RESEARCH ARTICLE OPEN ACCESS

A Study on Invasion Detection into Deep Packet Inspection for Finding Malware

Khadarbasha.Dada [1], T Sai Prasad Reddy [2]
M.Tech (CSE) [1], Associate Professor [2]
Department of Computer Science and Engineering
Narayana Engineering College, (Affiliated to JNTUniversity, Ananthapuram)
Nellore
Andhra Pradesh - India

ABSTRACT

Cloud computing is an increasingly popular platform for both industry and consumers. The cloud presents a number of unique security issues, such as a high level of distribution or system homogeneity, which require special consideration. In an abstract terms, the cloud computing technology enable the users to access the large infrastructures and resources for high speed computing through a different middleware that are similar to existing Grid and HPC computing. In an abstract terms, the cloud computing technology enable the users to access the large infrastructures and resources for high speed computing through a different middleware that are similar to existing Grid and HPC computing. In this paper we introduce and discuss an online cloud anomaly detection approach, comprising dedicated detection components of our cloud resilience architecture. More specifically, we exhibit the applicability of novelty detection under the one-class support Vector Machine (SVM) formulation at the hypervisor level, through the utilisation of features gathered at the system and network levels of a cloud node. We demonstrate that our scheme can reach a high detection accuracy of over 90% whilst detecting various types of malware and DoS attacks. Furthermore, we evaluate the merits of considering not only system-level data, but also network-level data depending on the attack type. In this paper, malware detection and research work on these techniques are presented to get the internals of cloud security and putting the advance malware detection techniques to protect the cloud infrastructures. In a cloud network, the resources are provided to the end user in the form of virtual machines, which make them vulnerable to malware exploits, VM Escape based attacks and even distributed denial of service attacks of the resources hosted over the cloud network.

Keywords: - Cloud Computing & Security, Malwares, Anti-Virus, Resilience, Invasive Software, Multi-Agent Systems, Databases, Virtualization.

I. INTRODUCTİON

Over the past years, the cloud computing is becoming a dominant technology and widely adopted by the companies and user communities. The kind of flexibility and scalability provided by the Cloud Service Provides enable to more and more users of the technology. Instead of increasing use of cloud computing technology, it is very prone to various security risks. Cloud data centers are beginning to be used for a range of always-on services across private, public and commercial domains. These need to be secure and resilient in the face of challenges that include cyber attacks as well as component failures and mis-configurations. However, clouds have characteristics and intrinsic internal operational structures that impair the use of traditional detection systems. In particular, the range of beneficial properties offered by the cloud, such as service transparency and elasticity, introduce a number of vulnerabilities which are the outcome of its underlying virtualised nature. Moreover, an indirect problem lies with the cloud's external dependency on IP networks, where their resilience and security has been extensively studied, but nevertheless remains an issue [1] The approach taken in this paper relies on the principles and guidelines provided by an existing resilience framework [2]. The underlying assumption is that in the near future, cloud infrastructures will be increasingly subjected to novel attacks and other anomalies, for which conventional signature based detection systems will be insufficiently equipped and therefore ineffective. Moreover, the majority of current signature-based schemes employ resource intensive deep packet inspection (DPI) that relies heavily on payload information where in many cases this payload can be encrypted, thus extra decryption cost is incurred. Our proposed scheme goes beyond these limitations since its operation does not depend on a-priori attack signatures and it does not consider payload information, but rather depends on per-flow meta-statistics as derived from packet header and volumetric information (i.e. counts of packets,

bytes, etc.). Nonetheless, we argue that our scheme can synergistically operate with signature-based approaches on an online basis in scenarios were decryption is feasible and cost-effective. Overall, it is our goal to develop detection techniques that are specifically targeted at the cloud and integrate with the infrastructure itself in order to, not only detect, but also provide resilience through remediation. At the infrastructure level we consider: the elements that make up a cloud data centre, i.e. cloud nodes, which are hardware servers that run a hypervisor in order to host a number of Virtual Machines (VMs); and network infrastructure elements that provide the connectivity within the cloud and connectivity to external service users. A cloud service is provided through one or more interconnected VMs that offer access to the outside world. Cloud services can be divided into three categories based on the amount of control retained by the cloud providers. Software as a Service (SaaS) retains the most control and allows customers to access software functionality on demand, but little else. Platform as a Service (PaaS) provides customers with a choice of execution environment, development tools, etc., but not the ability to administer their own Operating System (OS). Infrastructure as a Service (IaaS) relinquishes the most control by providing customers with the ability to install and administer their own choice of OS and install and run anything on the provided virtualised hardware; as such, IaaS clouds present the most challenges in terms of maintaining a properly functioning system. Such a system would ideally be free from malware and from vulnerabilities that could lead to an attack. In order to increase the resilience of cloud infrastructures we have already defined a resilience architecture in our previous works [3], [4] that comprises anomaly detection, remediation and also coordination elements.

In this paper we discuss the detection of anomalies using a novelty detection approach that employs the one-class Support Vector Machine (SVM) algorithm and demonstrate the effectiveness of detection under different anomaly types. More specifically, we evaluate our approach using malware and Denial of Service (DoS) attacks as emulated within a controlled experimental test bed. The malware samples used are Kelihos and multiple variants of Zeus. We have selected these particular malware samples and their variants since they have been identified as posing recent and evolving threats for a range of Windows OS flavors that have already compromised more than 3.6 million machines worldwide between 2010 and 2014; mainly due to their varying and sophisticated evasion techniques, as well as their stealthy propagation1. Our contributions are as follows:

1.Experiments carried out in this work are done so in the context of an overall cloud resilience architecture under the implementation of one-class Support Vector Machines (SVMs). The resulting experimental findings show that anomalies can be effectively detected online, with minimal time cost for reasonably realistic data samples per Virtual Machine (VM), using the one-class SVM approach, with an overall accuracy of greater than 90% in most cases.

