International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 6 (2018) pp. 4576-4582

Performance Analysis of Anomaly Detection of KDD Cup Dataset in R Environment

Dr. Anitha Patil[1]* and Srikanth Yada M.[2]

[1]Proferssor, Department of Computer Science and Engineering, Pillai HOC College of Engineering and Technology, Maharashtra, India.

[2]Associate Professor, Department of CSE, Tirumala Engineering College, Narasaraopet, Guntur, Andhra Pradesh, India.

** Corresponding author*

Abstract: In the present computing world, there is a fast development of innovations in the field of the network security and there are challenges through rising system dangers. Thusly, this influenced us to take these worldwide difficulties as a high need undertaking to ensure the system. The most vital reason for intrusion detection system is to distinguish assaults against data frameworks. It is a security technique endeavoring to distinguish different assaults. In this paper, we utilized KDDCUP dataset we distinguish assaults utilizing Snort on this dataset.

Keywords: Intrusion Detection System; Anomaly; Misuse; Network

INTRODUCTION

These days, it is the primary issue to keep up the system security. As PC arrange is developing step by step, security is the most effective system for a PC organize. Firewalls are very little proficient to secure system from assaults since firewall can just distinguish the assaults which originate from outside of the system [8]. With the quick utilization of PCs and simple entry to the web on the planet, the approach to assault and cheat a framework has likewise quickly expanded. Intrusion, as it were, is an illicit procedure of entering or claiming another's benefits.

System Security is turning into a critical issue for every one of the associations, and with the expansion in the information of programmers and gatecrashers, they have influenced numerous fruitful endeavors to cut down prominent to organization systems and web administrations. With the current advances in the field of system security, a method called Intrusion Detection System are create to additionally upgrade and influence your system to secure. It is a route by which we can shield our interior system from outside assault, and can make a fitting move if necessary. Utilizing Intrusion recognition techniques, data can be gathered from known sorts of assault and can be utilized to distinguish in the event that somebody is attempting to assault the system.

This paper central focuses on recognizing the unpredictable relationship that has been seen by our IDS through Snort when we stream the KDDCUP DataSet over the system. Intrusion Detection System components as an arrangement of associations or system bundle mouth or sniffer, which in light of relationship of information little bundle inside with perceived sickness Anomalies abridge as a procedure, can start act and confirmation activities and data related to them in a recorded document as well as the database. Grunt is an all-around preferred Network-based Intrusion identification framework that is utilized to review organize bundles and contrast those parcels and the data of superb assault Anomaly and Snorts assault signature data database may likewise be legitimized time by time [12].

As the arrangement of associations, environment transforms into the multifaceted and gigantic level, and Intrusion occasions are changed over into incidental. It recommends far over the ground strains to the Intrusion identification advancements, requesting Intrusion location framework to unite dataset from the disparate arrangement of associations and host and mediator the activity of the entire arrangement of associations, proper advised Intrusion recognition and answer subsequently [9].

RELATED DATA

Intrusion identification is a procedure in PC arranges which assume a critical part in distinguishing distinctive kind of assaults. It is the method of watching the activities which go in a PC framework. Intrusion discovery gives three fundamental security marvels, for example, checking, distinguishing, and reacting. The thought process of Intrusion Detection System is to recognize inward and in addition external assaults. In like manner, we can state that an IDS's comprise of equipment part. To run equipment part good programming is likewise continuing with the framework [9].

Working on Intrusion Detection System resembles the security protection. The two suppositions in the field of Intrusion discovery are 1) client and program occasions are checked by PC frameworks and 2) conventional and Intrusion exercises can have very surprising conduct [10].

Types of Attack in a Network

Regarding the connection interloper casualty, assaults are, Internal, originating from possessing endeavor's workers or their business accomplices or clients and External, originating from outside, as often as possible by means of the web.

SYN Flooding

The SYN surge assault is, essentially, to send countless parcels and never recognize any of the answers. An SYN

surge is a type of foreswearing of-benefit assault in which an assailant sends a progression of SYN solicitations to an objective's framework. The assailant sends a few parcels yet does not send the "ACK" back to the server. The associations are henceforth half-opened and expanding server assets, a true blue client, try to interface yet the server declines to open an association bringing about a foreswearing of administration.

