Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Secure virtualization for cloud computing

Flavio Lombardi [a], Roberto Di Pietro [b,c,]*

[a] Consiglio Nazionale delle Ricerche, DCSPI-Sistemi Informativi, Piazzale Aldo Moro 7, 00187 Roma, Italy
[b] Università di Roma Tre, Dipartimento di Matematica, L.go S. Leonardo Murialdo, 1 00149 Roma, Italy
[c] UNESCO Chair in Data Privacy, Universitat Rovira i Virgili, Tarragona, Spain

ARTICLE INFO

Article history:
Received 29 November 2009
Received in revised form
23 April 2010
Accepted 7 June 2010
Available online 22 June 2010

Keywords:
Security
Cloud computing
Virtualization technologies

ABSTRACT

Cloud computing adoption and diffusion are threatened by unresolved security issues that affect both the cloud provider and the cloud user. In this paper, we show how virtualization can increase the security of cloud computing, by protecting both the integrity of guest virtual machines and the cloud infrastructure components. In particular, we propose a novel architecture, Advanced Cloud Protection System (ACPS), aimed at guaranteeing increased security to cloud resources. ACPS can be deployed on several cloud solutions and can effectively monitor the integrity of guest and infrastructure components while remaining fully transparent to virtual machines and to cloud users. ACPS can locally react to security breaches as well as notify a further security management layer of such events. A prototype of our ACPS proposal is fully implemented on two current open source solutions: Eucalyptus and OpenECP. The prototype is tested against effectiveness and performance. In particular: (a) effectiveness is shown testing our prototype against attacks known in the literature; (b) performance evaluation of the ACPS prototype is carried out under different types of workload. Results show that our proposal is resilient against attacks and that the introduced overhead is small when compared to the provided features.

1. Introduction

Internet is on the edge of another revolution, where resources are globally networked and can be easily shared. *Cloud computing* is the main component of this paradigm, that renders the Internet a large repository where resources are available to everyone as services. In particular, cloud nodes are increasingly popular even though unresolved security and privacy issues are slowing down their adoption and success. Indeed, integrity, confidentiality, and availability concerns are still open problems that call for effective and efficient solutions. Cloud nodes are inherently more vulnerable to cyber attacks than traditional solutions, given their size and underlying service-related complexity—that brings an unprecedented exposure to third parties of services and interfaces. In fact, the cloud "is" the Internet, with all the pros and cons of this pervasive system. As a consequence, increased protection of cloud internetworked nodes is a challenging task. It becomes then crucial to recognize the possible threats and to establish security processes to protect services and hosting platforms from attacks.

Cloud Computing already leverages virtualization for load balancing via dynamic provisioning and migration of virtual machines (VM or *guest* in the following) among physical nodes. VMs on the Internet are exposed to many kinds of interactions that virtualization technology can help filtering while assuring a higher degree of security. In particular, virtualization can also be used as a security component; for instance, to provide monitoring of VMs, allowing easier management of the security of complex cluster, server farms, and cloud computing infrastructures to cite a few. However, virtualization technologies also create new potential concerns with respect to security, as we will see in Section 4.

Contributions: The goal of this paper is twofold: (a) to investigate the security issues of cloud computing; (b) to provide a solution to the above issues.

We analyzed cloud security issues and model, examined threats and identified the main requirements of a protection system. In particular, we developed an architecture framework, Advanced Cloud Protection System (ACPS), to increase the security of cloud nodes. ACPS is based on the results of KvmSec (Lombardi and Di Pietro, 2009) and KvmSma (Lombardi and Di Pietro, 2010) prototype security extensions of the Linux Kernel Virtual Machine (KVM Qumranet, year), It is also inspired by the TCPS architecture (Lombardi and Di Pietro, 2010). ACPS is a complete protection system for clouds that transparently monitors cloud components

* Corresponding author at: Università di Roma Tre, Dipartimento di Matematica, L.go S. Leonardo Murialdo, 1 00149 Roma, Italy. Tel.: +39 06 57338246.
E-mail addresses: flavio.lombardi@cnr.it (F. Lombardi), dipietro@mat.uniroma3.it, roberto.dipietro@urv.cat (R. Di Pietro).

1084-8045/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jnca.2010.06.008

and interacts with local and remote parties to protect and to recover from attacks.

In the following we show how ACPS can leverage full virtualization to provide increased protection to actually deployed cloud systems such as Eucalyptus (Nurmi et al., 2009) and (Openecp, 2010) (also referred to as Enomalism Enomaly, 2009 in the following). In fact, OpenECP is a fully open source code fork of the previously open source Enomalism offer; as such, it shares the same architecture and codebase. A prototype implementation is presented. Its effectiveness and performance are tested. Results indicate that our proposal is resilient against attacks and that the introduced overhead is small—especially when compared to the features provided.

One main outcome of our research is a framework that allows virtualization-supported cloud protection across physical hosts over the Internet.

Roadmap. The remainder of this document is organized as follows: next section surveys related work. Section 3 provides background information, while Section 4 classifies cloud security issues. Section 5 describes ACPS requirements and architecture. In Section 6 implementation details are provided, while effectiveness and performance are discussed in Section 7. Finally, Section 8 draws some conclusions.