2.Our work is the first to explicitly address the aspect of malware detection in pragmatic cloud-oriented scenarios as performed by cloud providers, such as VM live-migration.

3.We provide an online novelty detection implementation that allows the adaptive SVM-specific parameter estimation for providing better detection accuracy benefits.

4.This work assesses the VM-based feature selection spectrum (i.e. system, network-based or joint datasets) with respect to the detection performance benefits on two distinct network-wise attacks (malware and DDoS) under novelty detection

II. RELATED WORK

The intrinsic properties of virtualised infrastructures (such as elasticity, dynamic resource allocation, service co-hosting and migration) make clouds attractive as service platforms. Though, at the same time they create a new set of security challenges. These have to be understood in order to better protect such systems and make them more secure. A number of studies have addressed aspects of cloud security from different viewpoints (e.g. the network, hypervisor, guest VM and Operating System (OS)) under various approaches derived either from traditional rule-based Intrusion Detection Systems (IDSs) or statistical anomaly detection models. This paper presents a cloud security solution derived from a sub-domain of anomaly detection.

A. Malware and Detection Method

One of the biggest challenges within the development of resilient and secure cloud-oriented mechanisms is related to the adequate identification and detection of malware. This is due to the fact that, in the majority of cases, malware is the first point of initiation for large-scale Distributed Denial of Service (DDoS) attacks, phishing and email spamming [3], [8], mainly through the deployment of botware. Current methods of detecting attacks on cloud infrastructures or the VMs resident within them do not sufficiently address cloud

specific issues. Despite the huge efforts employed in past studies regarding the behaviour of certain types of malware in the Internet, so far little has been done to tackle malware presence in clouds. Current methods of detecting attacks on cloud infrastructures or the VMs resident within them do not sufficiently address cloud specific issues. Despite the huge efforts employed in past studies regarding the behaviour of certain types of malware in the Internet [13], [14], so far little has been done to tackle malware presence in clouds. In particular, the studies in [15], [16] aimed to adjust the performance of traditional Intrusion Detection Systems (IDS) under signature-based techniques that employ Deep Packet Inspection (DPI) on network packets. Nevertheless, despite the important lessons learned from these studies they do not develop an overall online detection strategy that considers real-time measurement samples from each VM. Further, these approaches are purely signature based, and as such are not in a position to provide a robust scheme for any future threats posed by novel malware strains due to their simplistic rule-based nature. Each solution to detection is performed in an isolated manner and neglects to consider the unique topology of the cloud, which is at its heart a network of interconnected nodes, each with their own isolated execution environments. If a detection system is to perform effectively within a cloud it is required to possess the capability of communicating detected faults and challenges across the whole infrastructure, especially if it is to perform as part of a larger, autonomous and self-organising, cloud resilience system.

B.Virtualisation & Cloud Technologies

In [3], [8], [9] the specific security threats and challenges introduced into clouds through the use of core virtualisation technologies are discussed. Despite the end-user benefits gained by virtualisation it also comes with a range of threats that include: exploits to security holes on virtual machines (e.g. rootkit attacks on virtual machines [10]); mutated cloud-specific Internet-based attacks that aim to compromise cloud networks (e.g. malware [11], [3]; and DDoS attacks on cloud services [11]). According to [12] blackhat hackers have already identified the potential of the cloud since the instantiation, maintenance and continued operation of botnets seems to be much more effective under a cloud paradigm. In parallel, co-residence as a security concern has been explored in [10] and is the result of VMs belonging to different customers being hosted on the same cloud node. It was revealed that the outcome of co-residence is to enable shared memory attacks that, at their most benign, are capable of leaking sensitive information, and at their most destructive are capable of taking control of the entire node. Moreover, the aspect of VM migration is also a possible enabler of malicious side effects in cases where infected VMs are migrated around the cloud to a number of nodes. The cause of migration could be as a result of the provider's load balancing policy, but as an unwanted side-effect the result is to place malware in contact with a larger number of potential targets throughout the cloud infrastructure. Additionally, automation is becoming an increasingly integral part of computer system configuration through the use of dedicated tools (e.g. Ansible2) or simply by creating new VMs from clones or snapshots. This results in a collection of servers, all with the same functionality, being configured in precisely the same way. Hence, vulnerabilities and threats are being repeatedly instantiated across large portions of the cloud and malware can more easily propagate and exploit said vulnerabilities.

III. MALWARE DETECTION TECNIQUES

Since malware has different types, behaviors and different level of risk, the same detection methods and mechanisms cannot be used in all cases. It is impractical to have just one security software to efficiently handle the malwares. Hence having different detection methods for different environments becomes unavoidable. This study had focused on the most common and powerful techniques such as malicious based detection, anomaly based. The experiment added a great value to the field of malware detection since it was able to detect many malwares which were not detectable by normal detection methods, going forward, we can clearly see that the detection process needs more computer processing power and advance techniques to make sure that the nature and behavior of malware are clear and covered from all the angles and views.

