Intrusion Prevention System for Cloud Environment

Vimmi Pandey
PhD Research Scholar, Department of Computer Science, Rani Durgavati University, Jabalpur, India

ABSTRACT: Cloud is becoming a dominant computing platform. Naturally, a question that arises is whether we can beat notorious DDoS attacks in a cloud environment. Researchers have demonstrated that the essential issue of DDoS attack and defence is resource competition between defenders and attackers. A cloud usually possesses profound resources and has full control and dynamic allocation capability of its resources. Therefore, cloud offers us the potential to overcome DDoS attacks. However, individual cloud hosted servers are still vulnerable to DDoS attacks if they still run in the traditional way. In this paper, we propose a dynamic resource allocation strategy to counter DDoS attacks against individual cloud customers. When a DDoS attack occurs, we employ the idle resources of the cloud to clone sufficient intrusion prevention servers for the victim in order to quickly filter out attack packets and guarantee the quality of the service for benign users simultaneously. We establish a mathematical model to approximate the needs of our resource investment based on queuing theory. Through careful system analysis and real-world data set experiments, we conclude that we can defeat DDoS attacks in a cloud environment.

I. INTRODUCTION

Over the last decade, our society has become technology dependent. People rely on computer networks to receive news, stock prices, email and online shopping. The integrity and availability of all these systems need to be defended against a number of threats. Amateur hackers, rival corporations, terrorists and even foreign governments have the motive and capability to carry out sophisticated attacks against computer systems [1]. Therefore, the field of information security has become vital to the safety and economic well being of society as a whole. The rapid growth and widespread use of electronic data processing and electronic business conducted through the massive use of the wired and wireless communication networks, Internet, Web application, cloud computing along with numerous occurrences of international terrorism, raises the need for providing secure and safe information security systems through the use of firewalls, intrusion detection and prevention systems, encryption, authentication and other hardware and software solutions. In this struggle to secure our stored data and the systems, IDPS can prove to be an invaluable tool, where its goal is to perform early detection of malicious activity and possibly prevent more serious damage to the protected systems [2]. By using IDPS, one can potentially identify an attack and notify appropriate personnel immediately or prevent it from succeeding, so that the threat can be contained. IDPS can also be a very useful tool for recording forensic evidence that may be used in legal proceedings if the perpetrator of a criminal breach is prosecuted [3]. However, IDPS performance is hindered by the high false alarm rate it produces [4]. This is a serious concern in information security because any false alarms will onset a severe impact to the system such as the disruption of information availability because of IDPS blockage in suspecting the information to be an attack attempt.

This paper mostly focuses on server virtualization and on protection from intrusions at the virtual level, meaning securing the user VMs. This is an emerging area for security monitoring services, going a step beyond traditional intrusion monitoring for physical infrastructures, namely Intrusion Detection and Prevention Services (IDPS). A wide variety of existing approaches are applicable [4], ranging from monitoring intrusions from the network (network-based security), from the system (host-based security), or from the virtualization layer (hypervisor-based security). Similarly, protection may be operated by different stakeholders: VM user, cloud provider, or third parties. Hence the need to get a clearer picture of the domain to identify the most promising designs.

The objectives of this paper are to: (1) give an overview of existing techniques for building intrusion monitoring services, and explain their limitations for protecting the virtual layer to handle the change of paradigm induced by the cloud, namely scalability, dynamicity, multi-tenancy, and virtualization attributes; (2) present the emerging monitoring
approach based on the hypervisor, together with a review of the associated monitoring architectures and their benefits; (3) provide an assessment of the main IDPS architectural approaches depending on which stakeholder provides intrusion monitoring services.

Based on this evaluation, we make a position statement that network- and host-based security monitoring are not suitable for the cloud, and that IDPS should be placed at the hypervisor level, as that choice provides many security benefits compared to other designs. This analysis also highlights the privileged role the cloud provider may play to operate such type of IDPS: the cloud provider, both as provider of infrastructure and of security services, is then able to strengthen the security of the virtual layer thanks to integrated security monitoring.

The fully distributed and open structure of cloud computing and services becomes an even more attractive target for potential intruders. It involves multi-mesh distributed and service oriented paradigms, multi-tenancies, multi-domains, and multi-user autonomous administrative infrastructures which are more vulnerable and prone to security risks. Cloud computing service architecture combines three layers of interdependent infrastructure, platform and application; each layer may suffer from certain vulnerabilities which are introduced by different programming or configuration errors of the user or the service provider.

The remainder of the paper is organized as follows. Section II provides background on IaaS infrastructure security and IDPS techniques. Section III analyzes intrusion monitoring requirements derived from the cloud attributes. Section IV

1 Throughout the paper, the acronym IDPS is used whenever the discussion applies to both detection and prevention. Otherwise, IDS or IPS are respectively used when examining specifically detection or prevention.

![Fig. 1. Elements of a IaaS Infrastructure.](image)

reviews the limitations of traditional IDPS techniques when applied to the cloud setting. Section V presents the hypervisor-based monitoring approach enabling to meet cloud protection requirements. Section VI provides an overview of how those principles are applied in current systems with an assessment of the main architectural approaches. Finally, Section VII presents some future directions, notably discussing the most promising technical approach available to a cloud provider to monitor intrusions in IaaS clouds.

II. BACKGROUND ON IAAS AND THREAT MONITORING

A. IaaS Infrastructure

This paper focuses on the Infrastructure-as-a-Service (IaaS) service delivery model [5]. A IaaS infrastructure lets users rent computing, networking, and storage resources. They may then deploy and run arbitrary software (including
their own OS). Hardware and low-level software management remain in the provider’s hands. One of the most emblematic IaaS infrastructures is Amazon’s Elastic Compute Cloud (EC2) platform [6].

