An Optimal Approach for Intrusion Security in Cloud

Jitendra Kumar Seth
Assistant Professor
Dept of Information Technology
Ajay Kumar Garg Engineering College,Ghaziabad, India

Satish Chandra
Assistant Professor
Dept of CSE
Jaypee Institute of Information Technology, Noida, India

ABSTRACT
Cloud computing is internet based technology that provides online resources on pay per use basis. This computing technology is dependent on virtualization. By registering with cloud providers customers can use services (soft and hard) on pay per use basis. Customers are getting services in the form of virtual machines, they interact with these virtual machines to do their jobs. In recent survey it is found that security is the top most concern associated with cloud services. Customers are always afraid of adopting clouds services as their data are stored out of their network boundary and administration. Researchers in cloud are trying to develop security models to protect cloud services from intrusions including internal and external. In this paper we have proposed an intrusion security model that uses the optimum cloud resources and increases the resource and service availability.

Keywords
IDS, Cloud, Virtual Machine, Availability, Classifier, ANN

1. INTRODUCTION
Cloud is an online service hence all threats to internet are also applicable to cloud services either in the same form or in some enhanced form. Security and trust are major hurdles in adopting cloud services. IDS are software or hardware based or both systems that monitors the system events for any security signature as per the system security policy. There are two types of intrusion detection systems misuse detection and anomaly detection. Misuse detection system uses pattern of well known attacks or weak spots of the system to match and identify known intrusions. Misuse detection systems are not effective for novel intrusions that have no known signature, rule or pattern. Anomaly detection systems are effective in novel attacks or unknown attacks in which no prior knowledge of intrusion are required. It uses deviation from normal usage profile of users that may be an intrusion. Anomaly based intrusions raised more false alarms in comparison with misuse detection because a new usage profile may be a new normal use of the system [1]. Cloud is providing services to a number of users, few of them may be untrustworthy. There are always a tradeoff between the security level and the performance of the IDS. If IDS is providing greater security by using huge datasets and patterns then it consumes more resources in terms of CPU Cycles, memory and may degrade the performance in terms of training time. An efficient IDS is required that does well on both side in security as well as in performance. Even a number of research efforts had been carried out in area of efficient IDS, the perfect IDS is still a research challenge in the area of computer security [2].

2. LITERATURE REVIEW
MADAM ID [1] is a misuse detection model. Audit data from the 1998 DARPA Intrusion Detection Evaluation Program are used for experiment. *tcpdump* data of 7 weeks of network traffic and Solaris BSM (Basic Security Module) data are taken. Bro is used for packet filtering and reassembling feature. The intrusion detection framework make use of program for learning classifier and meta classifier, association rules for link analysis, and frequent episodes for sequence analysis. Data mining methods (association rules and frequent episodes) are applied to compute the frequent pattern. Classification program RIPPER is used to learn detection model. RIPPER, a classification rule learning program, generates rules for classifying the telnet connections. Information gain method is used for feature selection. The experiment results shows that the frequent patterns mined from audit data can be used as reliable user anomaly detection models, and as guidelines for selecting temporal statistical features to build effective classification models.

Chirag Modi et. al [3] proposed a Framework to integrate NIDS in Cloud Infrastructure. Snort and Decision Tree (DT) classifier are used to implement the proposed framework. NSL-KDD and KDD experimental intrusion datasets are used to evaluate the performance of the framework. Snort is used to detect known attacks and DT is used to predict the intrusion based on stored network events. Each node of decision tree represent name of packet feature. And leaf represents the class label. ID3 decision tree classifier is being used. The eucalyptus on ubuntu was used for experiment. NIDS are installed at each node. The feature with higher information gain is used to select splitting node. Out of 41 features in dataset, 17 features were used to train the classifier. These features are protocol type, Service, Flag, src bytes, dst bytes, Land, wrong fragment, Hot, num failed logins, root shell, is guest login, Count, srv count, srv rerror rate, diff srv rate, dst host same src port rate, dst host srv serror rate. Results show that more than 95% intrusions on both data sets are detected correctly.

Leila Mechtril et.al. [4] proposed a PCA based intrusion detection method. PCA is a dimensionality reduction technique. It finds combination of original variables with largest variance. Author project the original data about user to principal components. It finds min and max threshold (using Euclidian distance of training set) of each user class. The principal component of new vector is extracted and the Euclidian distance of projected vector is computed with stored vectors, if the difference reside in the interval of thresholds then it was normal if not, abnormal behavior is detected. The same practice is repeated for all attacks. The KDD Cup '99 dataset were used for experiment and 100 connection records were taken for experiment. Three principle components were chosen to perform the detection. Experiment shows false

positive 1%, false negative 0%, true detection 99%, identification of attack type is 100%.