A. Malicious Based Detection

Cloud computing is recognized as an alternative to traditional information technology due to its intrinsic resource-sharing and low-maintenance characteristics. In cloud computing, the cloud service providers (CSPs), such as Amazon, are able to deliver various services to cloud users with the help of powerful data centers. By migrating the local data management systems into cloud servers, users can enjoy high-quality services and save significant investments on their local infrastructures. One of the most fundamental services offered by cloud providers is data storage. However, it

also poses a significant risk to the confidentiality of those stored files. Specifically, the cloud servers managed by cloud providers are not fully trusted by users while the data files stored in the cloud may be sensitive and confidential, such as business plans and other data. To preserve data privacy, a basic solution is to encrypt data files, and then upload the encrypted data into the cloud. Unfortunately, while uploading the data malware files can also be uploaded. To detect the malware and sending the alert message using the malicious based detection. According to intrusion detection system, they suggested a detection system to expose intruders and attacks in a cloud computing environment based on the malicious method. This system which is used to check the malware files which are present in the cloud infrastructure. After finding the malware files it sends the alerts to the providers. Malware detection in cloud computing presented a model to detect malware on cloud computing integrating intrusion ontology representation using malicious based methods. This model uses multiple engine services which follows a set of defined parameters and standards for web service technologies. This model is founded on analysis with specific applications residing on the client. It can enhance their performance if they are moved to the network, where instead of running complicated software on every host, it gives each process a light to enter the system files. Then it sends them to the network to be analyzed by multiple engines and then to decide whether or not they are executed according to the report of threat delivered. This model is a multi-engine based file analysis service deployed in cloud computing, via a group of protocols and standards for web services. It is used to identify the files with malicious codes through the remote analysis by multiple engines and send the alert to the service provider.

B. Anomaly Based Detection

Anomaly-based detection looks for unexpected or abnormal behaviour indicators, which indicate the presence of malware. In more detail, anomaly based detection creates a baseline of expected operation. After this baseline has been created, any different form of baseline is recognized as malware. We have identified that the anomaly based detection technique uses the previous knowledge of what is known as normal to find out what is malicious. A special type of anomaly based detection techniques is specification based detection. A specification based detection uses set of rules to determine what is considered as normal, with the purpose of making a decision about the maliciousness of the program that breaches the rule set. The basic limitation of the specification based system technique is the difficulty to correctly determine the program or system behaviour. A number of anomaly detection techniques [21], [22], [23], [24], [25], [26] aim to proactively and reactively detect cloud-specific threats, but due to their complex statistical measures they mostly lack scalability and often require prior knowledge, thus making them unsuitable for online detection in cloud infrastructures. The work by Wang et al. [27] produced the EbAT system that allowed the online analysis of multiple metrics obtained from system-level components (e.g. CPU utilization on rack servers, memory utilization, read/write counts of the OS, etc.). The proposed system showed potential in the areas of detection accuracy and monitoring scalability, however it's evaluation did not adequately emphasise pragmatic cloud scenarios. The work in [24] provided a novel prototype that enabled an online spatio-temporal anomaly detection scheme in a cloud scenario. Thus, the authors were able to initially formulate and further implement a wavelet-based multi-scale anomaly detection system. The system relies on measured cloud performance metrics (e.g. CPU utilization, memory) gathered by multiple components (e.g. hardware, software, system) within the examined institution-wide cloud environment. The resulting experimental outcomes were quite promising since the proposed approach reached a 93:3% of sensitivity on detecting anomalous events with only just a 6:1% of the reported events to be false alarms. In particular, the authors in [30] instrumented an online adaptive anomaly detection (AAD) framework that was able to detect failures through the analysis of execution and runtime metrics using the traditional two-class Support Vector Machine (SVM) algorithm. Under a real experimentation, over a 362-node cloud computing environment in a university campus, the produced results were extremely promising since they exhibited the efficiency of the proposed scheme, which reached an overall of over 87% of anomaly detection sensitivity. However, the main issue raised by this study was that the formulation of the two-class SVM algorithm suffered from the data imbalance problem [31], which affected the training phase, and consequently led to several mis-classifications of newly tested anomalies. Moreover, in contrast with our work the proposed approach did not explicitly address the aspect of early attack detection, but rather was mainly aimed at various faults in the cloud infrastructure. Therefore, apart from providing an online anomaly detection approach, our work is also aimed at confronting an algorithmic constraint that is inherited in most of the traditional two-class on n-class Machine-Learning based techniques (e.g. two-class SVMs, Artificial Neural Networks, Bayesian Classifiers) when applied to cloud environments (e.g. [30], [32]); data imbalance. As indicated in [31], [33] a dataset is

imbalanced if the classification labels are not approximately equally represented. In simple terms, the imbalanced nature of training datasets3 invoke high classification

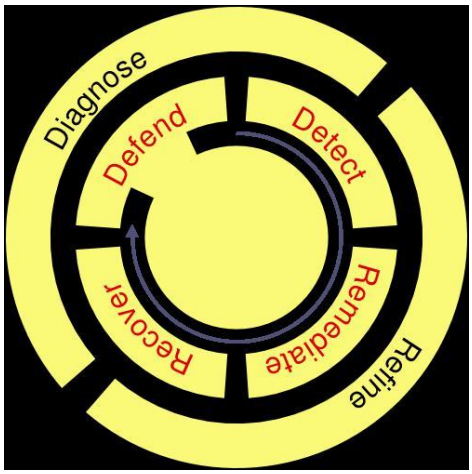

Figure 1: A high level overview of the D2R2 + DR network resilience framework [2]

errors and problematic labelling of the training instances that naturally engage high rates of misclassification throughout the testing phase of n-class classifiers (e.g. traditional SVMs). Hence, in this work we are inspired by the findings in [31], where one-class SVMs perform much better than two-class SVMs, as well as Artificial Neural Networks (ANNs), in the context of classifying DSL-level faults. Here we employ them explicitly for the detection of anomalous events in cloud environments, in particular those resulting from the execution of malware. One further reason to use one-class SVMs in the context of our research is the lack of dependence on prior knowledge regarding a particular cause for anomalous behaviour and the ability to detect new types of anomalous events as "novelties". As a result, it is possible to detect anomalies that are not well understood (i.e. no prior models) under the concept of novelty detection since they were not experienced throughout the training phase of a one-class SVM4.