Flood Attack

The soonest type of dissent of administration assault was the surge assault. The assailant just sends more movement than the casualty could deal with. This requires the aggressor to have a speedier system association than the casualty. This is the least level of the refusal of administration assaults, and furthermore, the hardest to totally counteract, for instance, a UDP surge assault is a disavowal of administration assault (DOS) assault utilizing User Datagram convention, a sessionless/connectionless PC organizing convention. A UDP convention assault can be started by sending a vast number of UDP convention parcels to arbitrary ports on a remote host. Accordingly, the arbitrary host will:

- Check for application tuning in on that host.
- Sees that no application tunes in on that port.
- Reply with an ICMP goal inaccessible bundle.

Packet Sniffing

A bundle sniffer, in some cases alluded to as a system screen or system analyzer, can be utilized authentically by a system or framework head to screen and investigate organize an activity. Utilizing the data caught by the Packet sniffer, a manager can distinguish incorrect parcels and utilize the information to pinpoint bottlenecks and help keep up proficient system information transmission. In its basic frame, a parcel sniffer basically catches all bundles of information that goes through a given system interface. Normally, the bundle sniffer would just catch parcels that were proposed for the machine being referred to. In any case, if put into the wanton mode, the bundle sniffer is additionally equipped for catching all parcels navigating the system paying little heed to goal. By putting a parcel sniffer on a system in unbridled mode, a noxious interloper can catch and investigate the majority of the system activity. Inside a given system, username and secret key data are by and largely transmitted in clear content which implies that the data would be perceptible by dissecting the bundles being transmitted [8].

Spoofing

With regards to organize security, a caricaturing assault is a circumstance in which one individual or program effectively gives some sort of false data and along these lines picking up an ill-conceived advantage.

Viruses

A little bit of programming that duplicates itself on genuine projects and runs each time a program runs.Most can replicate and assault different projects. The accompanying is the most widely recognized sorts of infections:

Email infections: Moves around in email messages, as a rule, repeats itself via naturally mailing itself to many individuals in the casualty's email address book.

Worms: A little bit of programming that utilizations PC systems and security gaps to duplicate it. Worms can grow quickly checks a system for another machine that has a particular security gap and duplicates itself to the machine.

Spyware

Spyware is PC programming that is introduced surreptitiously on a PC to gather data about a client, their PC or perusing propensities without the client's educated assent. There are three classes of Spyware: Harmless yet irritating: This will change the default landing page of your program to some objective advertisements, fly up and so on.

Data gathering: This class of spyware is by and large inspired by gathering some sort of helpful data about you, the destinations you went by most, thus that outsider can send you focused on popup and advertisements.

Noxious: This class incorporates full logging and gathering data alongside sending private and secret data to the server.

Perfect IDS ought to have the accompanying highlights:-

Convenience: The property ensures that any irregular conduct can be identified inside a stipulated time or soon after that time.

High likelihood of location: It certifications to distinguish a large portion of the anomalous conduct in the system.

Low false-caution rate: This property permits a couple of quantities of false alerts.

Specificity: once the assault is distinguished adequate nitty-gritty data must be accessible so to show signs of improvement reaction.

Versatility: Scalability can be connected to of all shapes and sizes systems.

Low from the earlier data: This property needs a minimum of prior data concerning potential aggressors and their techniques [4].

LITERATURE SURVEY

The IDS appeared at the start of 1980, with James Anderson's paper, Computer Security Threat Monitoring, and Surveillance. How about we concentrate on how IDS has advanced since its Inception in mid-1980's. In 1983, SRI International, and particularly Dr. Dorothy Denning, started

chipping away at an administration venture which helps Intrusion location improvement. The point was to make client's profile in view of their movement by dissecting the review trail One year later, the primary model for Intrusion discovery, the Intrusion Detection Expert System (IDES), was produced by Dr. Denning which gave the structure to the IDS innovation development [15].

Mahoney portrayed the two models for inconsistency recognition framework for checking dubious movement. Above all else for passing just the information parcels of the greater part prerequisite, e.g. to begin with a few bundles of internal server asks for, the movement was sifted. Second, at the bundle byte stage to signal occasions that have not been found for a long traverse of time, the most widely recognized usable system conventions (IP, TCP, Telnet, FTP, SMTP, HTTP) were composed [2].

Mahoney and Chan describe an exact PHAD that decides the normal scope of qualities for 33 fields of the Ethernet, IP, TCP, UDP, and ICMP conventions. On the KDDCUP informational collection, PHAD distinguishes 72 of 201 objects of assaults, together with everything except 3 sorts that achievement the system conventions tried, at a speed of 10 false alerts for every day playing out the preparation on 7 days of assault less inward system activity. PHAD examined in different ways, and the better results were found by investigating system bundles and fields independently, and by utilizing uncomplicated no stationary structures [7].