IaaS is the foundation of most cloud services, including: Platform-as-a-Service (PaaS) which lets users deploy in the cloud their own applications, created using development en-vironments and tools supported by the cloud provider; and Software-as-a-Service (SaaS) which gives users access to the provider’s applications running in the cloud. Therefore, advances in IaaS security monitoring will also benefit to PaaS and SaaS services.

A IaaS cloud is organized in three main software and hardware layers, as shown in Figure 1.

The virtual layer is composed of multiple instances of a software stack (applications, middleware, and OS) in the form of Virtual Machines (VMs). Among these VMs, the user VMs contain the user code and data which may be security sensitive. The various VMs belonging to a given user viewed as a whole is often called the user workload. It may be seen as a virtual and distributed IT environment running on top of the cloud infrastructure under the control of the cloud provider.

The virtualization layer enables to deploy and run con-currently the VMs on a same physical host, through virtualization [7]. This task is performed by a specific component, the hypervisor (also named VMM, for Virtual Machine Mon-itor), which manages the allocation of the physical machine resources among VM instances. It notably guarantees strict isolation between VMs. The hypervisor can also establish virtual networks (vNetwork) for the VMs to communicate through a low-level software bus, the virtual switch (vSwitch).

The physical layer consists of computing (CPU, memory), networking, and storage resources distributed over the cloud infrastructure. Those hardware resources are virtualized and shared among VMs by the virtualization layer.

B. Threats

In the cloud, some of the key security concerns are: multi-tenancy, resources being shared among multiple tenants with possibly conflicting interests; new vulnerabilities introduced by virtualization; dissolved organization boundaries, making it more difficult to define the “inside” and the “outside” of a network, and to place counter-measures to guarantee perimeter security – and more generally resource isolation; and loss of control over outsourced applications and data, resources running, moving, and being stored in shared environments.

For a IaaS infrastructure, those concerns materialize by a wide array of threats, described next.

a) VM-to-VM attacks: A malicious VM may attempt to fool the IaaS VM placement strategy to run on the same physical machine as the attack target (another VM). It may then take advantage of a flaw in hypervisor isolation to launch a side-channel attack to steal / corrupt information from the target VM [8], [9].

b) Hypervisor subversion: More potent attacks attempt to take control of the hypervisor itself from a malicious VM (hyperjacking). Such attacks undermine the commonly estab-lished idea that commodity hypervisors have a relatively low surface of attack. A VM is able to “escape” from hypervisor isolation enforcement to take full control of the virtualization layer. Possible attack vectors include misconfigurations, and malicious or poorly confined device drivers in the hypervisor. Possible subsequent steps include compromising hypervisor integrity, installing rootkits, or launching an attack against another VM, resulting in breaches in confidentiality, integrity, or availability (e.g., Denial of Service). In the past few years, such isolation breakout attacks have been published for nearly all main hypervisors [10], [11], [12], [13].

c) Network threats: Traditional network security threats such as traffic snooping (intercepting network traffic), address spoofing (forging VM MAC or IP addresses), or VLAN hop-ping (breaking out of traffic segregation) are also possible, either between physical hosts, or between VMs on the same host.
d) **Availability threats:** This is also a major issue due to resource sharing. Faulty or malicious VM behavior may lead to resource starvation, interruption of service or unexpected VM migrations. The cloud platform may be brought to a halt, hosts running out of memory, or VMs becoming momentarily inaccessible due to greedy VM behaviours. Such events may greatly alter the cloud provider image [14]. Crime ware-as-a-Service scenarios may be even worse: the large amount of cloud resources is then deliberately used by hackers to launch massive attacks against juicy targets, e.g., using botnets or other malwares.

e) **Information security threats:** Violations of confidentiality or integrity should also be taken into account for stored data. Few techniques are available to address those threats specifically, apart from traditional cryptographic counter-measures (encryption, signature, and authentication).

This paper focuses on attacks that may be detected when monitoring access and use of resources (computing, network-ing and storage) allocated to and manipulated by VMs.

C. **Threat Monitoring**

To mitigate the previous threats, it is necessary to detect and prevent intrusions in the IaaS infrastructure as early as possible. This goal may be achieved using *Intrusion Detection Services* (IDS) and *Intrusion Prevention Services* (IPS) [15].

The goal of an IDS is “to monitor an information system and analyze its behaviour to detect attacks” [16]. Thus, the infrastructure security health may be assessed. Two broad families of methodologies are available to detect incidents, both known and unknown [17].

*Misuse-based detection* (also called *knowledge-based* or *signature-based* detection) uses knowledge accumulated about attacks to look for evidence of exploitation of a vulnerability. Misuse-based detection is very effective at detecting known threats by comparing attack signatures against observed events. With this method, the attack context is fully understood, and may be used to facilitate decision-making to trigger counter-measures. However, signature definition and matching may be expensive to perform.

*Anomaly-based detection* (also called *behaviour-based* detection) builds a reference model of the usual behaviour of the monitored system (e.g., in terms of bandwidth consumption, Number of failed logins, processor usage level...) and looks for deviations from this model. Anomaly-based detection is effective at detecting previously unknown threats whenever attacks produce a deviation from the model of normal activity. However, the root cause of the intrusion remains hard to find.