IV. CLOUD RESILIENCE ARCHITECTURE

The examination presented in this paper is a piece of a bigger global research activity on system and framework versatility. It depends on the D2R2 + DR organize flexibility system [2]. This structure involves two settled methods of operation. An inward ongoing control circle involving Defending the framework, Detecting flaws and inconsistencies, Remediating against them, lastly Recovering from any recognized shortcomings. What's more, an external circle that Diagnoses shortcomings in the present setup and Refinines the general framework and flexibility technique. While the inward control circle goes for insurance progressively, the external control circle is led over a more drawn out timeframe (see Figure 1). With a specific end goal to understand the D2R2 + DR procedure, arrange what's more, framework particular versatility structures have been produced with the point of giving interoperable versatility frameworks that host the parts important to empower different versatility strategies and systems. In [4] we presented a cloud versatility design that indicates the parts through which identification and remediation in 4. For instance, in our work we prepare the classifier to name include vectors that entirely speak to typical conduct. Along these lines, malware cases, which subsequently change the factual properties of recently tried element vectors, are marked as "oddities" since they speak to deviations from the ordinary operation of the cloud. the cloud is figured it out. The strength framework is dispersed furthermore, self-sorting out, and is made out of individual programming examples, known as Cloud Resilience Managers (CRMs). Each CRM is made out of four programming parts, or motors, which are appeared in Figure 25. The product parts inside each CRM are: the Framework Analysis Engine (SAE), the Network Analysis Engine (NAE), the System Resilience Engine (SRE) and the Coordination and Organization Engine (COE). The CRM on every hub performs neighborhood inconsistency identification in light of highlights assembled from its hub's VMs and its neighborhood organize see, where those elements are taken care of by the SAE and NAE segments separately. The SRE segment is in control of remediation and recuperation activities in view of the yield from the investigation motors (i.e. the NAE and SAE), which is passed on to it by the COE. At long last, the COE segment arranges and scatters data between other examples and the segments inside its own particular hub. It is the COE that is at last accountable for the support of the associations between its CRM companions and exemplifies the self-arranging part of the general framework .Notwithstanding hub level versatility, the recognition framework is equipped for social event and breaking down information at the system segment level through the sending of system CRMs as appeared by C in Figure 2. Arrange level CRMs work in the very same way as the CRMs onveyed inside the cloud, however can watch organize activity from an interesting

advantage indicate not accessible the inward system. For instance, a CRM conveyed on an entrance/departure switch (i.e. D in the figure) can watch activity before it is firewalled, empowering it to impart important data once more into the cloud. An entrance/departure CRM is additionally ready to examine the activity from numerous hubs, permitting the nearness of a botnet to be recognized, imparted to each interior CRM, and foiled by the SREs on every hub.

Notwithstanding, the exploration displayed in this paper is worried with the online recognition segment inside the System Analysis Motor (SAE) and Network Analysis Engine (NAE), henceforth additionally insights about the general flexibility engineering can be found in [4], [3], [8]. In light of components assembled from every individual VM, the SAE and NAE are intended to uphold calculations that are fit for building models for ordinary VM operation. These are then used to pinpoint irregular occasions. In our usage, elements are removed from the virtual memory of each VM (e.g. prepare memory utilization) and also from the arrange interface of each VM and are joined to shape a highlight vector for every estimation interim. Under typical operation (i.e. with no malware injected)6 the greater part of the element vectors are joined into a preparation dataset for the one-class SVM detailing. On the other hand, under location conditions each recently observed and post-prepared component vector is tried against the preparation information with a specific end goal to decide regardless of whether it is bizarre or ordinary.

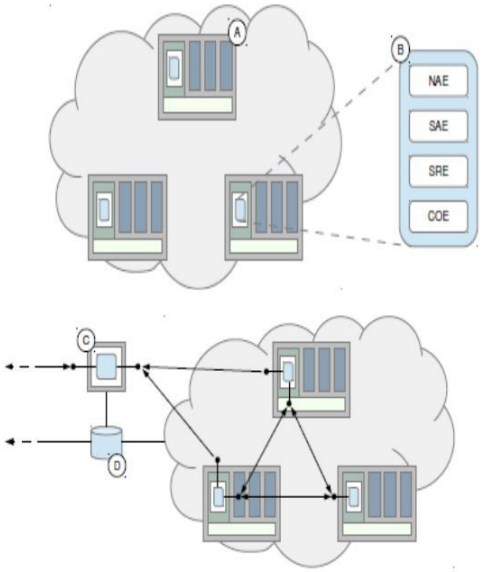

Figure 2: An overview of the detection system architecture

V .METHODOLOGY

The cloud tested utilized as a part of this work depends on KVM hypervisors under Linux (which thusly utilize Qemu for equipment imitating). The test bed involves two figure hubs, one of which additionally goes about as the capacity server for VM pictures, and a different controller server. The administration programming is Virtual Machine Manager (some of the time alluded to as virt-supervisor), which interfaces with lib virt daemons on the register hubs.

Cloud organization programming (such as Open Stack) is not considered vital for our specific tests since we are concerned exclusively with direct information obtaining from VMs and not the cooperation of the identification framework with administration programming. Be that as it may, the instruments utilized as a part of this work are perfect with any cloud coordination programming that utilizations either Xen or KVM as a hypervisor and the approach we take here could in this way be connected to such a situation. When all is said in done, our test bed is fit for a number of the capacities related with distributed computing for example, adaptable provisioning of VMs, cloning and snapshot ting VM pictures, and disconnected and online7 relocation.