Yong-Xiang Xia et.al [5] proposed a method of detecting intrusion using incremental SVM. SVM is used to learn from huge dataset with large dimension data. Computing a SVM is very costly in terms of time and memory consumption. Hence a number of SVMs are connected in such a way so that each incremental SVM in the chain is trained by the previous SVM output and new batch of training dataset. For two class problems SVM finds a hyperplane separated by support vector, for more than two class data set, it is mapped to high dimensional data using kernel function. In high dimensional space it is possible to find hyperplane separating linearly. SVM classifiers are built on local systems connected to a central system. SVM detects intrusion and produce the new sample and send to central server which in turn distribute to all other SVM and evaluate the new sample to train other local SVM classifier. Author ranked features based on audit trail for each class. Five classes in KDD CUP 1999 dataset are found i.e. normal, probe, DoS, U2R, and R2L.Gaussian kernel function is used in SVM implementation, LIBSVM3 were used in experiment. From the experiment it has been observed that, model increases detection rate and decreases false positive detection.

Noreen Kausar et. al [6] proposed a SVM based IDS mechanism with Principal Component Analysis (PCA). KDD 99 dataset is used. PCA is used to reduce the dimension of dataset. Radial Basis Function is used as SVM Kernel. Extracted features of Dataset is divided into feature subsets S10, S15, S20 etc. The idea is to use less features with higher accuracy which reduces training time. These subsets are provided to SVM with RBF kernel function one by one for training and testing purpose. The response of each training and testing subsets are evaluated depending upon the true positive, false positive, true negative, false negative etc. The subset with maximum accuracy and least false alarms is the best reduced PCA feature subset with SVM classifier for IDS. In the experiment S10 has best sensitivity (True Positive Rate) and S30 has best specificity (True Negative Rate). The subset S10 consisting of 10 features has maximum accuracy 99.465% and false alarms only 0. 525% is the best among all selected subsets reduced using PCA. The subset S10 was selected as best reduced feature subset using PCA traditional technique. There is no complexity as such in the classifier architecture and these reduced features improve classifier performance without having training overhead in processing all features.

Xin Zhang [7] proposed an IDS model using Rough Set Theory (RST) and Support Vector Machine. Author uses two packages WINPCAP and JPCAP. WINPCAP interacts with the OS and NIC to capture the packets. While JPCAP is a java package which gets the captured packets from the WINPCAP to the java program. From real time network, author extracted 14 features using rough set and PCA. Out of 29 only 14 features are extracted using RST and PCA and associate with SVM classifier. The experiment demonstrates that RST-SVM yields a better accuracy in comparison with PCA.

VICTOR [8] is integrated into VMM. It detects and isolates the malicious entity. The entity may be the VM, application or process. Author assumes that all inter and intra VM communication is through IP packets and all packets passing through the VICTOR. VICTOR architecture consists of various components, these are Packet Differentiator, OS Library and Repository, Detection/prevention Engine, Shared Packet Buffer, Analyzer. Packets generated by the VM are captured by the packet differentiator. The differentiator knows the packet originator OS, process or application. The differentiator sends packet originator information to OS library and repository and send packet to the detection/prevention engine. The OSLR has knowledge about resources allocated to each virtual machine and applications running on each virtual machine. The OSLR verifies the information sent by differentiator. If OSLR does not find any information about reported packet information by differentiator then it is treated as malicious and an alert sent to Detection/prevention engine. The detection engine is trained on OSLR records of VM, application and process to distinguish the traffics as malicious or normal. If the detection engine can not identify the packet then the packet is kept in packet buffer and a copy is sent to analyzer. The analyzer decides whether packet is malicious or normal in packet shared buffer with the help of OSLR and subsequent packet from the same virtual machine. Experiment is setup on two XEN VMM servers with 3 virtual machines. VICTOR is tested for flooding attack. Experiment shows VICTOR can also be used in event triggered attack.

