Proactive Autonomous Defense Shield (PADS) for Infrastructure as a Service (IaaS)

Khinar Thukral, Ankur Zilpelwar and M.Madiajagan

***Abstract*— Securing assets in the digital realm has become a challenge as there has been an exponential increment in the companies relying on the next generation technologies such as cloud computing. With the evolution of utility computing and Service Oriented Architecture as well as convergence of virtualization and web services into the notion of cloud computing, the attacks on these systems are indeed becoming more intricate and perplexing. Most inherently susceptible to the attack is Infrastructure as a Service (IaaS). This paper presents a technique of reinforcing the security of virtual machines whilst reducing the instances of false positives.**

***Index Terms:* Proactive Intrusion Detection, Virtual Machine Monitor, Simple Object Access Protocol, Hierarchical Autonomous- Cloud Intrusion Detection System.**

I. Introduction

Since the dawn of the internet, the analogy of 'Pay As You Go'(PAYG) type of services has been gaining popularity particularly, amongst enterprises requiring extensive but elastic computing infrastructure. The term 'Cloud Computing' although vaguely defined has different meanings under variegated personas. But in a nutshell, Cloud, offers a wide array of dynamic yet economical resources to the user. One of the prime reasons behind this extensive popularity of cloud was the need of a means to increase computing capacity of present infrastructure dynamically while at the same time without investing crucial budget in new infrastructure.

However, like any technology, cloud computing was ridden with its share of holes and gaps. Most common amongst the attacks on cloud infrastructure are Data Mining based attacks which are directed at data repositories on cloud with the intention of discerning knowledge from data. Another common attack on digital infrastructure is Distributed denial of Service (DDoS) attacks. These attacks are prominent in 'Cloud Computing' due to disseminated data and services on cloud. These attacks threaten the availability of infrastructure on cloud thereby affecting the credibility of Cloud provider as well as reducing the efficiency user operation.

Due to inherent weakness of TCP/IP stack, attacks such as ARP Poisoning, Zero Day attacks, phishing are most common in the Cloud Computing as are the XML Signature wrapping attacks, though seldom observed in Cloud [9]. In order to achieve the objective of securing cloud transactions, a variety of techniques are analysed here and knowledge gained is imbibed into the model of Proactive Autonomous Defence Shield (PADS). PADS would serve as an umbrella thwarting instances of attacks while at the same time learning from these attacks and assimilating the knowledge for future protection.

Manuscript received 23/03/2014; revised 09/04/2014.

K. Thukral is with the CS Department, Birla Institute of Technology and Science, Dubai Campus, Dubai UAE, pursuing M.E. in Software Systems; (e-mail: khinart@gmail.com).

A. Zilpelwar is with the CS Department, Birla Institute of Technology and Science, Dubai Campus, Dubai UAE, pursuing M.E. in Software Systems; (e-mail: ankur.zilpelwar@gmail.com).

Dr. M. Madiajagan is an Assistant Professor with the CS Department, Birla Institute of Technology and Science, Dubai Campus, Dubai UAE;(email: madiajagan@dubai.bits-pilani.ac.in)

II. Background

The service delivery models of cloud computing also known as the all too familiar SPI or SaaS (Software as a Service), PaaS (Platform as a Service) and IaaS (Infrastructure as a Service) are prone to attacks of different dimensionalities. Some attacks focus on the resources while others tend to focus on the virtualization aspect of cloud. Other category of attacks just aim at disrupting the service being provided by overloading the servers and causing both direct and indirect consequences to user operation. This paper is divided into seven sections: Securing Virtual Assets which deals with security aspects of IaaS computing in hybrid clouds; CloudSec-A novel VMM Approach which enumerates a security model for dealing with kernel level threats; Data Mining Based Attacks which explains the impact of data mining attacks and an approach to deal with them; Proactive Intrusion Detection which explains the driving motive towards proactive intrusion detection; and finally PADS which explains the Proactive Autonomous Defence Shield Model itself.