A. Data Collection & Feature Extraction

Dataset is accomplished through the checking of a VM that has been made from a known-to-be-perfect plate picture. Each VM preview that is gathered is put away in a solitary document that speaks to the typical conduct of that VM picture. At 8 second interims the Volatility apparatus is conjured with our custom module that creeps VM memory for each inhabitant procedure structure. From each procedure we extricate the accompanying crude highlights per handle

- memory usage (i.e. actual size of the process in memory)
- peak memory usage (i.e. the requested memory allocation)
- number of handles (resources the process has open, e.g. files)

At the network level the NAE gathers data through tcp dump, which separates packets into 8 second time bins. Features are then extracted using the CAIDA Coral Reef suite of tools, which provides the capability to generate statistics per unidirectional TCP and UDP flow. The raw features include:

- **packets per address pair**
- **bytes per address pair**
- **flows per address pair**

B. One-Class SVM

The center of our online identification approach inside the SAE and NAE lies with the execution of the directedone-class SVM calculation, which is an expansion of conventional two-class SVM, and was proposed by Scholkopf et al. in [35]. By and by, the one-class SVM detailing handles cases utilizing unlabelled information (i.e. oddity discovery), the principle objective of which is to deliver a choice capacity that can give back a class vector y given an info network x in view of the dispersion of a preparation dataset. The class y is a parallel class where one result is the known class, which for our situation is the typical VM conduct, what's more, the other is the novel class, which speaks to any testing occasions that are obscure to the classifier. On the off chance that we let x = (x1; x2; : ; xn?1; xn) speak to an element vector, which contains the greater part of the VM-related components portrayed prior (area 3.1), then the choice capacity f(x) takes the shape:

$$f(x) = \sum_{i=1}^{N} \alpha_i k(x, x_i) - \rho \qquad (1)$$

However, in order to achieve f(x) and attain the i multiplier over the kernel function k(x; xi) it is firstly required to solve the optimisation problem in Equation 2 using lagrange multipliers, as follows:

$$\begin{aligned} &\min_{w,\xi_i,\rho} \tfrac{1}{2}\|w\|^2 + \tfrac{1}{\nu n}\textstyle\sum_{i=1}^{n}\xi_n - \rho \\ &\text{subject to:} \\ &(w \cdot \phi(x_i)) \geq \rho - \xi_i \quad \text{for all } i = 1, \ldots, n \\ &\xi_i \geq 0 \quad \text{for all } i = 1, \ldots, n \end{aligned} \qquad (2)$$

C.SAE & NAE One-Class SVM Tuning

Before the preparation procedure, the SAE and NAE motors automatically transform the underlying assembled dataset by scaling them towards a Gaussian conveyance. These is because of requirement of the RBF part that the information be centred on zero and have unit difference. Consequently the tuning process embedded in the SAE and NAE expels the mean from each highlight and partitions the component vector by the standard deviation. The preparation procedure in this way includes passing the scaled preparing dataset as a contribution to the one-class SVM calculation, which creates a choice capacity that is ready to arrange new component vectors. When all is said in done, the preparation procedure is dictated by four components: the size and substance of the preparation dataset and the two parameters _ and . The preparation dataset size is resolved by the time span over which VM checking is directed, after which it is conceivable to choose subsets of the accessible information bringing about a refinement of preparing information furthermore, a decrease in dataset estimate if required. Dataset substance is dictated by the conduct of the procedures in the VM and is not precisely controllable, henceforth the main impact that can be forced on the information is by differing the applications also, the heaps on each of them. Interestingly, the parameters and can be finely controlled and are picked at preparing time to adjust the exactness of the classifier regarding the accessible preparing information.

D.SAE & NAE Online Detection Process

As described in the previous subsections, the one-class SVM classifier within our SAE and NAE implementation is trained to identify anomalies by training it on a dataset of normal VM behaviour. This is embodied in a dataset comprising features obtained during normal operation and is used to generate a decision function that is capable of classifying novel samples (i.e. anomalous behaviour). Once trained, the classifier operates on feature vectors in an online capacity in order to produce a classification in real-time. The evaluation of the classifier within the SAE is conducted experimentally through the following procedure:

A clean VM is created from a known-to-be-clean disk image. The VM is monitored for a period of 10 minutes in what we refer to as the "normal phase". Malware is injected and a further 10 minutes of monitoring follows in what we refer to as the "anomalous phase".

$$FPR = \frac{FP}{FP+TN}$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

$$|Precision = \frac{TP}{TP+FP}$$

$$Recall = \frac{TP}{TP+FN}$$

$$F\ score = 2 \times \left(\frac{Precision \times Recall}{Precision + Recall}\right)$$

$$G\ mean = \sqrt{Precision \times Recall}$$

Accuracy is the degree to which the detector classifies any newly tested data samples correctly whereas

precision is a measure of how many of the positive classifications are correct, i.e. the probability that a detected anomaly has been correctly classified. The recall metric is a measure of the detector's ability to correctly identify an anomaly, i.e. the probability that an anomalous sample will be correctly detected. The final two metrics are the harmonic mean (F score) and geometric mean (G mean), which provide a more rounded measure of the performance of a particular detector by accounting for all of the outcomes to some degree.

