IOSR Journal of Engineering (IOSRJEN)
e-ISSN: 2250-3021, **p**-ISSN: 2278-8719
Vol. 3, Issue 8 (August. 2013), ||V4|| PP 52-60

Designing a multi-layered security algorithms for network intrusion detection systems on virtual machines in the cloud computing

Mohammad Rafiee, Majid Mohammadi
[1] *Department of Science and Research Branch, Islamic Azad university, Kerman, Iran*
[2] *International Center for Science, High Technology and Environmental Sciences,Shahid Bahonar Kerman University, Kerman, Iran.*

ABSTRACT : With the emergence of new generation networks and providing new types of services, as cloud computing networks, the need to provide security that are compatible with the performance of these networks lead us to offer an algorithm for using the context of security available in prior networks. In this algorithm, intrusion detection systems(IDS) are activated depending on the service which selected by the user. The packages of each user is related to the network layer and standard protocols and will be selected considering that intrusion detection system. The techniques used in this paper are based on the parallel use of intrusion detection systems on network virtual machines. As objectives of this algorithm, we can point to reducing the computational load in contrast to large number of users and applications. Considering that internet is the context of using these networks, we need to reduce the response time to the user as short as possible.

Keywords: - *Information Security, Parallel Processing, Intrusion Detection System (IDS), Virtual Machine(VM)*

I. INTRODUCTION

According to growing demands and joining new customers to the world of computing, computing systems should change and act stronger and more flexible than before. Among these; cloud computing was offered as a model superior to a system that currently has the ability to response to most of the demands and requirements.[1] Flexible infrastructure of cloud computing and virtualization technology has provided the new facilities to support business activities. But for cloud service providers, attending in market competition is very important and thus they are trying to have secure and flexible data centers using different techniques of management, security, computing and storage.[2] One techniques for improving the safety and increasing the responding speed is using the security multi-layered algorithm for intrusion detection systems. we provided security by using different mechanisms and security policies for our organization and hid its properties behind at the Fire-walls, DMZ, VPN and…. before this. In Classical methods the firewall was used to encounter the attacks. In the fire-walls access to specific ports or protocols were limited. Since in majority of web attacks ports and protocols (HTTP protocol and port 80) are used which they are applied in normal ones, these methods are not suitable for preventing and coping with current web attacks. Therefore, intrusion detection systems have been used for web attacks. [3].

II. THE NEED FOR NETWORK SECURITY IN CLOUD COMPUTING

In general; if all of the security criteria implemented on the large scale, they will be much cheaper .So much better investments on security lead to better protection against risks. Security is the most important issue of cloud customers. Customers' purchase decisions based on the reputation in the field of privacy, precision and the flexibility of security services that are provided by suppliers. This has led to intense competition among cloud providers in security issues. [4]

In 1998, the Lincoln Laboratory of MIT University, generated DARPA data set[5] for which a simulation environment was created attempted to simulate a common LAN network traffic of U.S.A Air Force. LAN network worked like a real environment, but had different attacks as well as normal traffic simulated and entered in the data set. In this data set for each TCP/IP connection 41 quantitative and qualitative features were derived, 34 of them are numeric and 7 ones are symbolic. These data include 24 types of attacks that are classified into four categories. [6] And usually same classification is used for web attacks in general status:

- **Denial of service attacks (DOS):**Denial of service are some attacks in which attacker made processing source or memory so busy that it cannot respond to requests from Authorized users. Their requests are rejected and therefore authorized users have no access to the system.
- **Remote to local attacks (R2L):**

In some attacks attacker send packages through network to a machine, and exploit machine vulnerability and then reach impermissibly to local access. This attack occurs when the attacker is able to send a package through the network to a machine, but the machine has no account for user and through exploitation of specific vulnerability in a machine can create an authorized user name.

- **User attack to root(U2R):**

In these attacks, the attacker has a normal account on the system by which can improve his/her accessibility and with exploitation of a vulnerability it can reach a higher level of accessibility or access level of root user.
There are various kinds of U2R attacks among them buffer overflow is one of the most well known.

- **Probing:**

In these attacks, the attacker makes a computer network fully scanned to gather information on network machines, or to find known vulnerabilities of its machines. With the help of this information various parameters can be adjusted for exploitation of vulnerability.