III. Securing Virtual assets

Cloud achieves the balance between efficient utilization of resources and customer satisfaction through two key characteristics :

- Multitenancy: Multitenancy in cloud refers to having multiple users (tenants) of the clouds sharing provider's infrastructure, including computational resources, storage services and applications [4].
- Elasticity: The concept of cloud computing involves ability of users to access resources as needed. This dynamic allocation (and eventual reallocation) of resources gave rise to the idea of elasticity.

Virtualization which lies at the core of IaaS offering involves the mutual operation of a number of components, each of which may itself be open to security breaches. IaaS service delivery model consists of following components:

a) Service level Agreements(SLA)

SLA in cloud environment is a contractual document that guarantees the acceptable level of Quality of Services

ISBN: 978-988-19252-7-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

(QoS) that were initially agreed upon by the user [6]. It consists of SLA contract definition, SLA negotiation, SLA monitoring and SLA enforcement [2]. Specifically, Web Services Level Agreement (WSLA) has been developed to monitor and ensure the QoS as offered by the providers are met on the client side.

b) Utility Computing

Utility Computing is a model of packaging infrastructure resources (computing power, processing units, storage and monitoring) and offering these packages over the internet to the client as the paid utilities. This aspect of IaaS is the most vulnerable to attacks.

A wide array of attacks called Flooding Attacks involve an attacker sending a huge amount of 'nonsense' requests to a certain service in the system[7]. Hypervisor will be required to determine the validity of each and every request. This increases the workload of the system, which in the event of flooding attacks would eventually lead to a Denial of Service (DoS) to the legitimate user. When the attack is being generated by multiple systems (possible other virtual machines in the same system) targeting a single system, these attacks take the form of distributed denial of service attacks (DDoS).

The impact of Flooding Attacks has been described below:-

- Direct Denial of Service:

When hypervisor notices high workload on the flooded service, it will start to provide more computational power (more virtual machines, more server instances) to cope up with the additional workloads[7]. So, although cloud system is trying to work against the attacker, it is somehow also supporting the attacker by allowing him to do most possible damage to the availability of the system from a single flooding attack entry point.

- Indirect Denial of Service:

In this case, if other service instances happen to run on the same server as flooded service instance ,it may affect their own availability[7]. As the hardware resources are exhausted by processing the flooding requests, other service instances collocated on the same hardware are also no longer able to function as intended. Thus, denial of service on the targeted instance is also going to cause the denial of service of the services deployed on the same hardware server.

c) Cloud Software

An important component of IaaS is the Cloud Software. As cloud software is deployed over the internet, various instances of this software communicate through SOAP which is vulnerable to simple injection attacks. Most common attacks in this scenario involve injection of malicious service or VMs on the cloud through illegal modifications of service metadata. Another taxonomy of attacks involve creation of malicious user instances thus wasting the resources of the system (called metadata spoofing attack) or obtaining administrative rights to a machine through re-engineering of web service specifications (XML wrapping)[7].

IV. CloudSec- A novel VMM Approach

An approach to ensure data and operational security is Virtual Machine Introspection (VMI). This technique involves ways of bridging the gap between the raw stream of bytes of data flowing in and out of the VMs and the I/O requests, processes and system calls that this data can execute on the virtual machine OS.

The Cloud Virtualization Infrastructure (CVI) is a combination of three components:

- Hypervisor: It monitors the VM and controls the operation of the cloud.
- vSwitch: A mechanism that allows communication between VMs.
- Hosted Virtual Machines: VMs are instances of virtualized resources running on top of hardware layer. The VMs hide the details of implementation and present an abstract view of the system to the client.