Mahoney and Chan presented an arrangement of direction called learning calculation which structures plan of common nature from inconsistencies free system activity. Nature that bifurcates from the known typical outline signals conceivable novel assaults. Their Intrusion location framework is unique in two perspectives. In, to begin with, the no stationary display is exhibited in which the planning chances in view of the traverse of time from the time when the event of last occasion rather than the rate. Presently in the second, the Intrusion location framework screens the convention gathering keeping in mind the end goal to distinguish the obscure assaults that attempt to hurt plan blames in ineffectively observed qualities of the object programming. On the 1999 DARPA Intrusion location framework assessment data set, they distinguished 70 of 180 assaults and assigned to client behavioral peculiarities and convention abnormalities. As their ways are elective, there is a representative non-cover of their Intrusion recognition framework with the authentic. DARPA individuals, which symbolize that they can be taken in general to upgrade the scope [3].

Mahoney and Chan presented an arrangement of guideline called LERAD that works standards for distinguishing a couple of events in ordinary time arrangement data with long request dependence. They utilized LERAD to recognizing irregularities in arrange movement bundles and TCP sessions to distinguish novel Intrusions. LERAD comes about the real members in the DARPA dataset and recognized all assaults that emerge a firewall. LEARD is efficient for three causes. To begin with, just a little piece of the movement has been tried. Second, the standards utilizing just a little example of the preparation data has been created. Third, to build a little gathering of the rule that generally covers the data; a scope test has been utilized [4].

Aydin et al proposed a half and half Intrusion recognition framework which is the blend of abuse and irregularity based Intrusion location. In this paper, they took grunt as abuse based on PHAD and NETAD as inconsistency based Intrusion discovery. PHAD and NETAD are the oddities based factual strategy. Right off the bat, grunt is tasted on KDDCUP dataset then it distinguishes 27 assaults out of 201 assaults, furthermore PHAD is added to the grunt as a preprocessor (Snort + PHAD) is tried on the same dataset then the quantity of assaults identified is incremented up to 51 out of 201 assaults, at long last NETAD is added to the grunt and PHAD as a preprocessor (Snort + PHAD + NETAD) is tried on same dataset then the quantity of assaults distinguished is incremented up to 146 out of 201 assaults [8].

Nandiammai and Hemalatha proposed a technique named as crossbreed Intrusion discovery in which first they utilized the measurable based irregularity strategies, for example, ALAD, LERAD and PHAD at that point consolidate these techniques with a grunt which is abuse based. Initially grunt is tried on KDD Cup 99 dataset then it identifies 77 assaults out of 180 assaults after that PHAD is added to the grunt as a preprocessor (Snort + PHAD) is tried on the same dataset then the quantity of recognized assaults raises to 105 out of 180 assaults after that ALAD is added to the grunt and PHAD as a preprocessor (Snort + PHAD + ALAD) is tried on the same KDD Cup 99 dataset then the quantity of assaults distinguished increments to 124 out of 180 assaults after that LERAD and ALAD is added to the grunt as a preprocessor (Snort + LERAD + ALAD) is tried on the same KDD Cup 99 dataset then the quantity of assaults identified increments up to 149 out of 180 assaults. Besides, the upside of both regulated and unsupervised strategies has been utilized to build up a semi-administered technique. The semi-regulated approach requires less measure of named information with an overwhelming measure of unlabeled information. For semi-directed approach 5000 datasets are taken, in that 2500 taken as preparing stage and minimum is taken as a testing stage. Preparing stage incorporates both the named and unlabeled information together. The consequence of semi-managed approach indicates 98.88 % identification rate and 0.5529 % false caution rate [5].

Nandiammai and Hemalatha proposed an Intrusion identification framework which is the blend of four methodologies, for example, grouping of information named as EDADT (mix of cross breed PSO with C4.5), grunt based preparing named as half-breed IDS (mix of grunt which is abuse based IDS with ALAD and LERAD which are peculiarity based factual calculation), semi-managed approach, moving DDoS assaults named as Varying HOPERAA. Right off the bat, EDADT calculation gives the result as 92.51% affectability, 88.39% specificity, 95.37% precision, 0.72% false alert rate.