2.1. Intrusion detection systems and their types

Detection of computer attacks (intrusion) has been defined as "to identify people who use computer systems without permission (crackers) or those who have legal access to the system, but abuse their authority (domestic threats)."[7]
All current intrusion detection systems control attacks or host computers or network connections to gain data and information, hence they are divided into two categories:

- Host intrusion detection systems

These intrusion detection systems, located on a specific host computer and control it. One of the most important decisions in designing host intrusion detection systems is to choose features and traits that we want to control. In the design stage, we should determine how much data we should collect for processing. Because of high complexity we cannot control all features.

- Network intrusion detection systems

A network intrusion detection system, monitor and control all passing packages through a given network connection. Such systems placed the network interfaces in irregular mode, and monitor whole network through hiding themselves from attackers.
Nowadays intrusion detection system, use three ways to analyse and detect attacks. Therefore, intrusion detection systems are divided into three general categories; abuse detection systems, abnormality diagnosis and description-base detection.

2.2. Proposed model for using intrusion detection system

To clarify this proposed model we begin with an example. This is simply sensible and understandable, increasing the number of used filters to purify water will decrease the speed of accessing water in addition to increasing the benefits of using cleaner water and ensuring no unauthorized penetration of particles in water. So for the benefit of cleanliness and suitable speed of accessing water we should make a balance between filters number and their using places.

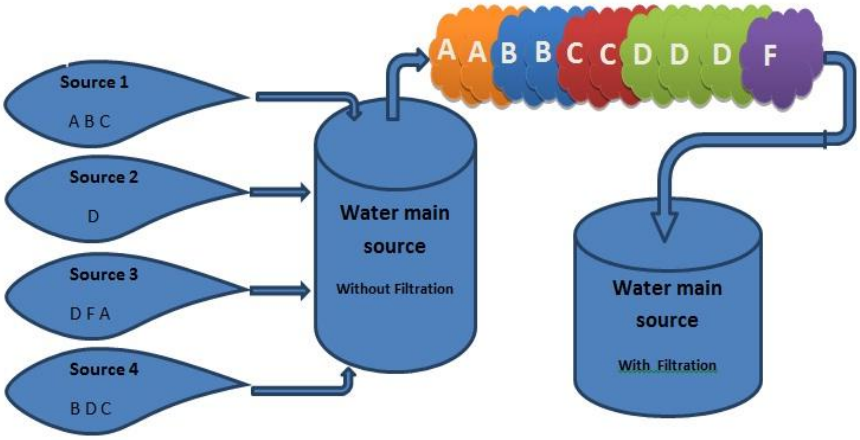

Fig.2-1: consecutive use of filters

As shown in Figure 2-1, water comes from four different sources into the main source, each of them have different pollution. For example, source 1 has the pollution of types A, B, C and source 2 has pollution of type D. Some sources have common pollutions and some others have unique pollutions. One of the disadvantages of the combination of water with different pollution is combination of some pollutions and

creation of new type of pollution that is impossible to identify. If the nature of water types and its pollutions are identified before using filters to purify water, we can use appropriate filters for different types of water and pollutions. In this way, the time, energy, and cost of purifying water will be decreased.