2. Related work

While privacy issues in clouds have been described in depth by Pearson (2009), cloud security is less discussed in the literature (Gu and Cheung, 2009). Some interesting security issues are discussed in Siebenlist (2009), while an almost complete survey of security in the context of cloud storage services is provided by Cachin et al. (2009). An exhaustive cloud security risk assessment has been recently presented by Enisa (2009). Also worth reading is the survey on cloud computing presented in Armbrust et al. (2009). These papers have been the starting points of our work and we refer to them in terms of problems and terms definition.

A fundamental reference for our research is the work on co-location (Ristenpart, 2009) by Ristenpart. This work shows that it is possible to instantiate an increasing number of guest VMs until one is placed co-resident with the target VM. Once successfully achieved co-residence, attacks can theoretically extract information from a target VM on the same machine. An attacker might also actively trigger new victim instances exploiting cloud auto-scaling systems. Ristenpart shows that it practical to hire additional VMs whose launch can produce a high chance of co-residence with the target VM. He also shows that determining co-residence is quite simple.

Most current integrity monitoring and intrusion detection solutions can be successfully applied to cloud computing. Filesystem Integrity Tools and Intrusion Detection Systems such as *Tripwire* (Kim and Spafford, 1994) and (*AIDE*) (AIDEteam, 2005) can also be deployed in virtual machines, but are exposed to attacks possibly coming from a malicious guest machine user. Furthermore, when an attacker detects that the target machine is in a virtual environment, it may attempt to break out of the virtual environment through vulnerabilities (very rare at the time of writing Secunia, 2009) in the Virtual Machine Monitor (VMM). Most present approaches leverage VMM isolation properties to secure VMs by leveraging various levels of virtual introspection. Virtual introspection (Jiang et al., 2007) is a process that allows to observe the state of a VM from the VMM. *SecVisor* (Seshadri et al., 2007) *Lares* (Payne et al., 2008) and *KVM-L4* (Peter et al., 2009), to name a few, leverage virtualization to observe and monitor guest kernel code integrity from a privileged VM or from the VMM. *Nickle* (Riley et al., 2008) aims at detecting kernel rootkits by monitoring the integrity of kernel code. However, *Nickle* does not protect against kernel data attacks (Rhee et al., 2009), whereas our solution does. Most proposals have limitations that prevent them from being used in distributed computing scenarios (e.g.. *SecVisor* only supports one guest per each host) or just do not consider the special requirements or peculiarities of distributed systems; for instance, *KVM-L4* shares the same underlying technology as Lombardi and Di Pietro (2009) but the additional context switching overhead in the 64-bit scenario, representing the vast majority of cloud hosts, remains to be verified. Also worth citing are *IBMon* (Ranadive et al., 2009), a monitoring utility using introspection for asynchronous monitoring of virtualized network devices, and *LoGrid* (Salza et al., 2006), an example of autonomic reaction system.

Table 1
Comparison of features provided by ACPS, TCPS, KvmSma (KSma) and KvmSec (KSec).

Feature	KSec	KSma	TCPS	ACPS
Semantic View	N	Y	Y	Y
Guest Component	Y	N	N	N
Transparency	N	Y	Part.	Full
Non-Blocking	Y	Y	Y	Y
SWADR	N	N	N	Y
Hot Recovery (by Replacement)	N	N	N	Y
Accountability	N	N	N	Y

In an effort to make nodes resilient against long-lasting attacks, *Self-Cleansing Intrusion Tolerance* (SCIT) (Huang et al., 2006) treats all servers as potentially compromised (since undetected attacks are extremely dangerous over time). SCIT restores servers from secure images on a regular basis. The drawback of such a system is that it does not support long-lasting sessions required by most cloud applications. Similarly, *VM-FIT* (Distler et al., 2008) creates redundant server copies which can periodically be refreshed to increase the resilience of the server. Finally, Sousa et al. (2007) approach combines proactive recovery with services that allow correct replicas to react and be recovered when there is a sufficient probability that they have been compromised. Along with the many advantages brought by virtualization, there are additional technological challenges that virtualization presents, which include an increase in the complexity of digital forensics (Pollitt et al., 2008) investigations as well as questions regarding the forensics boundaries of a system.

Finally, the same authors of this paper proposed Transparent Cloud Protection System (TCPS)—appearing as a poster at SAC'10 (Lombardi and Di Pietro, 2010). That poster introduces some of the scenarios and requirements that are also common to ACPS, however they are only partly sketched in TCPS. In particular, ACPS and TCPS share the positioning of the monitoring system and the requirement that it has to be as much transparent as possible to guests. ACPS extends and completes the architecture just sketched in TCPS. For instance, ACPS enjoys unique features, such as the SWADR approach, the increased decoupling of action and reaction, the increased immunity and integrity of the platform—as well as the integration with real-world architecture—and the support for accountability. All these new relevant features, as well as extensive experiments on both security and performance, make the present proposal a novel contribution (see also Table 1).