The core idea behind CloudSec is to enable the monitoring of VMs externally at a hypervisor or VMM level by observing the hardware bytes such as memory pages and disk blocks and mapping this information to useful OS abstractions[9]. As the Kernel Data Structures (KDS) change dynamically due to user operations, CloudSec strives to reconstruct as well as monitor these changes to effectively detect and prevent kernel level data rootkits such as Dynamic Kernel Object Manipulation (DKOM) and Kernel Object Hooking[8].

CloudSec uses the notion of Virtual machine Introspection (VMI). It involves monitoring the virtual machine from a hypervisor level by measuring the information flowing in and out of the virtual machine. It then uses this information to reconstruct, what is actually going on at the Operating System level. This gap is referred to as the semantic gap. The key idea behind solving the semantic gap is how to accurately map underlying hardware memory layout and OS Kernel Level Architecture [10].

A high level architecture of the CloudSec model is shown in Fig 1.

- The Back-End Component:

This component enables the VMM to maintain control over the VMs which are hosted by the system in order to suspend the access to the hardware resources based on the triggers installed in the VM using front end component. The backend component notifies the CloudSec to perform necessary security checks, before control is given back to the VM executed instructions [10].

- The Front End Component:

This is a set of APIs which enables extracting information about the operating system of the VM which is being monitored from the hypervisor. These APIs enable the CloudSec security infrastructure to install memory triggers on the physical memory pages that need to be monitored making it an external extension of the hypervisor which controls a transparent access to physical memory [10].

Whenever the VM being hosted is up and operational, the VM with CloudSec is notified automatically by the backend via communication channels. The security software then creates a fresh thread for each newly active VM using "VM-Thread Pool Manager". Then the security infrastructure checks the control registers of the Virtual machines in order to know to memory layouts of the VM's architecture as well as kernel version which is then stored in Kernel Structures Definition (KSD). CloudSec security infrastructure then uses Semantic Gap Builder (SGB) to map

ISBN: 978-988-19252-7-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

the high level OS operations to low level byte streams. It then analyses this information to find the remnants of a possible threat.

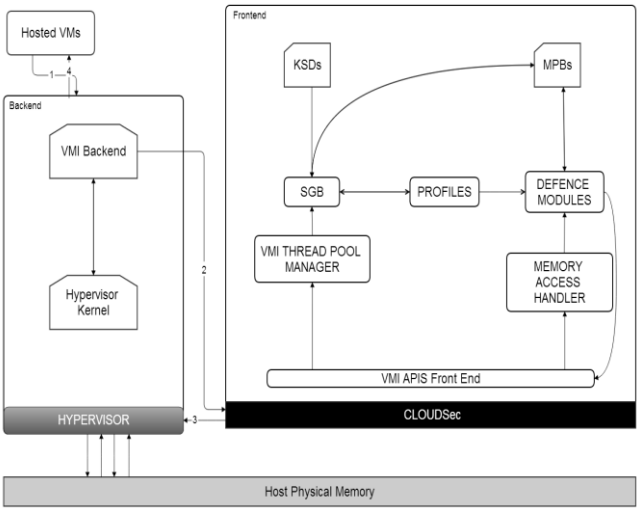

Fig 1: CloudSec Architecture

This approach as described in [11] preserves the integrity of user operations by running an IDS outside the virtual machine. But, where VMI can be used for monitoring the Virtual Machines for the purpose of providing security, the same method can also be used by an attacker to interpret the traffic flowing between the client and the VM in real time. Thus, VMI can be a double edged sword which can be utilized by the service provider and attacker alike.

V. Data Mining Based Attacks

The most common attacks on data are the data mining based attacks. These attacks tend to analyze globally stored data and interpret the data to gather knowledge.

Attackers outside cloud providers having an unauthorized access to the cloud, also have the opportunity to mine cloud data [14]. In order to successfully extract information using data mining techniques, two factors are necessary: proper amount of data and suitable mining algorithms. Hence, storing data at a centralized location will serve as a bottleneck being the single point of entry for all attacks.