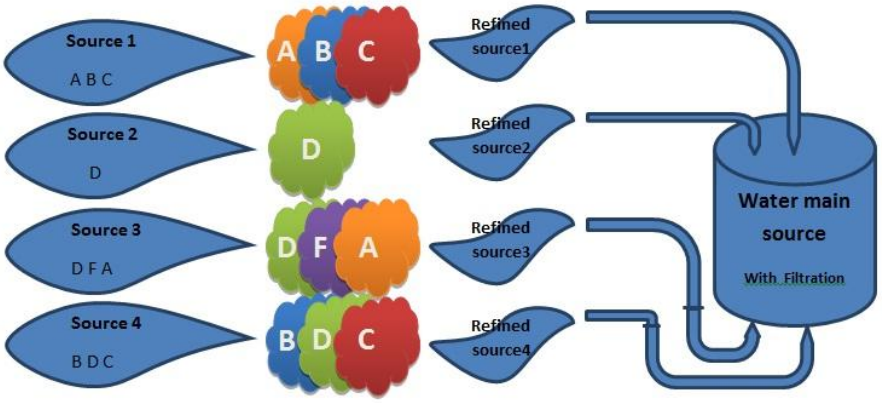

Fig.2-2: parallel use of filters

The same example can be used to establish security in cloud computing network. First, different types of services in the network should be detected, various services must be pre-defined[8. Users can select only the services they need among offered services and are not allowed to use these undefined services, therefore in order to use facilities of cloud computing network, people dealt with a list of various services that are offered by each network.

- Users should be able to select their needed services easily and without ambiguity.
- All of the capabilities and facilities that are offered by the cloud network, should be mentioned in form of a service with the suitable topic.
- Various services should be grouped according to the resources (hardware, software)and logical relation between the services.
- Never one type of service should be used for more than one title.

After identification of offered services and their performance, we should identify attacks and threats that affect these services and threaten their security. These attacks can be classified according to each category in different groups. If services are grouped according to mentioned items and uniformity in service type, we can predict that the threatening attacks in one group are all from one type of structure, thus they can be classified easily in equal groups.

Now is turn to identify intrusion detection systems for each mentioned grouped attacks; there are different types of intrusion detection systems with different rules for each group and each one acts in different ways.

Here we introduce shortly some intrusion detection systems. Each of these systems has different architecture, some are based on irregular statistic rule and some on signature or matching pattern.

AAFID: this system uses a series of factors to display packages in host network and in this way it detects intrusion.

AIDE: this tool applies cost free instead of Tripwire and acts almost similar to Tripwire.

Detection Toolkit: This system is used for Sprinkled Honeypot services in servers. Its usage is so that if attackers attack this system, in fact they have attacked virtual system and it is difficult to leave system without effect.

Defense worx: this system has acceptable application in high traffics and can identify unknown traffics.

Host Sentry: this system acts according to the Login Anomaly Detection (LAD). This state enables system managers to pay much attention on logins action.

ICMP info: this system monitors ICMP info package to identify packages with anomaly performance.

Imsafe: this software acts on Linux and monitor packages to identify attacks type.

IPlimit: this software is designed to prevent DOS attacks and limits connections to prevent such attacks.

Log Check: it shows system logs and security and error mail in certain times and according to timing program.

Packemon/ Packet Monster: this system listens to network traffic and registers doubtful packages and displays its abstract.

Port Sentry: it is for identifying systems which scan ports and stop traffic related to the port.

Saint Jude: it is for extension main core IDS and is used to protect hosts comprehensiveness.

Snort: this system is a type of open source intrusion detection. Snort does some works like Packet Sniffing, Packet Logging and NIDS.

If multiple intrusion detection systems are used together, they will use a standard format to send message to each other. Intrusion detection message exchange format (IDMEF) is a standard format that enables intrusion detection tools to send events and observe warnings to intrusion detection tools and managing unit.[9] Offered security system acts in a way that at first each user of cloud computing network selects his/her needed service in login time. After identification and authentication by password, there should be some cloud network services offered by index table. Each service has a unique ID up to end of operation and computation of user. Username is attributed is by this ID, at every possible time that a user is identified through applying service and transaction.

IDS attributed to each service should observe some rules to provide more security, in this regard, the following points will be considered:

- IDS are defined as unique and no two services can use the same ID.
- The validity period of each ID is until every user logs out the service and up to the end of service time.
- For same services and different users different IDS are defined.
- That means if at the same moment two users request A1 service from a group services, for each ID a unique ID will be defined.
- Never at each moment of time, there is a system of two identical ID. After entering each user ID, the network addresses server to select time. Then, index of selected service in database table of Cloud Storage and some other things about the service identifier is created for the user.

When the intrusion detection system is activated, for the better security of the cloud network environment, the higher security for users and increasing the reliability of cloud environments, and increasing competition among private cloud network, a group or a sub-group of the intrusion detection systems are activated instead of activate an intrusion detection system from one group.