IDS operation usually goes through the following phases:

- **A detection phase:** information related to the monitored system activity is collected through sensors, and correlated to derive an overall picture of the danger level. Alerts are raised whenever abnormal activity or known attack patterns are detected.

- **A decision phase:** alerts are analyzed and checked against the system security policy. In case of policy violation, this phase may trigger the launching of counter-measures in the infrastructure.

- **A reaction phase** (optional): counter-measures selected during the decision phase are activated to mitigate the detected attack. Whenever this phase is present, one rather speaks of IPS than IDS.

Two main types of IDS are traditionally distinguished, de-tecting intrusions either from network or system perspectives.
Network-based Intrusion Detection Services (NIDS) monitor and analyze network traffic. Sensors are distributed throughout the network, to avoid a single point of failure. This approach is particularly efficient to detect network attacks such as (Distributed) Denial of Service events. It is weakly vulnerable to attacks because security mechanisms are isolated from the execution context of the resource to protect.

Host-based Intrusion Detection Services (HIDS) monitor, correlate, and analyze event information on each host regarding the use of its resources. This approach has the advantage of being based on very precise audit data, facilitating knowledge of the attack context. Unfortunately, monitoring and monitored system reside on the same host which may have security and performance impacts.

For detecting intrusions in the cloud, a hard issue is the placement of security mechanisms in the infrastructure, i.e., for detection, decision, and reaction components. The placement issue may be addressed horizontally, i.e., in which part of the security architecture (network or host) are protection mechanisms most relevant? It may also be considered vertically, i.e., in which infrastructure layer are they most effective?

Traditionally, horizontal placement between network and host was solved by combining NIDS and HIDS methods. Vertical placement is an issue that was magnified with virtualization, a choice becoming possible between virtual, virtualization, and physical layers to set the IDS components. This choice depends on the stakeholders (e.g., cloud provider or customer) operating the layers. It may also vary according to their objectives regarding cloud infrastructure security man-agreement, and on the level of openness provided to other parties for layer resource management. It also highly dependent on the cloud model, which provides more or less openness, and where responsibilities for security management may be different between parties.

For the IaaS model, it is commonly reported that security is mostly a customer responsibility. The user must protect his VMs adequately, once isolation between VMs is considered as guaranteed by the provider. In this paper, we consider and argue that this claim must be qualified: to guarantee VM security, the provider must also get involved in security monitoring of the virtual layer. With this goal in mind, we now present the architectural requirements for a security monitoring infrastructure due to cloud specificities.

III. IMPACT OF CLOUD ATTRIBUTES ON INTRUSION MONITORING

There is a substantial gap in security monitoring between cloud and non-cloud paradigms [18]. Therefore, a cloud monitoring infrastructure should satisfy the following properties.

A. Multi-Tenant Security

Cloud computing differs from traditional approaches in IT architecture: it takes advantage of large-scale sharing of computing resources among multiple tenants at several levels - network, host, and application - with the help of virtualization. This is to be compared with building dedicated computing en-vvironments for the sole use of single tenants, especially in cor-porate IT infrastructures – protection was usually performed accordingly with dedicated security supervision architectures per monitored environment.

With the cloud, the user could manage the security of his VMs using a similar approach: protecting a single-tenant infrastructure with a dedicated security monitoring architecture. However, the attained levels of security would not be comparable, due to VM exposure to a shared risk, as the computing environment is shared between multiple VMs (so-called fate-reputation sharing [19]).

If the cloud provider is to be responsible for the security of the workloads he hosts, security monitoring architectures must evolve to handle multi-tenancy efficiently. Distributed and shared security control principles must then be adopted: indeed, operating as many independent IDPS systems as individual IT infrastructures is not efficient at cloud provider level.
B. Scalability

Cloud computing introduces also a major evolution regarding scalability, with objectives of massive hosting, communication, and storage capabilities. In such a large-scale, multi-tenant context, security expectations will be stronger for VM hosting, with a likely raise of the need for effective IDPS protection. Shared security monitoring infrastructures should be able to auto-scale in terms of resource capabilities to monitor both large and small IT infrastructures. Besides, a current operational IDPS limitation comes from the too large amount of security alerts to be analyzed. Scalability is thus definitely a major concern when intending to deploy IDPS services at cloud level.

C. Adaptability

Cloud elasticity means that users are able to rapidly instantiate or release computing resources as needed. This results in user context changing over time. Future resource needs become hard to predict by the cloud provider. This dynamicity may also induce VM migrations between hosts. Moreover, as users self-provision their computing resources, they may perform changes to their own VMs. The IDPS infrastructure should therefore be adaptable enough to keep up with such dynamic contexts. A degree further, one may wish the IDPS to be self-configuring, to enable protecting with the same mechanisms several IT infrastructures operated in different technological contexts.

IV. LIMITATIONS OF TRADITIONAL IDS TECHNIQUES

In terms of architecture, we now review the limitations of traditional network-based and host-based security monitoring when applied to the cloud setting. We also briefly introduce the hypervisor-based monitoring approach (described more in-depth in Section V), and discuss the relevance of existing detection methodologies in the cloud context.

A. IDS Types and Architectures

1) Network-based Security Monitoring:

   a) Perimeter protection.: Originally, IDS technologies were preferentially introduced as network appliances: these are machines located on the network to monitor and inspect traffic for anomalies. This design offers some valuable advantages in terms of simplicity of implementation, with a few appliances to be managed under the control of security teams. This type of IDS technology is known as network-based IDS.