A novel approach as described by Himel Dev et al. in [15] involves splitting of data into chunks and then distributing data amongst several cloud providers. This approach is based on the idea that even if the cloud provider performs mining on chunks provided to the provider, the extracted knowledge remains incomplete so that mining data from distributed sources is challenging [15].

This approach to protect the data is quite effective against mining based attacks such as multivariate analysis or clustering associative algorithms but since the data has now been distributed, it has become more vulnerable to distributed attacks specifically in hybrid clouds. These attacks include an intensive breach, which lasts for very short period of time that involves attacks being distributed across sites. Each attacker utilizes a slow paced time bound attacks while at the same time making sure that attacking frequency does not exceed the attack detection threshold of the system. An approach to deal with these threats has been enumerated by Hassan et. al. called Cloud Distributed Intrusion Detection System (CDIDS)[14]. CDIDS collects logs from various providers spread across the cloud infrastructure using various collectors. These collectors analyze the collected information for potential security breaches. If any anomaly is detected, then the information is sent to the Log Manager system. Log manager formats the alerts received from various providers and correlates them with the general security rules to decide whether the system is under attack or not. These detected attacks, if any, are forwarded up to the alarm mechanism to alert the cloud provider for a possible breach [14].

VI. Proactive Intrusion Detection

In accordance with the principles of cloud, an intrusion detection system no matter how effective in detecting threats should also be able to make autonomous decisions. One such system has been proposed by Kholidy et. el. in [17] called Hierarchical Autonomous- Cloud IDS (HA-CIDS) for cloud systems. Its components as proposed in [17] are

- Event Collectors: These sensors gather logs and data from the virtual machine operating systems as well as monitor traffic which is flowing between virtual switches(vSwitches).
- Event Correlators: This component carries out the task of integrating as well as correlating the data that has been collected from event collectors. This component groups data according to the source IP and session initiation times so that HA-CIDS is able to detect any unusual activity in multiple VMs and reduce false alarms.
- Events Analyser: This component uses proprietary analysis engines to detect the host as well as network intrusion attempts. It strives to compute the probability whether the attack scenario is in fact an attack or not.
- Controller: A controller uses attack estimator component to determine the risk which the attack poses to the system and based on that information selects the most suitable response in order to protect network and host from the attacker. It takes into account, the criticality of attack, the risk level, observed system damage caused by it the prediction of its future impact and efficiency of protection method [17].

The notion of proactive intrusion uses the idea of machine where the classifier with the help of extracted data and behavioural patterns determines legitimacy of the user. The idea of a machine learning based system is to observe the behaviour of a legitimate user and use that behaviour as rule for classification of the future sessions of the user as being legitimate or not. Basically, a user whose behaviour may deviate from preconfigured and learn behavioural patterns may be termed as an attacker. However, if an adversary has an opportunity to learn the conscious behavioural traits of a legitimate user, the opponent can trick the IDS system to believe the actions performed by him/her are the actions of the actual user. This situation is even more prevalent in IaaS systems as the actual control of VM is allocated to a third party.

Consider a situation in which Alice is a legitimate user such as a database administrator and Bob is the attacker who wants to access the plans for a building stored in the database. If Bob gets hold of the Alice's credentials, then he will definitely be able to bypass the defences of the security system. Let us consider another scenario. While accessing

ISBN: 978-988-19252-7-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

the system during a session Bob is presented with a database failure error. Alice will definitely know at an instinctive level how to deal with the situation. Bob, on the other hand will not be able to deal with the situation in a way the classifier knows Alice will do. As a result, IDS will be able to detect that user session is an illegitimate one and will be able to deal with the situation. This is the main idea behind proactive intrusion detection. That is, the IDS proactively presents uncommon situations to the user and observes the user response for any deviation from normal user behaviour.

