system at the SVC and each SD. The distribution of the SVC still among the limits of the proposed scheme since it is designed for a centralized SVC. We have presented an intrusion detection and tolerance scheme that may be integrated or added to traditional intrusion detection and tolerance systems in order to detect attacks related to traffic exchanged between transaction-based applications running in a wireless environment. The proposed scheme introduces an enhancement to available intrusion detection schemes by using communication model that enables the exchange of a set of notifications messages that enhances intrusion detection and uses a set of rules that are suitable for transactions structure and processing in wireless environments.

REFERENCES

[1]. Anderson, J.P., Computer Security Threat Monitoring and Surveillance, Technical report, James P. Anderson Co., Fort Washington, PA., April 1980. On Software Engineering, vol. SE-13, pp. 222-232, February 1987.

[2]. 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

[3]. Singh, S.P. (2010) Data Clustering Using K-Mean Algorithm For Network Intrusion Detection, Thesis, Lovely Professional University, Jalandhar.

[4]. 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.

[5]. C. Kruegel, F. Valeur, and G. Vigna. Intrusion Detection and Correlation: Challenges and Solutions. Springer-Verlag Telos, 2004.

[6]. L. R. Halme and R. K. Bauer. AINT misbehaving – A taxonomy of anti-intrusion techniques. In Proc. of 18th NIST-NCSC National Information Systems Security Conference, pages 163– 172, 1995.

[7]. D.E. Denning, An Intrusion-Detection Model, IEEE Transactions on Software Engineering, vol. SE-13, pp. 222-232, 1987.

[8]. Dinakara K, "Anomaly Based Network Intrusion Detection System", Thesis Report, Dept. of Computer Science and Engineering, IIT Khargpur 2008

[9]. Guy Bruneau – GSEC Version 1.2f," The History and Evolution of Intrusion Detection", SANS Institute 2001.

[10]. Ilgun, Koral, USTAT:a real time IDS for Unix, Proceedings of the 1993 IEEE Computer Society Symposium on research insecurity and privacy, 1993.

[11]. Mark Crosbie, Gene Spafford, Defending a Computer System using Autonomous Agents, Technical report No. 95-022, COAST Laboratory, Department of Computer Sciences, Purdue University, March 1994.

[12]. D. Anderson, T. Frivold, A. Valdes, Next-generation intrusion detection expert system (NIDES), Technical report, SRI-CSL95-07, SRI International, Computer Science Lab, May 1995."

[13]. Paxson, Vern, Bro: A system for detecting network intruders in real-time, Computer Network, v 31, n 23, Dec 1999.

[14]. Ning,Wang X.S, Jajodia S, Modelling requests among cooperating IDSs, Computer Communications, v 23, n 17, Nov, 2000."

[15]. J. E. Dickerson and J. A. Dickerson, "Fuzzy network profiling for intrusion detection," In Proceedings of the 19th International Conference of the North American Fuzzy Information Processing Society (NAFIPS), 13-15 July 2000, pp. 301 – 306.