   This type of IDS mostly offers perimeter protection usually deployed at the boundary between networks, in proximity to border firewalls or access routers which are interconnection points between the network to be protected and the rest of the world. Thus, such network-based appliances mostly target external threats, e.g., protocol violations, malicious code, viruses or spam coming from the rest of the world.

   b) ...also on internal networks.: Insider threats result from transactions coming from within (e.g., administrators accessing servers directly, users of applications abusing unauthorized access). Such threats are not visible from a point of control located at the network entrance point. This led to adopt network-based protection to monitor traffic also on internal networks, and not only at the IT infrastructure entry point.

   c) In the Cloud: Vitalizing such type of protection to secure virtual networks in the cloud might not be enough. Physical host sharing between VMs (possibly belonging to multiple users) introduces additional security requirements [20]. The following risks must be accounted for: (1) breach of network traffic isolation between VMs; (2) compromise of hardware isolation; and (3) possible side channel effects due to resource sharing. In this context, exposure of customer resources to threats does not only come from the external world through a unique entry point to be monitored and from the internal world, but potentially from every VM co-located on the same physical host.

2) Host-based Security Monitoring: The cloud virtualization layer hides the physical deployment of resources to users. The user simply does not know where the VMs he instantiates are physically located. Moreover, VMs may move when the user dynamically reconfigures his computing workload. From the user perspective, it is thus almost
impossible to architect his IT infrastructure to ensure that traffic keeps passing through network monitoring points when changes occur. Therefore, IDS protection deployed on internal networks might become inoperative to protect user resources (VMs and/or virtual networks). A host-based IDS technology might therefore be the solution for the cloud, because of its independence from the workload topology.

A host-based IDS monitors the properties of a single host and the events occurring within. It consists in a software agent installed within the host, i.e., within a VM in the cloud case. This technique offers the advantage of good visibility and good context analysis of the code running in the host. However, its deployment is rather intrusive. Moreover, it suffers from a strong dependence on the environment (OS, application.).

Host-based IDS technology may also come in the form of a dedicated standalone appliance, bound to the protected host, and monitoring network traffic going to and from the host. Although deployed in the network in front of the monitored host, this specialized IDS protecting a single host behaves effectively as a host-based IDS. To ensure the topology in-dependence required within the cloud, i.e., to keep the VM and the IDS bound together when the VM moves: either both the VM and IDS move together; and all network packets are redirected to the unmoving IDS, but at the price of performance degradations.

In both types of host-based IDS, operating such technologies on behalf of users is out of the scope of a IaaS cloud provider, as the IDS is too tightly coupled with the VM to protect, instantiated by the user and under his control.

3) A New Approach: Hypervisor-based Monitoring: For a delegation to the cloud provider of protection of user VMs from intrusions to be possible, mechanisms should be provided to monitor VM activities from “outside” the VM without being topology-dependent. This is not possible by monitoring traffic in the network or in front of physical hosts: this option misses traffic exchanged between VMs co-located on the same physical host. Said differently, such traditional approaches monitor the physical network infrastructure, but not the virtual network infrastructure.

A promising approach for virtualized platforms is the use of virtual machine introspection [21]. In this design, the IDS run at hypervisor level and allow to monitor and analyze communications between VMs, and between hypervisor and VMs. This solution will be presented more in depth in Section V.

This new approach, which takes advantage of virtualization and was unavailable to physical environments, falls completely within the scope of the cloud provider. Note that, as introspection offers a privileged access to the hypervisor, introspection-based IDS management by the IaaS user is clearly not adequate to avoid making exploitation of introspection a new vector of attacks. This approach is foreseen to open new opportunities to offer IDS Security in an “as-a-Service” manner once the technology reaches maturity.

B. Detection Methodologies

As recalled in Section II, there are two broad families of detection methodologies: knowledge-based and behaviour-based. When moving to the cloud, to improve detection accuracy of these methods, it is essential to tune the IDS configuration. This means that security-sensitive events to monitor should match as closely as possible the threats visible effects. The rule is simple: the better the IDS knows what it should protect (and against what), the better protection can be. Tuning is necessary to avoid: (1) failing to identify malicious activities (false negatives); (2) incorrectly identifying benign activities as malicious activities (false positives).

It is desirable for the tuning operation to be partially or completely automated. Indeed, as the user workload is by definition dynamic, the IDS must be able to keep in sync with user-performed changes at the virtual level: VM creation, cloning, migration, reconfiguration... This automation becomes absolutely crucial to optimize IDS security if user VM security monitoring is to be managed by the cloud provider, as the provider has no a priori knowledge of the code running in the user VMs.
1) Knowledge-based Systems: For signature-based systems, tuning means, for example, that the system should ignore attacks that target IIS web servers if there is no IIS web server in presence. In that case, the signature database should be restricted to signatures applicable to assets in presence (OS, applications...).

Some passive sensing technologies are already deployed to gather context about what is running in VMs, based on network traffic analysis. Unfortunately, manual tuning generally remains necessary and improvements are still required to bridge the information gap [22] – a situation when no information about the software running in the VM is available to the IaaS provider outside the VM. To our knowledge, opportunities for improving automated context discovery and understanding have not been fully explored yet in virtualized environments.

2) Behaviour-Based Methods: The applicability to the cloud of this class of detection methods still remains difficult for several reasons. First, the complex life-cycle of VMs (e.g., clone, suspend, transfer, modify, restart...) under the control of the user makes any assumption of reliable provider-performed training unrealistic at VM level. Moreover, the cloud provider cannot assume a trusted state to generate a reference profile. Furthermore, many false positives might be generated when behaviour deviates significantly from the reference profile, each time the workload is modified by the user.