Each Diagnostic system has variables for ranking in terms of response time, error rate, correct warnings and consumer resources. To activate the intrusion detection sub-systems, systems with higher level of ranking are activated, and at the end, ranking table is updated again. Activated systems act in parallel, and for each of these systems there will be a virtual machine for defining user service and these virtual machines consider the user transactions in parallel form. As soon as any of these machines detects any intrusion, they can prevent further transactions and retrieve resources from VM of user and then log the user out.

2.3. Way of facing intrusion in this pattern

Two attacks can affect cloud computing network, first state is done when attacks are done by members of the network who have a user account, next one is done by persons who are not members of the network and want to penetrate the system and use it without permission. To tackle the latter a firewall should be loaded on the network to block all ports and services. Therefore, only people whose passwords are valid can pass through the wall. In other words, we can say that until a particular service provider is not known, all groups of intrusion detection systems are active. When a valid ID and password is declared and verification and service selection take place (the first mode is activated) other groups except those associated with this service will be inactive.

The latter attacks can also be operated in the same way until we have not entered the first mode. Intrusion detection systems may operate in parallel .And some lines are determined to continue coordination between the transaction and the intrusion detection system. In this phase, the highest rank of any group can be considered active. All or a subset of any class will be activated to work on the network in parallel form. This selection is done by the manager and he is responsible for network security. How important is the degree of security for cloud computing network and to what extent this network would be attacked also are decided by the manager. In this part, the intrusion detection systems operate in parallel. And when each one recognizes any influence, they announce it to protect system. If intrusion detection system does not recognize any influence until reaching to coordination lines and the transactions continue working until all intrusion detection systems reach to coordination lines. The last line of coordination is the moment that the service is selected by the user. The purpose of coordination lines creation is having some intrusion detection algorithms with more processing time is. And some have less response time, because it can be guaranteed that a transaction is acceptable from view of all intrusion detection systems. In order to notice that an intrusion has not been detected, we must wait until all diagnostic systems complete their own algorithm stages. The user will not notice the delay time to achieve coordination lines. In fact we try to eliminate or reduce the lag time as short as possible. To Process the stages, we do each algorithm of intrusion detection system in parallel or multi-threading form. Number of threads in each algorithm that are between the coordinated lines are equal. So we have no wasted time and delay for calculation and transaction. In each phase of synchronization, when thread processing of each algorithm in

each line fully completed, its processing resources will be divided between the other threads. The amount of the reliability and security of the coordination lines is dependent on number of the transactions being conducted. And reducing the number of lanes will reduce the security. if the running transaction is an attack, probability of its influence will increase. Therefore, increasing the coordination lines can cause extra overhead, delay time and lowering the speed of computation. We Should commensurate with the importance of security for cloud computing network, amount of computational resources, the number of available servers, members of cloud computing network. Offered services in this network determine the appropriate number of coordinate lines.

Computation base on cloud computing machines is according to network virtualization. A virtual machine on the servers of cloud computing network is created to offer selectable service for every user. To implement intrusion detection systems can be used in parallel virtual machines to execute each one. The virtual machines act together as a parallel processing system, and each has its own intrusion detection algorithms which process in parallel and in mode of a few threads.

2.4. Simulation and comparison

To compare two parallel using of intrusion detection systems and consecutive use of intrusion detection systems together we use Arena Master Development Software Manufacturing Rock Wall Company. for simulation, we should first define some entities as inputs, after definition of the entity, its arrival rate per unit time is denoted. This entry can be tailored to different states of the system defined according to the time. Entities can enter a fixed number per unit time, random number per unit time or regular numbers according to one of the statistical distribution (exponential, gamma, triangular, Poisson, etc). To enter system we can define several different inputs at the same time and make special arrangements for each entry. The unit time is also adjustable and we can select different times for each event. To proceed, we need a complete diagram of target system (Fig.2-3). In other words, we should identify all effective stages of system, obtain input and output of each stage. The effect of each input on the event can be calculated by considering the time used for processing each stage albeit in best, worst and normal state (time order). Therefore we can identify modes and conditions that result in changing event processing and so consider the total output of all the move steps up to the event.