SECaaS [9] is a service oriented architecture that handles security at Saas, PaaS and IaaS level. At SaaS level SECaaS protects user data and provider's software. At PaaS level SECaaS protects user's application and provider's platform. At IaaS level SECaaS protects virtual machine and provider's infrastructure. The main component of SECaaS is security manager. Security manager helps user to choose and configure the security service and single sign on for all chosen security service. Any security provider cloud is a subcloud of security manager. Security manager can be a web service. The limitation of SECaaS is communication overhead between cloud security service provider and user.

In all above survey the IDS system uniformly handles intrusion for all users regardless the security need of the user. In next section we have proposed a model IDSMM that categorize the user based upon their security requirement and assign them the optimistic IDS program to handle intrusion.

3. THE PROPOSED SECURITY MODEL

Cloud provides large scale computing resources to each customer. Cloud computing components are various distributed computing resources, hypervisor and data. Cloud computing uses virtualization using which cloud provides virtual OS to their users. They uses hypervisors that interacts with host OS on behalf of guest OS and produces response for the guest OS. Security of cloud includes securing of virtual machines as well as hypervisor security. The structure of VM is shown in figure1 below. Cloud computing is a network based computing technology hence all threats to a network are also applicable to cloud directly or in some enhanced forms. User's personal and private data are resides at cloud datacenters, out of user's administration hence there are

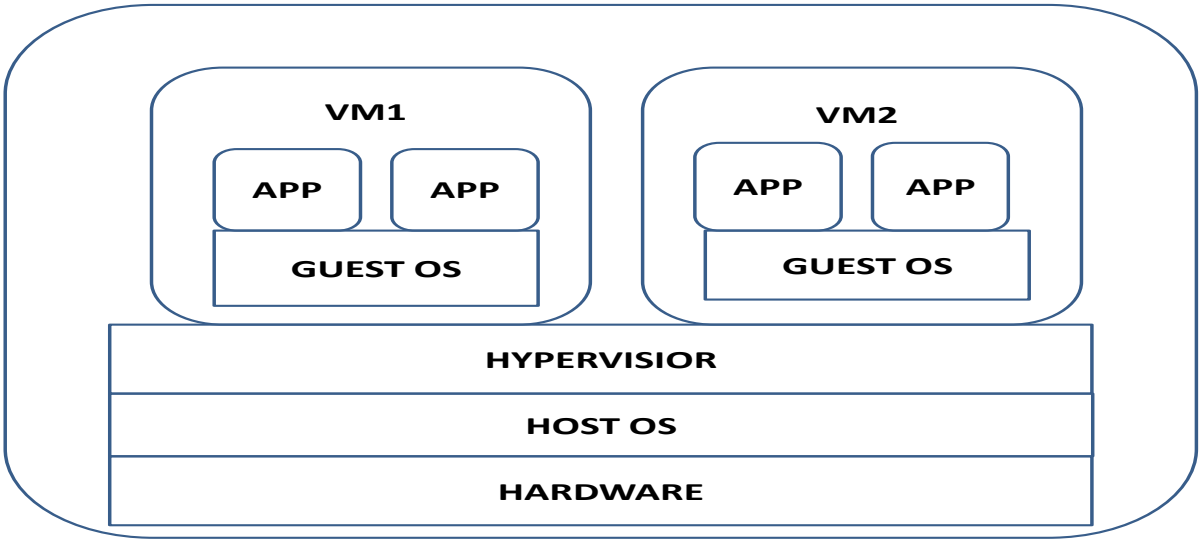

Figure1: Structure of VM

always risks to secure these data from all other users including intruders and administrators. Customers are running different operating systems and applications in their virtual machines hence it is a difficult job for cloud service providers to protect customer's virtual machine [8]. The customer's virtual machines are always prone to different type of attacks. Due to the loop holes in OSs and various weaknesses in TCP/IP stack attacker can easily attack on virtual machines by exploiting such weak points. It is shown in [10] that it is not a difficult task for an intruder to collect information for a victim