E.Classification Performance Metrics

The recognition execution of the classifier can be evaluated by deciding the contrast between the class it produces for a given information and the class it ought to create. For case, if a specimen of information contains no peculiarities due to a malware strain, and the classifier creates a yield of 1 for that information point, it is a right characterization. In request to measure the grouping execution we counsel a perplexity framework that depicts every single conceivable result of a forecast and has the shape:

		Actual Class	
		1	−1
Predicted Class	1	TN	FN
	−1	FP	TP

F.EXPERIMENTAL SCENARIOS & MALWARE DESCRIPTION

1. Malware Analysis on Static VMs

An underlying worry of any cloud supplier ought to be the part of VM screening; the way toward profiling the framework also, organize elements of a running VM and hence affirming that it is not tainted with malware. Along these lines, our to start with investigation as showed by means of Figure 3 used the test bed setup depicted before and expected to assess our screening procedure by infusing malware and furthermore imitating a DDoS assault (as portrayed in area 5.6) on a given VM. The VM in our experimentation has a straightforward web server that gives a HTTP administration to numerous customer demands. The examination went on for 20 minutes, with malware infusion (utilizing Kelihos and Zeus malware strains separately) on the tenth moment. With a specific end goal to create a few sensible foundation activity we built up some custom scripts on different has inside a similar LAN that empowered the arbitrary era of HTTP solicitations to the objective server14. The decision of HTTP for movement era is run of the mill of numerous cloud servers that host web servers or related REST based pplications. Likewise, these sorts of server are among the most focused by malware because of them being exceptionally open confronting, and hence require the most observing.

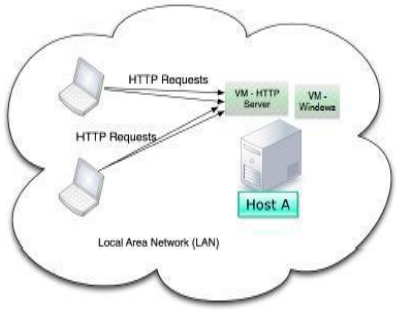

Figure 3: Visualization for the experimental setup for static malware analysis.

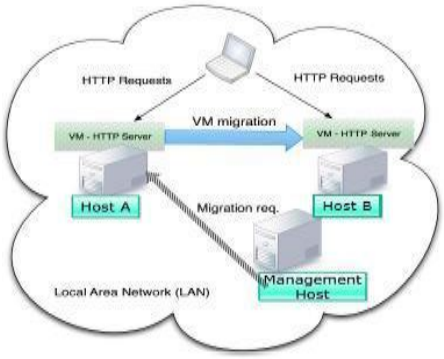

Figure4:Visualization for the experimental setup for malware analysis under VM migration.

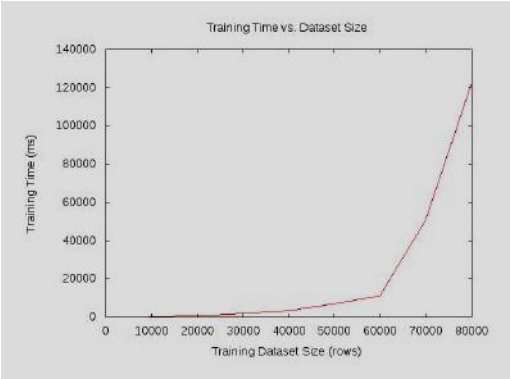

Figure. 5. Time taken to train the classifier vs. training dataset size

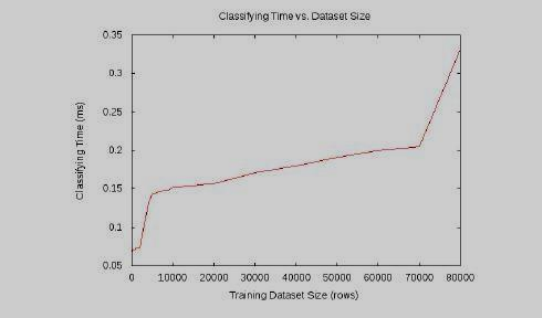

Figure 6: Time taken to output a class vs. training dataset size

2.Malware Analysis During Live-Migration

Cloud suppliers are additionally vigorously worried with the security suggestions related with the situation of VM/administration movement starting with one physical host then onto the next. Along these lines, in this work we have expressly focused on live movement for experimentation, since the best greater part of business cloud administration programming (e.g. VMWare VSphere15) utilize this usefulness of course. Accordingly, the targets of our second investigation were: to firstly figure out if malware occupant on a tainted VM would remain operational post-relocation; furthermore, we expected to address the real recognition of the malware from information accumulated at the hypervisor level of the hubs that facilitated the VM.

2. Malware Samples

In particular, the Kelihos malware spawns many child processes and subsequently exits from its main process. This is likely an obfuscation method to avoid detection, but has the effect of skewing system level features resulting in an obvious anomaly. The main purposes of these child processes are to monitor user activity and contact a Command and Control server (C&C) in order to join a botnet. At the same time, the Zeus malware and its variants, exhibit obfuscation techniques that tamper with security software installed on a given host. Its first action is to inject itself into one of the main system processes and to subsequently disable antivirus and security center applications. This behaviour leads to any attempt to detect it from within the OS futile and makes detection systems that exist outside the execution environment of the malware (such as the method used in this work) particularly applicable.The choice of Windows as the subject of experimentation is largely due to the fact that a range of IaaS clouds do demonstrate a higher need for Windows-based VMs as mentioned by cloud operators within the IU-ATC project [34]. In addition, most of the malware available in binary form have been compiled as Windows executable, thus we chose compatible target on which to unleash them.

VI. CONCLUSION AND FUTURE WORK

An online anomaly detection method that can be applied at the hypervisor level of the cloud infrastructure. The method is embodied by a resilience architecture that was initially defined in, further explored in and which comprises the System Analysis Engine (SAE) and Network Analysis Engine (NAE) components. These exist as submodules of the architecture's Cloud Resilience Managers (CRMs), which perform detection at the end-system, and in the network respectively. Our evaluation focused on detecting anomalies as produced by a variety of malware strains from the Kelihos and Zeus samples under the formulation of a novelty detector that employs the one-class Support Vector Machine (SVM) algorithm.