V. HYPERVISOR-BASED MONITORING AND VM INTROSPECTION

We now present the most promising approach for cloud VM monitoring. This method also known as Virtual Machine Introspection (VMI) operates detection mechanisms via the hypervisor.

A. VM Introspection

1) Principle: The hypervisor component has been viewed as a “security ecosystem enabler” [23]: virtualization is lever-aged to provide better security than in physical environments, compared to protection based on operating systems (host-based security) or appliances in the network (network-based security).

The underlying approach is based on the foundational work by Garfinkel et al. [21], [24] which introduced VMI in 2003. It consists in providing full visibility into the resources allocated to the VMs through the mediation of the hypervisor. This visibility of VM activity is offered to dedicated security VMs, by nature “outside” the VMs they monitor, and which may be pulled into a completely different hardware for security and reliability purposes. Thus, introspection-based security monitoring offers both the advantage of high visibility offered by host-based protection and of isolation inherent to network-based protection – due to separation from the execution context of the protected VM.

Protecting user VMs from “outside” using a security VM is called the security offloading approach: security functions are external to the protected VM, and may be provided to third parties as services. This paradigm, commonly referred to as Security-as-a-Service, designates enhanced cloud delivered security capabilities thanks to security offloading. The security VM is then called a Security Virtual Appliance (SVA) [25].

VMI leverages the capabilities of the hypervisor to provide transparent and privileged access to VM resources. For instance, legitimate memory usage can thus be controlled at a very fine-grained level by comparing monitoring information gathered from different layers, e.g., hypervisor and VM, to enforce integrity policies and detect violations. The security of this class of mechanisms results from the hypervisor being trusted, and running at a higher level of execution privilege.

This means that the hypervisor is assumed to be part of the TCB (Trusted Computing Base), i.e., the Cloud provider is seen as trustworthy. This assumption may be seen as reasonable in a first step: if many concerns have been raised regarding the security of the virtualization layer due to bugs or exploit reports, and some first solutions proposed such as hardening or minimizing the TCB, such solutions are still not mature, with many research challenges still to be overcome.
2) Introspection Capabilities: Three main categories of introspection capabilities may be distinguished according to the type of monitored resource: storage, networking, and computing.

a) Storage: Storage introspection offers the capability to read and write from/to VM virtual disk images. Typical uses of this API are: anti-virus/anti-root kit solutions, forensics and Compliance analysis, VM state discovery (OS, services, ports, etc.), or management of check pointing/recovery logs. Storage introspection is already available with first released SVAs.

b) Networking: Network introspection enables to pro-vide to SVAs a view of the network traffic in and out of each VM. This network monitoring capability is provided regardless of the virtual network configuration (i.e., the internal vSwitch setup). This means that no network reconfiguration is required when VMs migrate. This is different from network-based appliances which have to be located – through the internal vSwitch network configuration – on the virtual networks to be monitored, thus requiring network re-configuration in case of VM migration. Typical uses of this API are: firewalling, IDS/IPS, protocol analysis/Deep Packet Inspection, or network discovery. Network introspection raises no real scientific issue: the number of available industrial network-related SVAs is growing as implementation and integration challenges are overcome.

c) Computing: Compute introspection provides to SVAs the capability to monitor and control the virtual CPU state. It also allows inspection and modification of all memory pages of the VMs. Typical uses of this API are: rootkit detection, known virus and malware signature inspection, vCPU instruction analysis. Even if a (very limited) number of products are available, compute introspection remains a tough research challenge, with an ongoing scientific publication effort [26], [27], [28] as discussed in the next paragraph.

B. Introspection-Related Research

Since 2003, the work of Garfinkel et al. on VMI has paved the way of the security offloading philosophy for virtualized platforms. Compute introspection aims at monitoring VM CPU and memory usage directly at the hypervisor level – and not from inside the VM through its OS – to detect and prevent unwanted activity such as unauthorized access or use of VM resources.

The main introspection roadblock is the difficulty to bridge the semantic gap [29]: one needs to extract high-level semantic information from low-level binary data sources. This operation is performed without the help of the guest OS which provides the support to interpret the manipulated raw resources. Virtualized physical resources are examined at the hypervisor level, and thus from outside the VM.

This semantic interpretation is required for the security tools to check safety of systems, as they perform the security analysis from the knowledge of high-level qualified information. This interpretation is difficult because it requires a deep knowledge of the OS resource usage – hard to acquire when source code is not available. Moreover, this knowledge is highly specific to each OS type and version. Unfortunately, in IaaS clouds, the provider has a priori no knowledge of which OS runs in the user VMs (information gap issue [22]).

Using introspection offers obvious security benefits in terms of isolation of the IDPS from the monitored VM, when compared to host-based monitoring. The IDPS, being pulled Completely outside the VM to protect, becomes more difficult to subvert for an attacker gaining access to the monitored VM.

In recent years, the academic community has paid significant attention to the following introspection capabilities initially outlined by Garfinkel: inspection and interposition.

• Inspection enables the IDS to have a complete and consistent view of the VM system state through the hypervisor: CPU state, registers, and full state of memory and I/O devices. VMI-based security tools have in common the goal of minimizing the amount of necessary security extra-code inside the VM to help [30] the “out-of-VM” introspecting component (in the hypervisor) to bridge the semantic gap [21], [31], [32], [33], [34]. For
instance, CloudSec [32] examines particular data structures to look for hidden processes, and requires no code in the VM.