Furthermore, crossbreed IDS gives the result as examined above and the Third semi-managed gives the result as additionally talked about above. At long last in HOPERAA

calculation, a variable clock float strategy is proposed to dodge the customer sitting tight time for server and in the meantime, message misfortune is maintained a strategic distance from extraordinarily. In this way, HOPERAA can limit the message exchange delay and in addition execution time [6].

PROBLEM STATEMENT

With the developing interest of system and advances in the field of the system, nowadays every association wishes to have their individual system and what's more, they need to interface or communicate with each other dependably. So Network Security is ending up increasingly critical and furthermore getting the more confounding issue with late advances and with expanding request. At the point when an association endeavor to uncover itself internationally then the odds of having escape clauses in their system are high because of accessibility of their system universally and odds of getting assaults and even dangers increments.

PROPOSED SYSTEM

The Proposed system is consisting of the following modules.

Preparation Phase: The data required to perform the experimental analysis is carried out in this preparation phase. The major role of this phase is that it collects attack wise labeled data and prepares individual files.

Pre-processing phase: The process of handling noise and missing values is done in this phase. Attributes with less variant and high variant nature are going to be identified and eliminated by applying an unsupervised filter. The remaining attributes are used in the feature extraction process.

Feature Extraction phase: This phase evaluates the usefulness of attribute subset by taking into consideration of forecasting capability of an individual feature in addition to the rate of redundancy between each attribute. The features derived using proposed approach is shown in Table 1.

Table 1: List of Derived subset of Features

Attribute Number	Attribute Name
5	src_bytes
6	dst_bytes
12	logged_in
23	count
29	same_srv_rate
32	dst_host_count
33	dst_host_srv_count
34	dst_host_same_srv_rate
35	dst_host_diff_srv_rate
37	dst_host_srv_diff_host_rate
38	dst_host_serror_rate
40	dst_host_rerror_rate

Feature subsets are also derived and observed by using Support vector machine Attribute evaluator with the support of Ranker search method. The top ten ranked features are shown in table 2

Table 2: Feature Subset with Ranks

Rank	Attribute Number	Attribute Name
10	19	num_access_files
9	31	srv_diff_host_rate
8	24	srv_count
7	17	num_file_creations
6	22	is_guest_login
5	23	count
4	6	dst_bytes
3	21	is_host_login
2	20	num_outbound_cmds
1	9	urgent

Classification Phase: To classify the KDD cup dataset kernel-based support vector machine method is applied.

Total numbers of support vectors generated are 630 and the objective function value is 412.95 and observed training error is 0.045128. To obtain the better accurate results in the form of anomalies, we have proposed the kernel algorithm, which is in iterative nature and repeats for each and every support vector in order plot in the specific hyperplane. The proposed kernel has produced better results compared to existing methods. The parameter values for the proposed method are shown in Table 3.

Table 3: Parameters Taken and observed during analysis phase

Parameter	**Value**
SV type:	C-svc (classification)
cost	C = 1
kernel function	Gaussian Radial Basis
sigma	3.09620853587519e-06
Number of Support Vectors	630
Objective Function Value	412.9563
Training error	0.045128

The proposed algorithm has divided the vector into hyperplane values using p and q vector attributes and sigma value is used to measure the cost value based on the dimension of the data points chosen by the kernel function. The proposed kernel function is as follows in figure 1.

```
Ker_fun (sigma = 1)
{
  rval <- function(p, q = NULL) {
    if (!is(p, "vector"))
      Stop ("p must be a vector")
    if (!is(q, "vector") && !is.null(q))
      Stop ("q must a vector")
    if (is(p, "vector") && is.null(q)) {
      Return (1)
    }
    if (is(p, "vector") && is(q, "vector")) {
      if (!length(p) == length(q))
        stop ("no. of dimension must be the same on both data points")
      return (exp(sigma * (2 * cross_product(p, q) – cross_product (p) -
        cross_product (q))))
    }
  }
  return(new("rbfkernel", .Data = rval, kpar = list(sigma = sigma)))
}
```

Figure 1: Pseudo code for Kernel function

EXPERIMENT RESULTS

The KDD cup data has been tested using the kernel Support Vector Machine algorithm and the performance measures are as follows in Table 2. The Proposed algorithm has produced better results compared to the existing methods. The classifier parameters are as follows in Table 4.

The chosen parameters are in relevance to the proposed method and they are classification type and cost of the performance and Gaussian Kernel function.