Mathematically, a user interacting with a system draws an action from a sequence of actions . Intruders who pretend to be legitimate users not only obtain the login credentials but also learn the internal model of a legitimate user's action [16]. Hence, the task of a proactive IDS is to determine whether or not the window of actions of size N was in fact performed by a legitimate user or not. Let y_i be the i^{th} task in the window and y_i^1 be the sequence (y1,y2,y3….,yi). Then IDS is a classifier which identifies whether the set of actions Y_1^N was performed by the legitimate user or not [16]. The action set is performed by the legitimate user if for a threshold θ, $P(Y_1^N|Alice) < \theta$. If the statistical probability follows a markov model then over the set of N windows, the action is a legitimate one if

$$\prod_{i=1}^{N} P(Y_1^N|Alice) < \theta$$

As mentioned before, the key idea is that the behaviour of the user must be consistent in various situations – one of which would be a normal situation while others would be simulated situations that does not occur in normal conditions. These anomalous situations are also referred to as models of operations. The idea is to observe the behaviour of the user based on actions which are taken instinctively and not consciously to verify the legitimacy of the user.

VII. Proactive Autonomous Defence Shield (PADS)

PADS strives to provide an efficient yet strong and autonomous defence strategy to combat threats which are faced by an average user on IaaS environment deployed on hybrid clouds. The focus of PADS is to shield against data mining based attacks, distributed attacks and masquerade attacks.

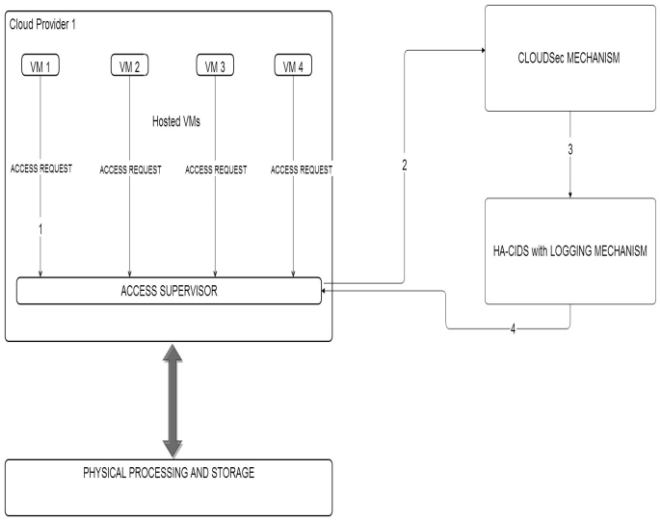

Fig 2: PADS Architecture

PADS architecture is shown in Fig: 2.It consists of the three components:

- Cloud Provider Infrastructure
- CloudSec Security Infrastructure
- HA-CIDS infrastructure with advanced logging capabilities.

This architecture shows how the control flows between components in the system.

User accesses the virtual machine through a cloud distributor. A cloud distributor randomly assigns virtual machines to the user to perform operations as well as store data. Every request is mapped to a specific VM that has been assigned to the user with the help of cloud distributor. There is no actual interaction between the cloud provider and the actual user. As soon as a VM is assigned to the user, that machine will try to access resources of the cloud provider.

Access to the physical resources such as memory, processing and computing power is controlled by Access Supervisor. A cloud provider may have multiple hypervisors in place to cater to the requests of many users. The access supervisor maintains complete control over all the hypervisors and coordinates the flow of information between them. As soon as the request for data or processing comes from any virtual machine, access supervisor maps the request to the particular hypervisor.

To map the user request for operations, the access supervisor invokes the CloudSec component. CloudSec is used to check the kernel level integrity of a virtual machine. As soon as CloudSec classifies the request for resources by a user process as a legitimate request, the control goes to HA-CIDS module which checks the VM for security breaches and distributed data compromises. As soon as HA-CIDs gives a clean bill of health to VM access request, the green signal sent to Access Supervisor authorizing VM to access the resources.