3. Background

A cloud (Vaquero et al., 2009) is a pool of virtualized resources across the Internet that follows a pay-per-use model and can be dynamically reconfigured to satisfy user requests via on-the-fly

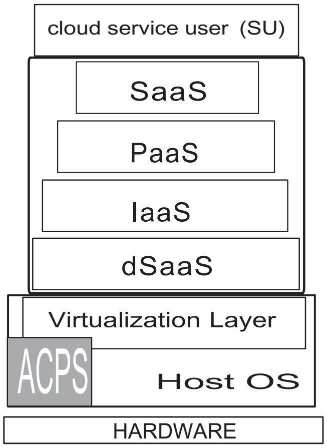

Fig. 1. Cloud layers and the advanced cloud protection system.

provisioning/deprovisioning of virtual machines. Cloud computing is a service model for IT provisioning, often based on virtualization and distributed computing technologies. Within the cloud paradigm, concepts such as virtualization, distributed computing and utility computing are applied (Lenk et al., 2009). Cloud computing approach to distributed computing shares many ideas with grid computing, but these two differ in target, in focus, and in the implementation technologies (Foster et al., 2009). On the one hand, with respect to a grid, the cloud user has less control over the location of data and computation. On the other hand, cloud computing management costs are usually much lower and management is less cumbersome. In the following we will also refer to the cloud infrastructure components as middleware.

Cloud services are available at different layers (see the *-as-a-Service or *aaS layers in Fig. 1): *dSaaS* The data Storage as a Service delivering basic storage capability over the network; *IaaS* The Infrastructure as a Service layer providing bare virtual hardware with no software stack; *PaaS* The Platform as a Service layer providing a virtualized servers, OS, and applications; *SaaS* The Software as a Service layer providing access to software over the Internet as a service.

In this work, efforts have been focused on the "lowest" computational layer (i.e. *IaaS*) since we can more effectively provide a security foundation on top of which more secure services can be offered. Most existing cloud computing systems are proprietary (even though APIs are open and well-known) and as such do not allow modifications, enhancements or integration with other systems for research purposes. This is the reason why we have chosen Eucalyptus and OpenECP, both open source cloud implementations, for integration with our architecture. In the following, even though we will focus on the security issues of those two platforms, most considerations will be general enough to be valid for other platforms as well.

4. Cloud security issues

One of the key issues of cloud computing (see Fig. 1) is loss of control. As a first example, the service user (SU) does not know where exactly its data is stored and processed in the cloud. Computation and data are mobile and can be migrated to systems the SU cannot directly control. Over the Internet, data is free to cross international borders and this can expose to further security threats. A second example of loss of control is that the cloud provider (CP) gets paid for running a service he does not know the details of. This is the dark side of the "Infrastructure as a Service" model, but also of other "as a Service" approaches. To date, misuse problems tend to be regulated by a service contract, where such an agreement should be enforced and controlled by monitoring tools (Haeberlen, 2009).

Some of the security issues of a cloud are (Foster et al., 2009):

SEI1 Privileged user access: access to sensitive outsourced data has to be limited to a subset of privileged users (to mitigate the risk of abuse of high privilege roles);
SEI2 Data segregation: one instance of customer data has to be fully segregated from other customer data;
SEI3 Privacy: exposure of sensitive information stored on the platforms implies legal liability and loss of reputation;
SEI4 Bug Exploitation: an attacker can exploit a software bug to steal valuable data or to take over resources and allow for further attacks;
SEI5 Recovery: the cloud provider has to provide an efficient replication and recovery mechanism to restore services, should a disaster occur;
SEI6 Accountability: even though cloud services are difficult to trace for accountability purposes, in some cases this is a mandatory application requirement.

With respect to the latter point, accountability can increase security and reduce risks for both the service user and the service provider. A trade-off between privacy and accountability exists, since the latter produces a record of actions that can be examined by a third party when something goes wrong. Such an investigation might show faulty components or internal cloud resource configuration details. This way, a cloud customer might be able to learn information about the internal structure of the cloud that could be used to perform an attack. A possible solution could be the use of obfuscation and privacy-preserving techniques to limit the information the VM exposes to the cloud (Bethencourt et al., 2009). Anyway, current technology cannot prevent a VMM from accessing guest raw memory. This leaves open confidentiality issues with respect to the service provider (or with respect to an attacker if he compromises the hosting platform).

4.1. Cloud security model

Fig. 2 illustrates the scenario we are concerned with in this paper. A service provider (SP) runs one or more service instances (SI) on the cloud, which can be accessed by a group of final service

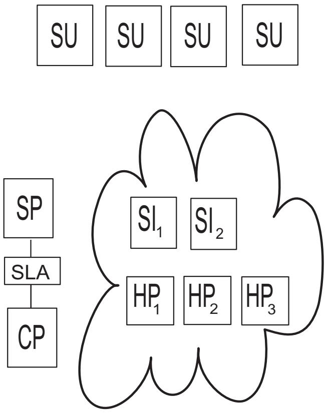

Fig. 2. Cloud service model components: Cloud Provider (CP), Hosting Platform (HP), Service Level Agreement (SLA), Service Provider (SP), Service Instance (SI), Service User (SU).