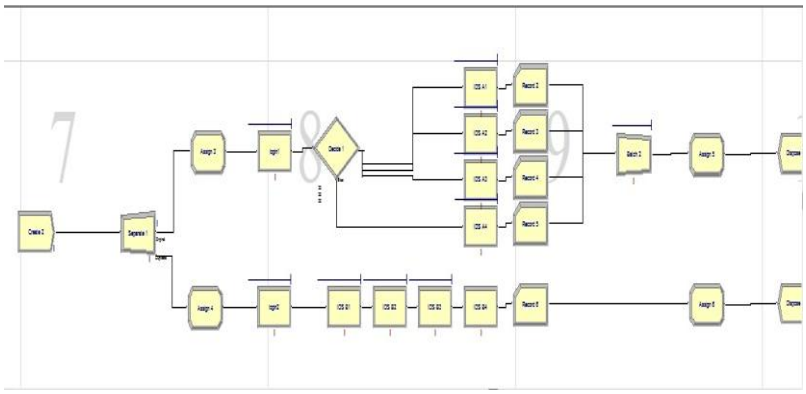

Fig.3-2: Diagram for state of system stages

After each processing module for virtual of intrusion detection systems, a data record module and variable values are used (Fig.5-2). all resources that are taken of the system should be returned to the system by the last step in processing. We've done it by the Delay Release. Finally, at the end of work we use Dispose module to receive inputs of system and compare input number of each system after a period of time.

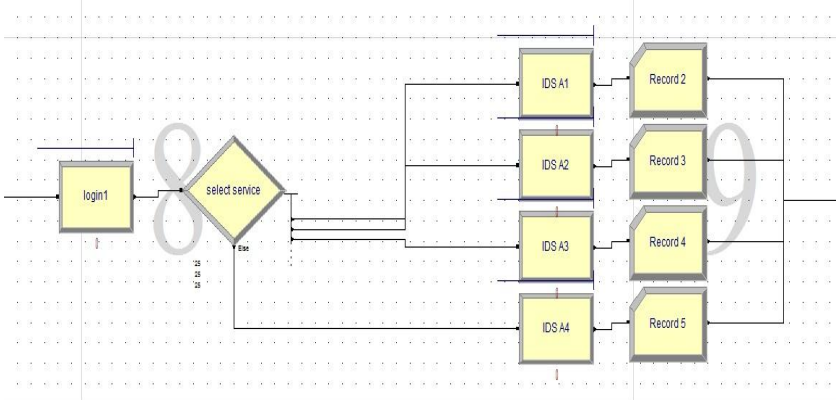

Fig.4-2: Selecting service and determining the direction of user movement in system

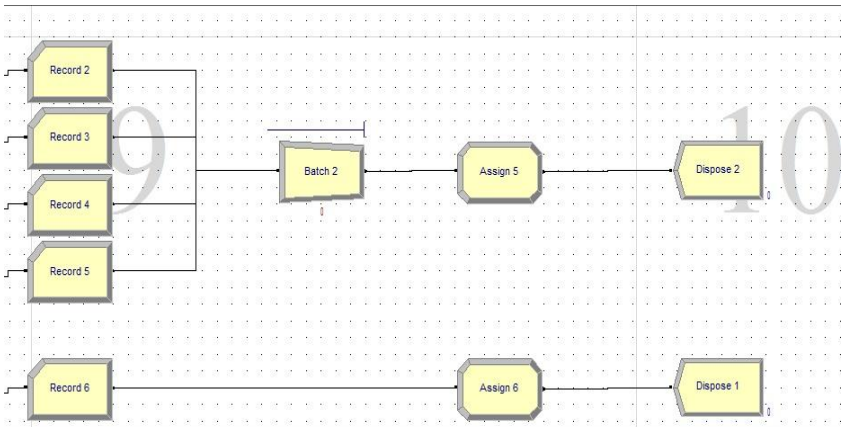

Fig.5-2: Saving changes and transactions by records before reaching to the output