Fig 3 shows the interactions between CloudSec and hypervisor which hosts the VM that has been requested for accessing the resources residing in it. The back-end component responds to triggers which have been installed by the front end component previously as soon as VM tries to access the parts of the memory which require to be protected. As soon as the triggers are activated, the VMI back end notifies the VMI front end of the restricted access(1.2). For each new VM that is powered up, front end initiates and maintains a new VM instance using the thread pool manager(1.3). As soon as VM receives a trigger, it gets the control registers of virtual machine in order to determine the kernel layout of VM's operating system (1.4 and 1.5). Most operating systems like Linux or Windows organise the kernel data in the form of data structures and definition. This information will later be used to bridge the semantic gap. The information about kernel and CPU structure is stored in the KSD component (1.6). CloudSec then starts resolving this semantic gap with the help of SGB component (1.7). The memory pages of the VM are read into the memory Pages Buffer (MPB) by the backend as the VMI component does not have direct access to the physical memory of the VM (1.8). Once a profile is prepared by the SGB about legitimacy of a VM access request using background access paths (1.9), defence modules determine an appropriate action for the VM request(1.10) on the basis of interaction with the MPB and the memory access handler (1.12 & 1.13).

ISBN: 978-988-19252-7-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

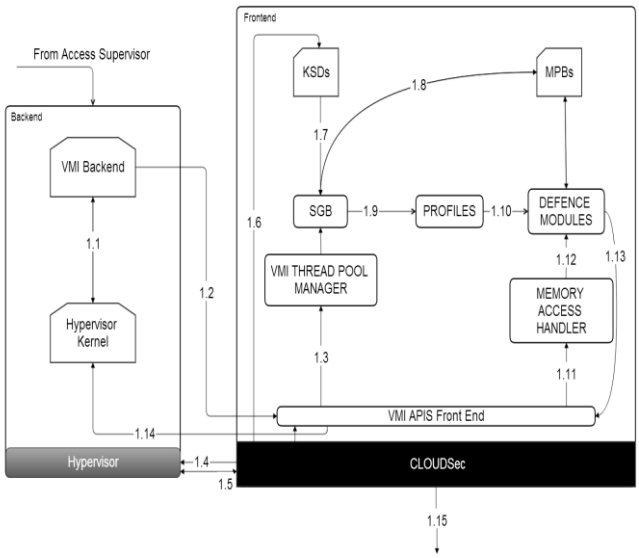

Fig 3:Interaction of Access Supervisor with CloudSec Security Mechanism

An action is then forwarded to the HA-CIDs module (1.15). The action is usually in the form of a green channel of access (i.e. VM is a legitimate user access) or red channel(i.e. VM is an attacker). HA-CIDS as explained is capable of making autonomous decisions about security events. The collection of logs from different virtual machines and their formation into standard alerts as well as general analysis is primarily used to reduce false positives and negatives. Once the system is sure enough that the scenario is actually an attack, an earning alarm is sounded at the cloud provider's infrastructure and the event is passed on to HA-CIDS.