Table 4: Measure Performance Values

Accuracy	Sensitivity	Specificity	FAR
99.62%	99.01%	100%	0.01

The anomalies observed aside with normal packets are sown in the figure2.

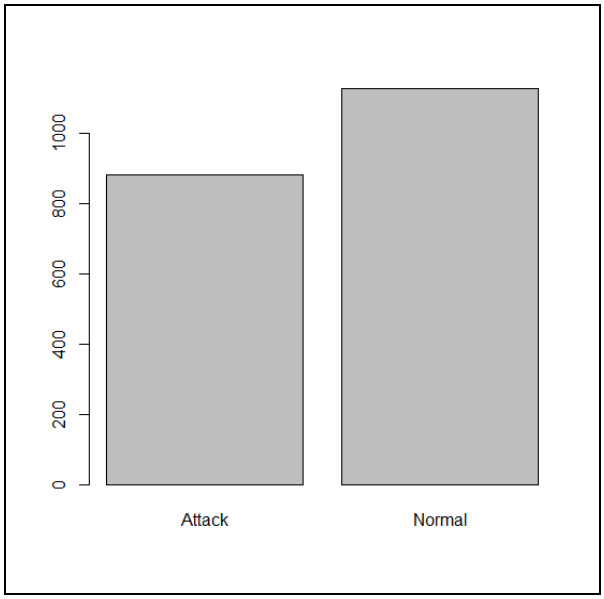

Figure 2: Plot-anomaly vs. normal packet Prediction

The Confusion matrix and statistics observed during the classification phase are shown in figure 3.

```
Confusion Matrix and Statistics

          Reference
Prediction Attack Normal
    Attack    852     24
    Normal     98   1018

               Accuracy : 0.9388
                 95% CI : (0.9273, 0.9489)
    No Information Rate : 0.5231
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.8768
 Mcnemar's Test P-Value : 3.866e-11

            Sensitivity : 0.8968
            Specificity : 0.9770
         Pos Pred Value : 0.9726
         Neg Pred Value : 0.9122
             Prevalence : 0.4769
         Detection Rate : 0.4277
   Detection Prevalence : 0.4398
      Balanced Accuracy : 0.9369

       'Positive' Class : Attack
```

Figure 3: Confusion Matrix and Statistics

The summarized Training set and Test set are measured with a total of 41 attributes and for each attribute. The obtained aggregate value is depicted in Table 5. For attributes of Factor, types will not have any aggregate values; such values are represented with label “NA”.

Table 5: Summarized Values of Training set and Test Set

	TrainingSet				TestSet			
name	median	min	max	nlevs	median	min	max	nlevs
duration	0	0	337	0	0	0	216	0
protocol_type	NA	211	3180	3	NA	110	1633	3
service	NA	0	2166	30	NA	0	1146	23
flag	NA	1	3192	9	NA	0	1690	7
src_bytes	237	0	54540	0	237	0	54540	0
dst_bytes	946	0	988002	0	896.5	0	125015	0
land	0	0	1	0	0	0	1	0
wrong_fragment	0	0	3	0	0	0	3	0
urgent	0	0	0	0	0	0	1	0
hot	0	0	30	0	0	0	30	0
num_failed_logins	0	0	5	0	0	0	1	0
logged_in	1	0	1	0	1	0	1	0
num_compromised	0	0	38	0	0	0	22	0
root_shell	0	0	1	0	0	0	1	0
su_attempted	0	0	0	0	0	0	0	0
num_root	0	0	54	0	0	0	39	0
num_file_creations	0	0	21	0	0	0	2	0
num_shells	0	0	2	0	0	0	2	0
num_access_files	0	0	2	0	0	0	1	0
num_outbound_cmds	0	0	0	0	0	0	0	0
is_host_login	0	0	0	0	0	0	0	0
is_guest_login	0	0	1	0	0	0	1	0
count	2	1	255	0	2	1	255	0
srv_count	4	1	160	0	4	1	160	0
serror_rate	0	0	1	0	0	0	1	0
srv_error_rate	0	0	1	0	0	0	1	0
rerror_rate	0	0	1	0	0	0	1	0
srv_rerror_rate	0	0	1	0	0	0	1	0
same_srv_rate	1	0	1	0	1	0	1	0
diff_srv_rate	0	0	1	0	0	0	1	0
srv_diff_host_rate	0	0	1	0	0	0	1	0
dst_host_count	1	1	255	0	1	1	255	0
dst_host_srv_count	255	0	255	0	255	0	255	0
dst_host_same_srv_rate	1	0	1	0	1	0	1	0
dst_host_diff_srv_rate	0	0	1	0	0	0	1	0
dst_host_same_src_port_rate	1	0	1	0	1	0	1	0
dst_host_srv_diff_host_rate	0	0	1	0	0	0	1	0
dst_host_serror_rate	0	0	1	0	0	0	1	0
dst_host_srv_serror_rate	0	0	1	0	0	0	1	0
dst_host_rerror_rate	0	0	1	0	0	0	1	0
dst_host_srv_rerror_rate	0	0	1	0	0	0	1	0
AttackType	NA	1945	1955	2	NA	965	1045	2