- Interposition refers to the VMI capability to intercept the VM execution control flow and to manipulate safety the VM state. This is of particular interest to react to intrusions. In active monitoring mode, the introspection component (through a security hook) can intercept the guest OS execution control flow and suspend VM execution while passing control to a security component for analysis. The security hook may be located either in the VM [35], or outside the VM in the hypervisor [32].

A large amount of the academic proposals have been proto-typed using the XenAccess [36] introspection library running on the Xen hypervisor. While the benefits of the “out-of-VM” approach are very attractive, this domain remains a challenging research topic [37], [38].

Many current proposals are partial and specific to some security mechanisms. One can cite integrity checking [33], forensics [39], [40], detection of rootkits [41] and malware analysis [31], [42], sandboxing [43], or privacy services [44]. VM introspection is also being applied to new application domains such as Android security [45].

The main tough question remains automated bridging of the semantic gap [29], [46] with issues such as: reverse engineering of low-level data formats data retrieved from virtualized physical memory through the hypervisor [47], [48] and signature generation [49], [50] from which high-level semantic information can be extracted; or closer cross-layer collaboration between the VM and the hypervisor, such as active help from the guest OS [51], [52] – so called symbiotic virtualization [53]. All in all, the scientific publication effort remains very active in this area [54], [55].

Some longer-term challenges include: how to guarantee security monitoring of the hypervisor layer itself – security foundation of the entire cloud stack – which has been shown to be increasingly vulnerable to attacks [12], [13]; and designing security architectures able to define monitoring scopes that can be flexibly coordinated both vertically (cross-layer) and horizontally (across security domains, in hosts and networks), to enable rich analysis of security events collected from different sources of monitoring (e.g., server logs, network flows, ...). We are currently working on a first version of such an architecture for the inter-cloud setting [61].

VI. LANDSCAPE OF VIRTUAL LAYER MONITORING ARCHITECTURES

We now present how the architectural principles (traditional and emerging) described previously are embodied in “real life”. We provide an overview of the current landscape of currently available IDPS solutions (mostly industrial) for the security of virtual environments, for each type of monitored resource: computing and storage, and networking. We finally provide an assessment of each architectural approach.

Note that most of the existing systems are dedicated to network traffic monitoring and analysis. They may be classified into two broad types of architectures: (1) network-based architectures; and (2) hypervisor-based architectures. These are referred to in the sequel as virtualization-unaware and virtualization-aware systems, respectively.

A. Computing and Storage Offloaded Monitoring

Currently, “out-of-VM” industrial IDPS dedicated to computing and storage resources are available on the VMware platform only. Analysis of computing resources is made possible through the VMware compute introspection API. It may be used to monitor VM memory changes of state, as for instance in IBM’s SVSP security appliance [56]. However, the use of this API remains today very limited due to the semantic gap issue already discussed.

For storage analysis, VMware provides two APIs offering different capabilities. The storage introspection API provides a low-level (i.e., block-level) view of storage in passive mode. Juniper vGW and Reflex VMC SVAs use this API to discover the landscape of hosted VMs [57], [58]. The End Point Sec (EPSec) API allows real-time and active storage analysis at high-level (i.e., file level) for anti-virus and anti-malware solutions. In this case, the SVA does not have a direct access to storage. Only a thin agent, installed in each VM to be protected, accesses storage directly. This
agent offloads file events towards the SVA for analysis, through the EPSec loadable kernel module located in the hypervisor. Trendmicro is the only vendor currently using this API in its Deep Security SVA [59].

B. Virtualization-Unaware Network Monitoring

To monitor virtual network resources, the first initiative of traditional IDPS vendors has been to virtualize their existing physical appliances to co-localize them with the VMs to protect, thus enabling inter-VM traffic monitoring. As such, these appliances have no particular capabilities to integrate virtualization infrastructure features. In virtual network-based architectures (VNET), these virtual IDPS can monitor the VM network traffic through supervision of the corresponding virtual network (Figure 2). This is only possible if the IDPS is located on this vNetwork, some network configuration being required through the virtual switch (vSwitch) setup. In particular, the IDPS may run as a virtual appliance for the entire virtual network of a single tenant. This use case is close to the hardware-based appliance solution, but with higher levels of flexibility and scalability.

![Fig. 2. VNET Architecture.](image)

These solutions do not take advantage of virtualization. They have the same limitations as network-based security solutions in physical environments. Network-based security was conceived to protect entire networks and is therefore not adequate in terms of configuration and cost to achieve a high level of security granularity (e.g., protection on a per-VM basis). This topology-dependent technology also leads to cost ineffective operation in case of VM migration, due to complex reconfiguration and/or high constraints induced when workloads become dynamic.

C. Virtualization-Aware Network Monitoring

Significantly different from localizing the security functions on the virtual networks is the approach where they are deeply integrated to the virtualization platform through security-dedicated APIs available at the hypervisor level (e.g., in-trospection APIs), as shown in Figure 3. In this type of architecture (HYP), VM network traffic is intercepted at the VM vNIC (virtual Network Interface Card) level.

![Fig. 3. HYP Architecture.](image)
With this design, security solutions become topology-independent, meaning that: (1) the SVA is a plug-and-play security component; (2) there is no need for network recon-figuration in case of VM migration. The VMs are simply under the protection of the SVA attached to the hypervisor they are running on. VMs migrate with their own security policies.