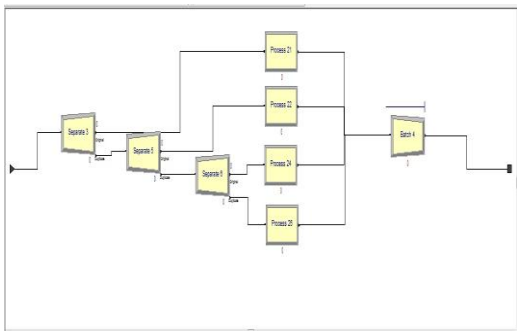

Fig.6-2: Universal IDS1

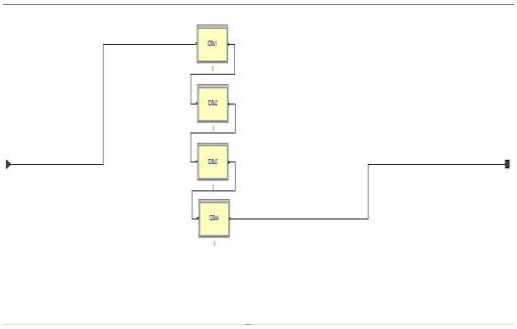

Fig.7-2: Universal IDS

Here the result of this simulation is presented during operation of the proposed system in 24 hours besides the presumption diagnostic system.

```
Reading program file: C:\USERS\MOHAMMAD\DOCUMENTS\GPSGATE\MODEL1.P
Beginning replication 1 of 1


SIMAN Run Controller.


24.178793 Hours>
                                ARENA Simulation Results
                              Linezer0 - License: 7328734345

                             Summary for Replication 1 of 1

Project: Unnamed Project                                  Run execution date : 4/28/2013
Analyst: Rockwell Automation                              Model revision date: 4/28/2013

Replication ended at time      : 24.178793 Hours
Base Time Units: Hours

                                      TALLY VARIABLES

Identifier                             Average   Half Width  Minimum    Maximum   Observations
-------------------------------------------------------------------------------------------

Request Transactions.VATime             2.0895     (Insuf)    1.1368     4.1655        300
Request Transactions.NVATime            .02431     (Insuf)    .00919     .17098        300
Request Transactions.WaitTime           10.420     (Insuf)    .01317     21.799        300
Request Transactions.TranTime           .00000     (Insuf)    .00000     .00000        300
Request Transactions.OtherTime          .00000     (Insuf)    .00000     .00000        300
Request Transactions.TotalTime          11.749     (Insuf)    .60995     23.674        300
IDS A4.Queue.WaitingTime                9.0922     (Insuf)    .01092     17.764         77
IDS B2.Queue.WaitingTime                5.6171     (Insuf)    .04044     11.127        106
login2.Queue.WaitingTime                5.4836      (Corr)    .00000     11.086        536
Batch 2.Queue.WaitingTime               .00000     (Insuf)    .00000     .00000        276
IDS B3.Queue.WaitingTime                5.2144     (Insuf)    .06509     10.992         47
IDS A1.Queue.WaitingTime                8.4306     (Insuf)    .00000     17.270         66
IDS B4.Queue.WaitingTime                5.0196     (Insuf)    .10736     10.938         24
Batch 4.Queue.WaitingTime               .00122      (Corr)    .00000     .02620       9740
IDS A2.Queue.WaitingTime                8.0368     (Insuf)    .00000     17.693         64
IDS A3.Queue.WaitingTime                8.7607     (Insuf)    .00000     17.245         70
IDS B1.Queue.WaitingTime                5.6239     (Insuf)    .00992     11.092        248
login1.Queue.WaitingTime                8.7905      (Corr)    .00000     17.757       1138

                                  DISCRETE-CHANGE VARIABLES
```