Moreover, in order to empower the generic properties of our detection approach we also assess the detection. Online anomaly detection under two pragmatic cloud scenarios, based on suggestions by cloud operators, which emulate "static" detection as well as detection under the scenario of VM "live" migration. The results obtained by strictly utilizing system-level data in our SAE detection, which was supported by an automatic SVM-specific parameter selection process, have shown excellent detection for all samples of malware under a variety of conditions (i.e. static and migration analysis) with an overall detection accuracy rate of well above Hence, demonstrate that the extracted features for classifier training were appropriate for our purposes and aided towards the detection of the investigated anomalies under minimal time cost throughout the training and testing phase. Nonetheless, in order to further the investigation, this feature set can easily be expanded to include statistics derived from usage and a deeper introspection of process handles, which could be beneficial for the detection of highly stealthy malware.

AUTHOR'S PROFILE

First Author**: Mr.Dada KhadarBasha** has received M.Sc review degree in Computer Science (CS) in the year 2016 and Pursuing M.Tech in Computer Science and Engineering(CSE) from Narayana College of Engineering (Affiliated to JNTUniversity, Ananthapuram), Nellore, Andhra Pradesh, India.

Second Author: **Associate Prof. T Sai Prasad Reddy**

REFERENCES

[1] A. Marnerides, C. James, A. Schaeffer, S. Sait, A. Mauthe,and H. Murthy, "Multi-level network resilience: Traffic analysis,anomaly detection and simulation," ICTACT Journal on Communication Technology, Special Issue on Next GenerationWireless Networks and Applications, vol. 2, pp. 345–356, June2011.

[2] J. P. G. Sterbenz, D. Hutchison, E. K. C¸ etinkaya, A. Jabbar, J. P.Rohrer, M. Sch¨ oller, and P. Smith, "Resilience and survivabilityin communication networks: Strategies, principles, and survey ofdisciplines," Comput. Netw., vol. 54, no. 8, pp. 1245--1265, Jun.2010. [Online]. Available:http://dx.doi.org/10.1016/j.comnet.2010.03.005

[3] A. K. Marnerides, M. R. Watson, N. Shirazi, A. Mauthe, and D. Hutchison, "Malware analysis in cloud computing: Networkand system characteristics," IEEE Globecom 2013, 2013.

[4] M. R. Watson, N. Shirazi, A. K. Marnerides, A. Mauthe, and D. Hutchison, "Towards a distributed, self-organizing approach tomalware detection in cloud computing," 7th IFIP/IFISC IWSOS,2013.

[5] M. Garnaeva, "Kelihos/Hlux Botnet Returns with New Techniques."Securelist http://www.securelist.com/en/blog/655/Kelihos Hlux botnet returns with new techniques.

[6] H. Binsalleeh, T. Ormerod, A. Boukhtouta, P. Sinha, A. Youssef, M. Debbabi, and L. Wang, "On the analysis of the zeus botnetcrimeware toolkit," in Privacy Security and Trust (PST), 2010 Eighth Annual International Conference on, Aug 2010, pp. 31–38.

[7] T. Brewster, "Game Over Zeus returns: thieving malware rises a month after Police actions," Guardian Newspaper, 11, July, 2014, http://www.theguardian.com/technology/2014/jul/11/ga meover-zeus-criminal-malware-police-hacking.

[8] A. K. Marnerides, P. Spachos, P. Chatzimisios, and A. Mauthe,"Malware detection in the cloud under ensembleempiricalmodeldecomposition,"in Proceedings of the 6th IEEE InternationalConference on Networking and Computing, 2015.

[9] L. Kaufman, "Data security in the world of cloud computing,"Security Privacy, IEEE, vol. 7, no. 4, pp. 61–64, July 2009.

[10] M. Christodorescu, R. Sailer, D. L. Schales, D. Sgandurra,and D. Zamboni, "Cloud security is not (just) virtualization security: A short paper," in Proceedings of the 2009 ACM Workshop on Cloud Computing Security, ser. CCSW '09. New York, NY, USA: ACM,2009, pp. 97–102. [Online]. Available http://doi.acm.org/10.1145/1655008.1655022

[11] N. Gruschka and M. Jensen, "Attack surfaces: A taxonomy forattacks on cloud services," in Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on, July 2010, pp. 276–279.Y. Chen, V. Paxson, and R. H. Katz, "Whats new about cloudcomputing security?" EECS Department, University of California,Berkeley, Tech. Rep. UCB/EECS-2010-5, Jan 2010. [Online].Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/2010/ EECS-2010-5.html

[12] G. Gu, P. Porras, V. Yegneswaran, M. Fong, and W. Lee,"Bothunter: Detecting malware infection through ids-driven dialog correlation," in Proceedings of 16t USENIX Security Symposium on USENIX Security Symposium, ser. SS'07.Berkeley, CA, USA: USENIX Association, 2007, pp. 12:1–12:16. [Online]. Available:http://dl.acm.org/citation.cfm?id=1362903.13 62915

[13] M. Bailey, J. Oberheide, J. Andersen, Z. Mao, F.Jahanian, andJ. Nazario, "Automated classification and analysis of internetmalware," in Recent Advances in Intrusion Detection, ser.Lecture Notes in Computer Science, C. Kruegel, R. Lippmann,and A. Clark, Eds. Springer Berlin Heidelberg, 2007, vol.4637, pp. 178– 197. [Online].Available:http://dx.doi.org/10.1007/978-3-540-74320-0 10

[14] C. Mazzariello, R. Bifulco, and R. Canonico, "Integrating a network ids into an open source cloud computing environment,"in Information Assurance and Security (IAS), 2010 Sixth International Conference on, Aug 2010, pp. 265–270.