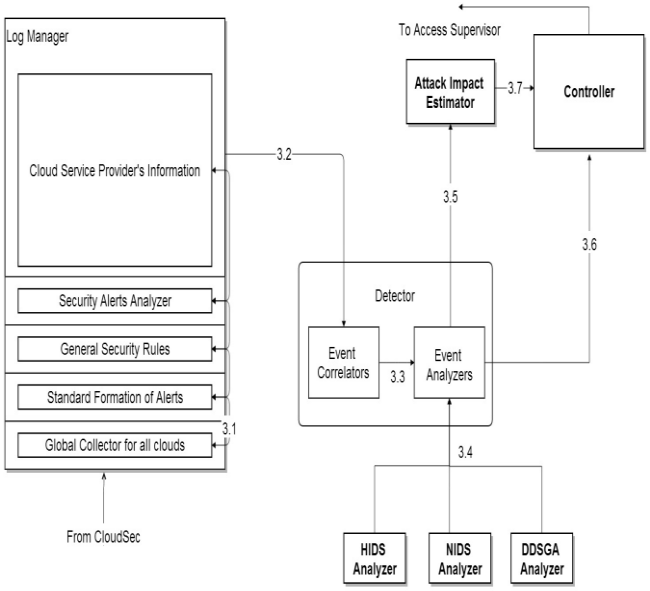

Fig 4 : Interactions between CloudSec and HA-CIDS

Fig 4 shows interaction inside HA-CIDS component. In HA-CIDS, the events from all the VMs across a single access supervisor are periodically collected. This information generally includes the services which are running on all the systems, size of virtual memory and other control information that may give an idea about the status and operational health of the VM.

These are then combined and correlated against a general set of rules which may classify the present situation as a normal one or an anomaly to normal operation. In the case of latter, the events are then analysed by analysers such as SNORT based NIDS analysers (3.2 and 3.3) to get a deeper insight into the kind of attack which is being experienced by the system(3.4). This information is then fed to the controller component which gives the access supervisor, the information about the impact of the attack on the system. This component of PADS shields against DDoS attacks which may target a single VM or multiple VMs in a system. In the event of analysers being not able to detect the attack, the system then runs the proactive analyser. As explained previously, the user is a legitimate one if

$$\prod_{i=1}^{N} P(X_1^N|\text{Alice}) < \theta$$

For every user action that goes beyond the threshold, the particular event is recorded and temporarily suspended. After analysis, the results are conveyed to the controller which then decides the severity of breach and action to be taken to control its effects. The auto-response capabilities as well as self-resilient features are inherent to HA-CIDS and hence the PADS model.

The auto response capability of this system is in regard to the ability of this system to measure the effectiveness of an attack through the attack impact estimator component [17]. In order to recover from an attack and identify the corrupted data, the system must be able to restore services in real time as soon as the attack is detected. PADS uses a detection and recovery approach based on security policies. Controller uses a policy based approach to decide in real time, the most appropriate response actions in order to first stop/contain active attacks and then take corrective actions all in real time[17].

VIII. Conclusion

The attacks countered by Proactive Autonomous Defence Shield (PADS) are by far the most common attacks on user data and operations on cloud. PADS continuously strives to improve its performance by learning legitimate user responses and proactively generating a feedback into the user process in the form of varying modes of operations to counter the threats. Since PADS deals with the some of the most prominent attacks on IaaS cloud environments, it in fact helps to uphold the service level agreements by ensuring VM security and integrity of user data on the VMs.

Accordingly, the quality of service standards although being affected by variegated factors are not affected at least by the denial-of-service threats which are prevented through a thorough event correlation analysis across distributed systems. Data mining attacks are handled in this model by distributing the data across various location and guarding against distributed, slow time paced attacks through the CDIDS component.

References

[1] Rittinghouse, John W., and James F. Ransome. Cloud computing: implementation, management, and security. CRC press, 2009.