In this type of architecture, the SVA includes: (1) capabil-ities of firewalling the VM network traffic with the help of the introspection engine; and (2) IDPS functions. Fire-walling consists either in allowing, rejecting the network traffic, or forwarding all or a selection of the traffic to the SVA IDPS component. According to whether firewalling is performed in the SVA or in the introspection engine, one often speaks of slow-path or fast-path modes. The fast-path mode usually shows better performance, and avoids traffic redirection to realize firewalling. Solutions combining IDPS features in the SVA and firewalling in fast-path mode are said to run in hybrid-path mode. In this class of virtualization-aware products are pure IDS/IPS or Defense-in-Depth solutions (i.e., security products including features in multiple layers, such as firewall, IDPS, anti-virus/anti-rootkit).

An alternative for the previous monitoring architecture (HYP-ALT) consists in redirecting towards a physical IDPS all or a selection of the traffic for analysis, as shown in Figure 4. Traffic redirection is made directly in the introspection engine, under the control of the firewalling SVA which runs in fast-path mode.

This solution offers two benefits. First, the IDPS is not co-located with the protected VMs. The security in the virtual layer is restricted to the firewalling function in charge of redirecting the traffic to be analyzed. Second, the performance and the effectiveness of the physical IDPS are well known. The main disadvantage is the cost of traffic redirection. This cost, together with the one of traffic analysis, is reduced when the analysis is performed only on a selection of traffic.

D. An Assessment

Overall, introducing virtualization-awareness to IDPS tech-nologies brings several valuable improvements. First, the topology-independence requirement is taken into considerations. Second, IDPS security policy enforcement can be very granular with a possible selection of traffic (per-VM, per-protocol,...). Third, the SVA becomes plug-and-play; it needs no configuration to take into account the dynamicity of VM behaviour (VM creation, deletion, migration, modification with network impact). Finally, retrieving the VM security view from outside the VM, at the hypervisor level, enables to isolate the user execution context from its associated security system, and more generally from VM management. Currently, this is the best response to security requirements.

However, these solutions also raise some new issues. First, the introspection engine is an additional piece of code inserted in the hypervisor. It may suffer from software design and coding flaws. Second, with introspection, the SVA gains privileges to access to the hypervisor. Such a hypervisor-centric approach of security could lead to a single point of failure, if not diversified with other types of security mechanisms.

Table I summarizes the main security products for network analysis which provide IDPS capabilities for virtualized en-vironments. Currently, all those industrial virtual IDPS run on the VMware platform. Only a few of them support...
other virtualization platforms. Moreover, the network introspection-based solutions are currently available on the VMware platform only. Note that introspection engines are by nature hypervisor specific. For introspection engines to be usable independently from the virtualization platforms, an abstraction layer should be supplied to security providers to perform operations regardless of the type of the hypervisor.

This assessment may be refined by evaluating deployment architectures associated to the available products through five criteria: virtualization awareness, cost effectiveness, maturity, robustness, and separation of duty. The results are shown in Figure 5.

The main drawback of the vNetwork-based architecture is its lack of virtualization awareness. The hypervisor-based architecture has been designed for being virtualization-aware. It may suffer from drawbacks of being entirely integrated in the virtual layer: lack of separation of duty and of robustness. Its variant with traffic redirection to a physically-separated IDPS maintains the benefit of virtualization awareness. Additionally, it may solve the drawbacks of the previous type of architecture with possible impact on cost. The criteria of performance (not represented on Figure 5) are highly dependent of the products themselves. This requires the product to be evaluated both in terms of the introspection mechanism (if any), and on the IDPS performance on its own in representative conditions of deployment.

![Fig. 5. Assessment of Monitoring Architectures.](image)

VII. FUTURE DIRECTIONS

A. A New Role for the Cloud Provider

In this paper, we highlighted some new orientations for security monitoring technologies to face the challenges identified for virtual environments. We conclude by elaborating on possible contributions of the IaaS cloud provider to detect and prevent intrusions in the cloud.

Without doubt, there is a need for the cloud provider to deploy traditional IDPS security to protect the cloud infrastructure: monitoring perimetric security at various access points to the physical premises of the cloud infrastructure essentially aims to guarantee availability of cloud services. As already seen, this is a holistic, monolithic approach to monitoring. Considering the various tenant workloads, monitored traffic is then considered and analyzed in a rather user-nonspecific manner, rather than being identified as belonging to a given user before customized analysis. Infrastructure security should also be strengthened at the hypervisor level, notably to ensure strict VM isolation.

However, the real challenge is to guarantee security in the virtual layer, as in the cloud boundaries is no longer physical but logical. We argue that the IaaS provider has also a key role to play to guarantee VM security at this level. Indeed, due to resource sharing, the vulnerabilities of a VM can affect the VMs of other tenants through side channels. Therefore, the IaaS provider is in a privileged position to monitor VM activities to guarantee workload security of co-located tenants.
Figure VII-A summarizes the strengths and weaknesses of each architectural approach (host-, network-, hypervisor-based) for operating IDPS protection in the virtual layer, from the perspective of relevant stakeholders (user, provider, third-party).

- **Traditional network perimeter protection** is unlikely to be applicable within the cloud because of the difficulties to monitor user VMs from entry points, as the user virtual workload is time-dynamic, distributed through the cloud and internally exposed to other user workloads due to their co-location on the same physical machine.

- **Host-based protection** (deployed within or in front of the user VM) is likely to be the most relevant option when managed by the user, because of the topology independence of this technology.

- **Hypervisor-based protection** seems to be the most appropriate choice to perform monitoring, if IDPS security of VMs is managed by the cloud provider.