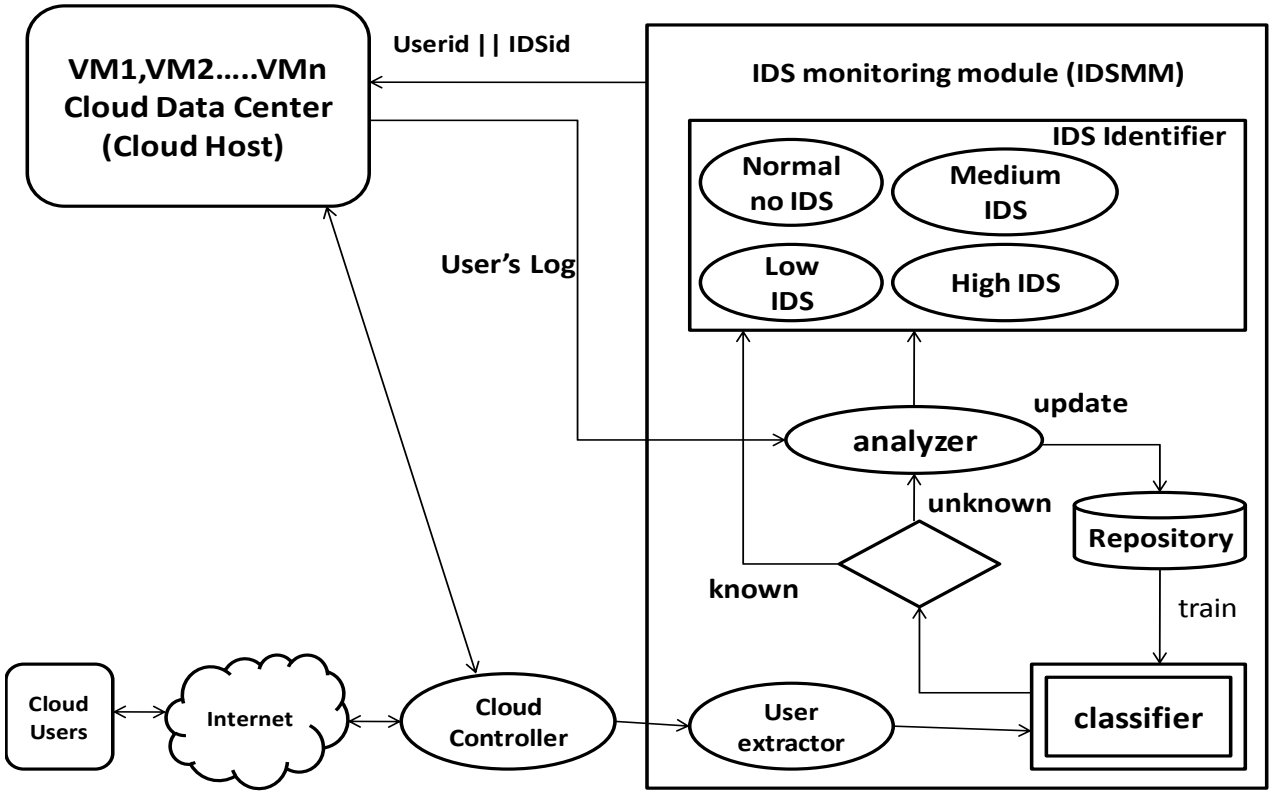

Figure 2: Proposed IDS model in cloud

virtual machine by co-locating own malicious virtual machine and attack on victim's virtual machine in cloud IaaS model. An IDS predicts attack based on a huge number of rules and attack patterns. Increasing the stored rule and patterns causes more comparison of incoming/outgoing packets hence performance overhead in IDS. Cloud providers have limited resource size to assign to IDS, as cloud users may demand any size of resource at any time. Cloud IDS should be optimized in processing and resource consumption. In this work we have proposed an IDS assignment model that classifies the user risk

level based upon stored rules and pattern and also updates stored rules repository on occurrence of new pattern and train the classifier at the time of minimal load on cloud. In our proposed model user is classified into various categories normal, low, medium and high based upon their risk level. Once the level of user is classified most suitable IDS (low, medium, high) with optimum resource can be assigned to user's VM to cope with the intrusion. Jun-Ho Lee et.al [11] proposed multilevel IDS for effective intrusion detection in cloud with efficient resource utilization and increases resource availability. AAA is defined as authentication, authorization and accounting module. AAA chooses most suitable IDS as per the user anomaly behavior. Three security level is defined high, medium and low. User's behavior is recorded in cloud database. When a user logins to cloud the user is analyzed by AAA module that assigns most suitable IDS to the user. For new users the low level IDS is assigned to the user and user behavior is monitored and recorded for most suitable IDS. High security level users log audit is at top priority and low security user log audit is at low priority.