ISBN: 978-988-19252-7-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

[2] Buyya, Rajkumar, James Broberg, and Andrzej M. Goscinski, eds. Cloud computing: Principles and paradigms. Vol. 87. John Wiley & Sons, 2010.
[3] Bouayad, A.; Blilat, A.; El Houda Mejhed, N.; El Ghazi, M., "Cloud computing: Security challenges," Information Science and Technology (CIST), 2012 Colloquium in , vol., no., pp.26,31, 22-24 Oct.2012
[4] Tianfield, H., "Security issues in cloud computing," Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on , vol., no., pp.1082,1089, 14-17 Oct. 2012.
[5] Hay, Brian; Nance, K.; Bishop, M., "Storm Clouds Rising: Security Challenges for IaaS Cloud Computing," System Sciences (HICSS), 2011 44th Hawaii International Conference on , vol., no., pp.1,7, 4-7 Jan. 2011.
[6] Dawoud, W.; Takouna, I.; Meinel, C., "Infrastructure as a service security: Challenges and solutions " Informatics and Systems (INFOS), 2010 The 7th International Conference on, vol., no., pp.1,8, 28-30 March 2010.
[7] Jensen, M.; Schwenk, J.; Gruschka, N.; Iacono, L.L., "On Technical Security Issues in Cloud Computing," Cloud Computing, 2009. CLOUD '09. IEEE International Conference on , vol., no., pp.109,116, 21-25 Sept. 2009.
[8] Ibrahim, A.S.; Hamlyn-Harris, J.; Grundy, John; Almorsy, M., "CloudSec: A security monitoring appliance for Virtual Machines in the IaaS cloud model," Network and System Security (NSS), 2011 5th International Conference on , vol., no., pp.113,120, 6-8 Sept. 2011.
[9] Tupakula, U.; Varadharajan, V.; Akku, N., "Intrusion Detection Techniques for Infrastructure as a Service Cloud," Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference on , vol., no., pp.744,751, 12-14 Dec. 2011.
[10] George W. Dunlap, Samuel T. King, Sukru Cinar, Murtaza A. Basrai, Peter M. Chen, 'ReVirt: Enabling Intrusion Analysis through Virtual-Machine Logging and Replay', Proceedings of OSDI, 2002.
[11] Garfinkel, Tal, and Mendel Rosenblum. "A Virtual Machine Introspection Based Architecture for Intrusion Detection." In NDSS. 2003.
[12] Qian Liu; Chuliang Weng; Minglu Li; Yuan Luo, "An In-VM Measuring Framework for Increasing Virtual Machine Security in Clouds," Security & Privacy, IEEE , vol.8, no.6, pp.56,62, Nov.-Dec. 2010
[13] M. Sharif et al., 'Secure In-VM Monitoring Using Hardware Virtualization', in 16th ACM CCS, ACM Press, 2009, pp. 477–487.
[14] Syed Rasheed Hassan, Julien Bourgeois, Vaidy Sunderam , Li Xiong, ' Detection of Distributed Attacks in Hybrid and Public Cloud Networks' In ICSKG, 2012, pp. 9-15.
[15] Dev, H.; Sen, T.; Basak, M.; Ali, M.E., "An Approach to Protect the Privacy of Cloud Data from Data Mining Based Attacks," High Performance Computing, Networking, Storage and Analysis (SCC), 2012 SC Companion:, vol., no., pp.1106,1115, 10-16 Nov. 2012
[16] Benjamin Liebald, Dan Roth, Neelay Shah and Vivek Srikumar, 'Proactive Intrusion Detection', in AAAI,2008.
[17] Kholidy, H.A.; Erradi, A.; Abdelwahed, S.; Baiardi, F., "HA-CIDS: A Hierarchical and Autonomous IDS for Cloud Systems," Computational Intelligence, Communication Systems and Networks (CICSyN), 2013 Fifth International Conference on , vol., no., pp.179,184, 5-7 June 2013
[18] Song Deng; Lin Wei-min; Tao Zhang; Yu Yong, "Distributed proactive defense based on cloud computing," Intelligent Computing and Integrated Systems (ICISS), 2010 International Conference on , vol., no., pp.95,98, 22-24 Oct. 2010
[19] Xue Jing; Zhang Jian-jun, "A Brief Survey on the Security Model of Cloud Computing," Distributed Computing and Applications to Business Engineering and Science (DCABES), 2010 Ninth International Symposium on , vol., no., pp.475,478, 10-12 Aug. 2010
[20] Sosinsky, Barrie. Cloud computing bible. Vol. 762. John Wiley & Sons, 2010.
[21] Behl, Akhil. "Emerging security challenges in cloud computing: An insight to cloud security challenges and their mitigation." In Information and Communication Technologies (WICT), 2011 World Congress on, pp. 217-222. IEEE, 2011.
[22] B.D.Pyne,'XenAccess', Available: http://xenaccess.org.

ISBN: 978-988-19252-7-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)