Note that for security reasons, hypervisor-based security monitoring architectures can be operated only by the IaaS cloud provider. Users or third-parties can, at best, interface with it to manage security. Overall, the IaaS provider seems to be in the best position to manage IDPS security, benefiting from being able to operate both physical and virtualized infrastructures. This may bring significant security improvement benefits:

- **Multi-layer IDS security**: correlating security information provided by IDSes monitoring physical infrastructure

<table>
<thead>
<tr>
<th>Product Name</th>
<th>Other Capabilities</th>
<th>Platform</th>
<th>Architecture</th>
<th>Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Proventia VMWare VNET</td>
<td></td>
<td></td>
<td></td>
<td>vShield (firewall remediation)</td>
</tr>
<tr>
<td>Sourcefire Virtual IPS</td>
<td></td>
<td>VMware VNET</td>
<td></td>
<td>vShield APP (with firewall remediation)</td>
</tr>
<tr>
<td>Catbird vSecurity</td>
<td>Defense in Depth</td>
<td>VMware, Hyper-V, Xen</td>
<td>VNET</td>
<td>via vShield APP (with firewall remediation)</td>
</tr>
<tr>
<td>Catbird vSecurity</td>
<td>Defense in Depth</td>
<td>VMware, Hyper-V, Xen</td>
<td>VNET</td>
<td>via vShield APP (with firewall remediation)</td>
</tr>
<tr>
<td>IBM SVSP</td>
<td></td>
<td>VMware HYP (hybrid path)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stonesoft StoneGate IPS</td>
<td></td>
<td>VMware HYP (hybrid path)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TrendMicro Deep Security</td>
<td>Defense in Depth</td>
<td>VMware HYP (hybrid path)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Check Point SGVE</td>
<td>Defense in Depth</td>
<td>VMware HYP (hybrid path)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Juniper vGW</td>
<td>Defense in Depth</td>
<td>VMware HYP (hybrid path)</td>
<td></td>
<td>HYP-ALT when integrated with Juniper SRX including IPS</td>
</tr>
<tr>
<td>Reflex VMC</td>
<td>Defense in Depth</td>
<td>VMware HYP (hybrid path)</td>
<td></td>
<td>HYP-ALT when integrated with HP Tipping Points 'N Platform 'IPS' or McAfee</td>
</tr>
</tbody>
</table>

**TABLE I**

**MAIN IDPS INDUSTRIAL SOLUTIONS FOR VIRTUALIZED ENVIRONMENTS.**
and virtual environment could allow better analysis and anticipation of attacks [60].

- **Optimized reactions to intrusions**: upon detection of an attack, VMs can for instance be protected using the live migration capabilities to isolate attacked VMs, or the firewalling remediation capabilities to adjust flow filtering. APIzation of network infrastructures, also known as Software-Defined Networking, also opens new ways to manage intrusion prevention in the cloud, driven from the network instead of being compute-driven. For instance, it enables network-driven quarantine management of infected VMs, or on-demand dynamic chaining of security appliances (firewall, DPI, IDPS, ...) for adaptive enforcement of security policies.

Thus, there are real opportunities and benefits for the cloud provider to provide IDPS services dedicated to the virtual level, in complement to traditional infrastructure security.

B. Could IDPS Security Solutions Use the Cloud

The cloud paradigm reveals an interesting potential of innovation in terms of security monitoring, should IDPS techniques in turn take advantage of the cloud attributes. We describe below some research avenues for security monitoring in the cloud.

By taking advantage of the cloud dynamicity property for its own purpose, the IDPS infrastructure could become *elastic*, i.e., dynamically allocated and provisioned depending on where and when needed, as opposed to a dedicated and static infrastructure.

Sharing of security monitoring infrastructures between various VMs hosted in the cloud should also result in improving cost-effectiveness and reducing the load of operational management (measured on a per-protected VM basis).

Furthermore, using resource migration capabilities available in the cloud might offer new ways to react to intrusions. For example, a VM under attack could be migrated to an isolated domain to let the security mechanism handle the attack and protect the co-located VMs.

Finally, with the help of the unified and user context-aware view of resources available to the cloud provider, some improvements can be expected in terms of proactive monitoring and analysis. For example, if an intrusion is detected on a VM, the attack might be prevented from spreading over VMs sharing any context (same physical machine, same virtual network...) with the resource under attack, by real-time setting of proactive protection to secure the resources suspected of exposure.

VIII. CONCLUSION

Protecting IaaS-hosted IT resources and services requires to enlarge the scope of security monitoring compared to traditional IT environments. New threats must be addressed, and a number of major challenges intrinsic to cloud environments (e.g., scalability, dynamicity, multi-tenancy, virtualization) must be overcome when defining the security architecture infrastructure, notably to detect and prevent intrusions (IDPS). A transposition in the virtual environment of existing network-based VM monitoring techniques does not address efficiently these issues. Host-based monitoring techniques, as in-guest security solutions, are not suitable for the IaaS provider to protect VMs on behalf of users.

A new approach, called hypervisor-based, takes benefit of virtualization to monitor, through the mediation of the hypervisor and from outside of the user virtual space, all virtualized resources allocated to user VMs. This technique is of particular interest to enable topology-independent monitoring of VM activity at the right level regarding the need for security policies enforcement. Currently, the corresponding industrial solutions focus mainly on network resource introspection and run on the VMware virtualization platform which is (nearly) the only one supporting this approach. Thus, offerings based on the hypervisor-based approach are still suffering from a lack of maturity. For instance, the assessment of cost effectiveness such as how to charge users for the IDPS service requires further investigation.
Fig. 6. Comparison of Architectural Approaches for Operating IDPS.

REFERENCES

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 2, February 2016