CONCLUSION & ACKNOWLEDGEMENT

Intrusion Detection System identifies assaults utilizing Anomalies that convey malignant and hurtful assaults. Anomaly-based IDS can be utilized to recognize known assaults; then again obscure assaults are distinguished trough Anomaly-based IDS. Irregularity based IDS empowers assault identification that has Anomalies which are not in the database of officially accessible assault designs.

Grunt is open-source IDS arrangement which isn't utilized for identifying assaults yet can be utilized for preventive activities as well, for example, when assaults are identified association can be blocked quickly to quit entering of any malevolent and assaults to the system framework. Accordingly, Snort ought to be refreshed much of the time since it must be made acquainted with new assaults and dangers. Grunt can be utilized for the security of system frameworks from any potential assaults or dangers before they make any harm to arrange framework.

REFERENCES

[1] Mohammadreza Ektefa, Sara Memar, Fatimah Sidi and Lilly Suriani Affendey, "Intrusion detection using data mining techniques", IEEE, 2010.

[2] Matthew V. Mahoney, "Network traffic anomaly detection based on packet bytes", ACM, 2003.

[3] Matthew V. Mahoney and Philip K. Chan, "Learning no stationary models of normal network traffic for detecting novel attacks", ACM, 2002.

[4] Matthew V. Mahoney and Philip K. Chan, "Learning Rules for Anomaly Detection of Hostile Network Traffic", Florida Institute of Technology, Melbourne, FL 32901.

[5] G. V. Nadiammai and M. Hemalatha, "Handling intrusion detection system using a snort based statistical algorithm and semi-supervised approach", Research Journal of Applied Sciences, Engineering and Technology 6(16): 2914-2922, 2013.

[6] G. V. Nadiammai and M. Hemalatha, "Effective approach toward intrusion detection system using data mining techniques", Egyptian Informatics Journal (2014) 15, 37–50.

[7] Matthew V. Mahoney and Philip K. Chan, "PHAD: Packet Header Anomaly Detection for Identifying Hostile Network Traffic, Florida Institute of Technology", Melbourne, FL 32901.

[8] M. Ali. Aydin, A. Halim Zaim and K. Gokhan Celyan, "A hybrid intrusion detection system design for computer network security", Computer and Electrical Engineering 35(2009) 517-526.

[9] Qingqing Zhang, Hongbian Yang, Kai Li, and Qian Zhang, "Research on the intrusion detection technology with hybrid model", 2nd Conference on environmental science and information application technology, IEEE, 2010.

[10] Sumaiya Thaseen and Ashwani Kumar, "Intrusion detection model using a fusion of PCA and optimized SVM", IEEE, 2014.

[11] Divya and Surendra Lakra, "SNORT: A Hybrid intrusion detection system using artificial intelligence with a snort", International journal computer technology & application, Vol 4(3), 466-470, 2013.

[12] Vinod Kumar and Dr. Om Prakash Sangwan, "Signature-based intrusion detection system using SNORT", International Journal of computer application & information technology, 2012.

[13] Natttawat Khamphakdee, Nunnapus Benjamas and Saiyan Saiyod, "Improving intrusion detection system based on snort rules for network probe attack detection", International conference on information and communication technology, IEEE, 2014.

[14] Kapil Wankhade, Sadia Patka, and Ravindra School, "An efficient approach for intrusion detection using data mining methods", IEEE, 2013.

[15] J.P. Anderson, "Computer Security Threat Monitoring and Surveillance", tech. The report, James P. Anderson Co., Fort Washington, Pa, 1980.

[16] M. Naga Surya Lakshmi, Dr. Y. Radhika," Effective Approach For Intrusion Detection Using KSVM And R", Journal of Theoretical and Applied Information Technology 15th September 2017. Vol.95. No.17