The basis of our proposed work is that the security need of cloud users are different, the cloud service provider should not provide security to all users in the same way. Few of the intrusions are handled by using very less resources and data processing and few are required very huge resource and data processing. In our proposed work, we have addressed this issue. The proposed work is shown in figure 2. Cloud user's behavior is recorded in the repository. The stored users behavior is classified into four category normal, low, medium and high. Each time user logins to the cloud, the cloud controller authenticate the user and sends the user details to the IDSMM (IDS monitoring Module). The IDSMM extract the user's attributes and forward to the ANN (Artificial Neural Network) classifier trained on the stored users behavior in repository and decides the risk level (normal, low, medium, high) of the user then sends the risk level to the IDS identifier. The IDS identifier sends userid and IDSid (normal, low, medium, high) to cloud host where all these IDS (low, medium, high) are residing. The cloud host then assigns the identified IDS to the user's virtual machine, if the user is classified as normal no IDS is assigned to the user.

If the user behavior is unknown or new user logins then the classifier is not able to classify the user's risk level in such case user is unclassified and no IDS is assigned to user and details are sent to the analyzer. The unclassified user's log is provided to the analyzer at high priority. Analyzer monitors the user for any suspicious behavior such as sudden burst of traffic, abnormal usage of resources (CPU, memory, network), random packet to the same/multiple destination, packets with spoofed source address, and packets destined to non standard port numbers etc. analyzer decides the user's risk level and updates the repository with user's risk level. The classifier is trained with the updated records when CPU is idle. The analyzer receives user logs from cloud host periodically.

If a user's behavior classified as of low risk level then low risk IDS is assigned to the user. If the behavior crosses the category and match with the signature of medium level risk then the analyzer replaces the low IDS with medium IDS, in the same way if high risk user reaches to low level risk and remains in the same risk level for a defined period of time then high risk IDS is replaced with low risk IDS by the analyzer and so on. On promotion and demotion of risk level the IDS is also replaced dynamically.

3.1 Benefits of proposed IDSMM

1. The proposed model increases resource availability as cloud user can demand any size of resource at any time hence resource availability is a major concern in cloud.

2. Optimum resource utilization by IDS system.

3. Faster response produced to normal users as no IDS monitoring is applied to normal users. As in cloud most of the users are normal user.

Few of the malicious events that may be used to categorize the risk level are as follows: Login failed, Admin login after working time, Increase in memory size of VM, Exponential increase in traffic on a VM, Communication of a VM to a new VM, Non registered IP address access, Defined vulnerable port number, Abnormal guest OS shutdown or restart [11], unauthorized file access, unauthorized root directory access etc.

4. CONCLUSION

In the proposed IDSMM the system uses optimistic IDS based upon requirement of user security level. Cloud is a pool of resources that must be allocated wisely to increase the resource availability. The proposed model optimizes the cloud resource allocation for user security hence increases the resource availability to the cloud users.

5. REFERENCES

[1] Wenke Lee, "A Framework for Constructing Features and Models for Intrusion Detection Systems" ACM Transactions on Information and System Security, Vol. 3, No. 4, November 2000.

[2] Leila Mechtri1 et.al. ," Intrusion Detection Using Principal Component Analysis" in (ICESMA), IEEE 2010

[3] Chirag Modi et. al. ," A Novel Framework for Intrusion Detection in Cloud" in SIN'12,ACM October 2012.

[4] Leila Mechtri1 et.al. ," Intrusion Detection Using Principal Component Analysis" in (ICESMA), IEEE 2010

[5] Yong-Xiang Xia et.al.," An Incremental SVM for Intrusion Detection Based on Key Feature Selection" in Third International Symposium on Intelligent Information Technology Application, IEEE,2009.

[6] Noreen Kausar et. al.," An Approach towards Intrusion Detection using PCA Feature Subsets and SVM", in ICCIS IEEE, 2012.

[7] Xin Zhang," The Application of Machine Learning Methods to Intrusion Detection" in IEEE 2012.

[8] Udaya Tupakula et. al. ," Intrusion Detection Techniques for Infrastructure as a Service Cloud" in Ninth IEEE International Conference on Dependable, Autonomic and Secure Computing 2011.

[9] Mohammed Hussain," SECaaS: Security as a Service for Cloud-based Applications", Second Kuwait Conf. on E-Services and E-Systems,ACM,2011.

[10] Thomas Ristenpart et. al. , "Hey, You, Get off my cloud: Exploring Information leakage in third-party compute clouds", Proceedings of ACM CCS 2009.

[11] Jun-Ho Lee et.al. ," Multilevel Intrusion Detection System and Log Management in cloud computing", Advanced Communication Technology (ICACT), IEEE, 2011.