Identifier	Average	Half Width	Minimum	Maximum	Final Value
Request Transactions.WIP	7719.6	(Corr)	.00000	15626.	15626.
Vitrual machine.NumberBusy	.84954	(Insuf)	.00000	1.0000	1.0000
Vitrual machine.NumberScheduled	1.0000	(Insuf)	1.0000	1.0000	1.0000
Vitrual machine.Utilization	.84954	(Insuf)	.00000	1.0000	1.0000
Vitrual machine1.NumberBusy	.97115	(Insuf)	.00000	1.0000	1.0000
Vitrual machine1.NumberScheduled	1.0000	(Insuf)	1.0000	1.0000	1.0000
Vitrual machine1.Utilization	.97115	(Insuf)	.00000	1.0000	1.0000
IDS A4.Queue.NumberInQueue	103.75	(Corr)	.00000	208.00	207.00
IDS B2.Queue.NumberInQueue	58.628	(Corr)	.00000	144.00	142.00
login2.Queue.NumberInQueue	279.24	(Corr)	.00000	678.00	678.00
Batch 2.Queue.NumberInQueue	.00000	(Insuf)	.00000	1.0000	.00000
IDS B3.Queue.NumberInQueue	22.922	(Insuf)	.00000	58.000	58.000
IDS A1.Queue.NumberInQueue	98.397	(Corr)	.00000	210.00	210.00
IDS B4.Queue.NumberInQueue	9.8309	(Insuf)	.00000	23.000	23.000
Batch 4.Queue.NumberInQueue	.49154	.02924	.00000	2.0000	.00000
IDS A2.Queue.NumberInQueue	102.23	(Insuf)	.00000	225.00	225.00
IDS A3.Queue.NumberInQueue	108.44	(Corr)	.00000	219.00	219.00
IDS B1.Queue.NumberInQueue	124.63	(Corr)	.00000	288.00	288.00
login1.Queue.NumberInQueue	1783.1	(Corr)	.00000	3733.0	3732.0

COUNTERS

Identifier	Count	Limit
Record 2	66	Infinite
Record 3	64	Infinite
Record 4	70	Infinite
Record 5	76	Infinite
Record 6	24	Infinite

OUTPUTS

Identifier	Value
Request Transactions.NumberIn	16754.
Request Transactions.NumberOut	1128.0
Vitrual machine.NumberSeized	961.00
Vitrual machine.ScheduledUtilization	.84954
Vitrual machine1.NumberSeized	1415.0
Vitrual machine1.ScheduledUtilization	.97115
System.NumberOut	300.00

Simulation run time: 0.60 minutes.
Simulation run complete.

Fig.8-2: Result of simulation performance in system during 24 hours and 17 minutes

Due to the variable values on system output, we can draw following diagrams for virtual used machines and the amount of recorded output packages from intrusion detection systems that shows during system work in gall resources of the virtual machine are at optimal use and number of all output packages are much more than the serial mode using of intrusion detection system.

Fig.9-2: Diagram for the amount of using virtual machines,VM1 in proposed system, VM in traditional system

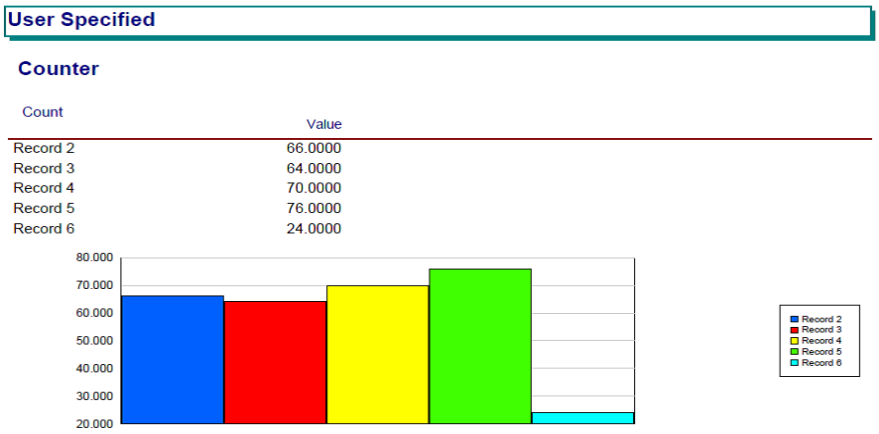

Fig.10-2: Chart for recorded and processed packages by intrusion detection systems in two proposed and ordinary system, in same time and same input packages number, Records 2,3,4,5 are related to proposed system and Record 6 is related to tradition system.