[15] S. Roschke, F. Cheng, and C. Meinel, "Intrusion detection in the cloud," in Dependable, Autonomic and Secure Computing, 2009.DASC '09. Eighth IEE International Conference on, Dec 2009, pp.729–734.

[16] A. Ibrahim, J. Hamlyn-Harris, J. Grundy, and M. Almorsy, "Cloudsec: A security monitoring appliance for virtual machines in the iaas cloud model," in Network and System Security (NSS),

2011 5th International Conference on, Sept 2011, pp. 113–120.

[17] B. Hay and K. Nance, "Forensics examination of volatile system data using virtual introspection," SIGOPS Oper. Syst. Rev., vol. 42, no. 3, pp. 74–82, Apr. 2008. [Online]. Available: http://doi.acm.org/10.1145/1368506.1368517

[18] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: Asurvey," ACM Computing Survey (CSUR), vol. 41, no. 3, p. 15,2009.

[19] A. Marnerides, A. Schaeffer-Filho, and A. Mauthe, "Traffic anomaly diagnosis in internet backbone networks: a survey,"Computer Networks, vol. 73, pp. 224–243, 2014.

[20] C. Wang, K. Viswanathan, L. Choudur, V. Talwar, W. Satterfield,and K. Schwan, "Statistical techniques for online anomaly detection in data centers," in Integrated Network Management (IM), 2011 IFIP/IEEE International Symposium on. IEEE, 2011, pp.385–392.

[22] C. Wang, V. Talwar, K. Schwan, and P. Ranganathan, "Online detection of utility cloud anomalies using metric distributions," in Network Operations and Management Symposium (NOMS), 2010 IEEE. IEEE, 2010, pp. 96–103. 1545-5971 (c) 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. Seehttp://www.ieee.org/publications_standards/publicat ions/rights/index.html for more information This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2015.2457918, IEEE Transactions on Dependable and Secure Computing 14

[23] Q. Guan and S. Fu, "Adaptive anomaly identification by exploring metric subspace in cloud computing infrastructures," in Reliable Distributed Systems (SRDS), 2013 IEEE 32nd International Symposium on. IEEE, 2013, pp. 205–214.

[24] Q. Guan, S. Fu, N. DeBardeleben, and S. Blanchard, "Exploringtime and frequency domains for accurate and automated anomaly detection in cloud computing systems," in Dependable Computing (PRDC), 2013 IEEE 19th Pacific Rim International Symposium on. IEEE, 2013, pp. 196–205.

[25] I. Cohen, J. S. Chase, M. Goldszmidt, T. Kelly, and J. Symons, "Correlating instrumentation data to system states: A building block for automated diagnosis and control." in OSDI, vol. 4, 2004, pp. 16–16.

[26] P. Bahl, R. Chandra, A. Greenberg, S. Kandula, D. A. Maltz, and M. Zhang, "Towards highly reliable enterprise network services via inference of multi-level dependencies," in ACM SIGCOMM Computer Communication Review, vol. 37, no. 4. ACM, 2007, pp. 13–24.

[27] C.Wang, "Ebat: online methods for detecting utility cloud anomalies," in Proceedings of the 6th Middleware Doctoral Symposium. ACM, 2009, p. 4.

[28] Y. Guan and J. Bao, "A cp intrusion detection strategy on cloud computing," in International Symposium on Web Information Systems and Applications (WISA), 2009, pp. 84–87.

[29] J.-H. Lee, M.-W. Park, J.-H. Eom, and T.-M. Chung, "Multi-level intrusion detection system and log management in cloud computing," in Advanced Communication Technology (ICACT), 2011 13th International Conference on. IEEE, 2011, pp. 552–555.

[30] H. S. Pannu, J. Liu, and S. Fu, "Aad: Adaptive anomaly detection system for cloud computing infrastructures," Reliable Distributed Systems, IEEE Symposium on, vol. 0, pp. 396–397, 2012.

[31] A. Marnerides, S. Malinowski, R. Morla, and H. Kim, "Fault diagnosis in fDSLg networks using support vector machines," Computer Communications, no. 0, pp. – , 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0140366415000080

[32] W.-H. Chen, S.-H. Hsu, and H.-P. Shen, "Application of svm and ann for intrusion detection," Computers & Operations Research vol. 32, no. 10, pp. 2617–2634, 2005.

[33] Y. Tang, Y.-Q. Zhang, N. Chawla, and S. Krasser, "Svms modelling for highly imbalanced classification,"Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, vol. 39, no. 1, pp. 281–288, Feb 2009.

[34] "India - uk advanced technology centre project," http://www. iu-atc.com/.

[35] B. Sch¨olkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, "Support vector method for novelty detection." in NIPS, vol. 12, 1999, pp. 582–588.

[36] A. Marnerides, M. Watson, N. Shirazi, A. Mauthe, and D. Hutchison, "Malware analysis in cloud computing: Network and system characteristics," in GlobecomWorkshops (GC Wkshps), 2013 IEEE, Dec 2013, pp. 482–487.

[37] N.-U.-H. Shirazi, S. Simpson, A. Marnerides, M. Watson, A. Mauthe, and D. Hutchison, "Assessing the impact of intra-cloud live migration on anomaly detection," in Cloud Networking (CloudNet), 2014 IEEE 3rd International Conference on, Oct 2014, pp. 52–57.