III. CONCLUSION

The If today's network of security providing use intrusion detection systems commonly as used in servers of small networks, loss of network efficiency and its servers is the result. And this is because of increased users and clients. In other words, if we have to pass all input packages and requests through intrusion detection systems to provide security and identify intruder transactions, the results will be increasing the amount of response time, increasing the amount of computation in processor, increasing energy consumption, reducing server lifetime, environment pollution, reducing number of users, reducing acceptance of network in competitive market, loss of income and failing the network. Therefore in order to process data faster and to respond more users at the same time, system should use parallel virtualization for transactions and different requests in its servers. According to different algorithms of machines migration load on servers should be distributed in the same ratio, so that servers can easily support their own virtual machines same as users requests for different services.

Performing virtualization for each user to offer specific service to him or her, it is better to use intrusion detection system in specific manner for each user by virtual machine. Consequently, other packages of this user will not monitor by other intrusion detection system that are not associated with this service, and processing load will be reduced, hence the response time will be less.

REFERENCES

[1] J.S. Chase, D.C. Anderson, P.N. Thakar , A.M.Vahdat, R.P. Doyle, "*Managing energy and server resources in hosting centers*" , Proceedings of the 18th ACM Symposium on Operating Systems Principles, PA 103-116, New York, USA, 2001.

[2] ZH. Shen, Li. Li, Fie.Yan, "*Cloud Computing System Based On Trusted Computing Platform*", International Conference on Intelligent Computation Technology an Automation, IEEE,2010.

[3] J. McHugh, *"Intrusion and intrusion detection"*, CERT Coordination Center, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA 15213-3890, USA,2002.

[4] Gartner. n, "*Seven Cloud-computing Security risks*", Network World, July 2008.

[5] KDD Cup 99 dataset, http://kdd.ics.uci.edu/database/kddcup99/kddcup99.html, August2005.

[6] Enica: Europen Network and Information Security Agency, "*Cloud Computing Benefits, risks and recommendations for information security*",2009.

[7] B. Mukherjee, T. L. Heberlein, and K. N. Levitt, *"Network intrusion detection"*, IEEE Network, 8(3):26–41, May/June 1994.

[8] W.Tsai, X. Sun, j. Balasooriya,"*Service – Oriented Cloud Computing Architecture*", Seventh International Conference on Information Technology, Arizona state, 2010.

[9] Qin X., Lee W., "*Statistical causality of INFOSEC alert data"*, Proceedings: Recent Advances in Intrusion Detection, LNCS 2820; Springer-Verlag, 2003, pp. 73-93.

[10] Michael Gregg, "*10 Security concerns for Cloud Computing*", Global knowledge instructor and ES Advanced Dragon IDS, Expert reference series of white papers, 2010.

[11] Damiano Bolzoni, Sandro Etalle and Pieter H. Hartel, "*Panacea: Automating attack classification for anomaly-based network intrusion detection sytems*", research program sentinels,Technology Foundation STW, Netherlands,2010.

[12] Risto Vaarandi, "*Real-time Classification of IDS Alerts with Data Mining Techniques*", Proceedings of the 2009 MILCOM Conference, ISBN: 978-1-4244-5239-2, 2009.
[13] Sanja Sharma, Sonika Soni and Swati Sengar, "*Security in Cloud Computing*", National Conference On Security Issues in Network Tecchnologies, NCSI-2012, August, 2012.
[14] Tejashri J.Madavi, Yashashree V.Gawande and Madhavi B.Kale, "*Security in cloud computing*", International Conference on Advances in computer and management, ICACM, January, 2012.
[15] Kazi Zunnurhain and Susan V.Vrbsky, " *Security in cloud computing*", Computer Science Journal, Alabama, 2010.
[16] Klara Nahrstedt and Roy Campbell, "*Security for Cloud Computing*", Report to the National Science Foundation, Directorate for Computer and Information Science and Engineering (CISE) , Virginia, Marth, 2012.
[17] Ramgovind S, Eloff MM and Smith E, "*The management of security in cloud computing*", IEEE Journal, ISBN: 978-1-4244-5495-2, 2010.
[18] John Harauz, Lori M.Kaufman, Bruce Potter, "*Data Security in the World of Cloud Computing*", IEEE Security and Privacy Journal, Copublished by the IEEE computer and reliability societies,61-64, August, 2009.
[19] S. Subashini and V. Kavitha, "*A survey on Security in service delivery models of cloud computing*", Journal of Network and Computer Applications, 2010.