Computers and Electrical Engineering 000 (2016) 1–15

Contents lists available at ScienceDirect

Computers and Electrical Engineering

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

Cloud security: Emerging threats and current solutions

Luigi Coppolino*, Salvatore D'Antonio, Giovanni Mazzeo, Luigi Romano

Dipt. Ingegneria, Univ. of Naples Parthenope (DI), Naples, Italy

ARTICLE INFO

Article history:
Received 12 August 2015
Revised 4 March 2016
Accepted 7 March 2016
Available online xxx

Keywords:
Cloud computing security
Security techniques
Intel SGX
Homomorphic cryptography
Cloud platforms

ABSTRACT

Many organizations are stuck in the cloudify or not to cloudify limbo, mainly due to concerns related to the security of enterprise sensitive data. Removing this barrier is a key pre-condition to fully unleash the tremendous potential of cloud computing. In this paper, we provide a comprehensive analysis of the main threats that hamper cloud computing adoption on a wide scale, and a right to the point review of the solutions that are currently being provided by the major vendors. The paper also presents the (near) future directions of cloud security research, by taking a snapshot of the main research trends and most accredited approaches. The study is done on a best of breed selection of proprietary and Open Source cloud offerings. The paper is thus a useful navigation tool, that can be used by the IT personnel to gain more insight into the security risks related to the use of cloud computing, as well as to quickly weigh the pros and cons of state of the art solutions.

1. Introduction

Cloud computing is gaining more and more momentum, due to a mix of market and technology related factors. Rapidly changing business conditions are driving a change in the computing infrastructure of many companies. The number of enterprise services and applications is constantly increasing, with new ones being continuously added and old ones being removed. Over three quarters of North American and European companies outsource parts of their business. This implies that relevant business-related information is not only spread across the different computing systems within one enterprise, but it is also distributed across multiple IT infrastructures of the company business network. On the technology axe, the availability of ever cheaper processors and lower latency networks, combined with the astonishing progress in virtualization, enable to move the computation from local IT platforms to distributed cloud infrastructures.

However, evidence is demonstrating that despite cloud computing being seen as a major business avenue for the next years, migration to the cloud paradigm is hampered by concerns on security. For example, financial institutes are attracted by cloud computing but for security reasons they are still in the early stages of adoption. Recent attacks to the cloud, as the one in 2014 when 50 million user accounts of Dropbox were hacked,[1] prove that cloud data security has become a hot topic. Evidence of the risks to which the cloud is exposed have been demonstrated by Al Awadhi et al. [16]. They used honey-pots to confirm that the cloud environment is insecure, and that it is the target of many attacks from different countries.

In order for cloud computing to be seen as a viable alternative, it must provide (at least) the same level of security as traditional IT systems. To achieve this goal, greater awareness is needed about measures and tools that are currently

* Corresponding author. Tel.: +390815476702.
E-mail addresses: luigi.copppolino@uniparthenope.it, luigi.coppolino@gmail.com (L. Coppolino), salvatore.dantonio@uniparthenope.it (S. D'Antonio), giovanni.mazzeo@uniparthenope.it (G. Mazzeo), luigi.romano@uniparthenope.it (L. Romano).

[1] https://blogs.dropbox.com/dropbox/2011/06/yesterdays-authentication-bug/

http://dx.doi.org/10.1016/j.compeleceng.2016.03.004
0045-7906/

available to counter malicious actions. In the literature, there are many works proposing surveys of defence mechanisms implemented in cloud infrastructures [1–7]. However, to the best of our knowledge, they all provide a partial view of the problem: some only describe the countermeasures taken by a specific provider and/or on a specific platform (i.e., [4]), others focus on security techniques that can be used in specific domains (i.e., [3]), yet others limit their analysis to Open Source cloud solutions (i.e., [5]). Our work here is a response to this gap in the literature, in that: i) it provides a study focusing on main challenges and issues at the various levels of a cloud stack, ii) it takes into account both general security approaches and specific solutions currently used in real cloud platforms, and iii) it covers both Open Source and proprietary cloud solutions. In this paper, we provide a comprehensive analysis of techniques and tools that are currently being used by cloud providers to secure their platforms. We perform such an analysis by cross-correlating methods that are applicable in different technical domains to multiple attack vectors, and in particular: network, hypervisor, and computing hardware. For each attack vector, we discuss the type of attacks that can be launched, and the type of countermeasures that are currently implemented/enforced by leading Cloud Providers (CP) and/or platform vendors. The platforms that are analyzed were chosen based on Gartner's magic quadrant[2] report. We selected: Amazon and Microsoft because they are leaders among IaaS providers, VMWare because it is one of the visionary providers, and OpenStack because it is one of the most widespread software stacks (since RackSpace, for example, offers private cloud solutions based on VMWare vCloud or OpenStack). Also importantly, our work provides pointers to emerging research trends and technologies for the (near) future.

The remainder of this paper is organized as follows. Section 2 discusses open issues and current threats in a cloud environment. Section 3 surveys the main attack vectors. Section 4 reviews related research.Sections 5–7 describe, for each attack vector, the mechanisms that are currently available to secure the cloud. Section 8 provides an overview of current trends for addressing open issues. Finally, Section 9 concludes the paper by summarizing the main results of the study and making some final remarks.

2. Security open issues and threats

Cloud computing suffers from a number of security issues which cannot be overlooked. ENISA in its recent report has identified thirteen technical risks. According to NIST, cloud computing presents certain unique security challenges resulting from the cloud's very high degree of outsourcing, dependence on networks, sharing (multi-tenancy), and scale [10]. Fernandes et al. [11] provide a thorough review of the research literature to clearly define cloud security open issues and challenges.

In light of [10,11], it can be claimed that nowadays main security challenges are:

- IS1: Shared technologies vulnerabilities: What makes the cloud so fascinating is also a point of criticality in terms of security. As Navati et al. [12] demonstrate that attackers could exploit vulnerabilities in the hypervisor and gain access to the physical host where other neighboring virtual machines (VM) reside.
- IS2: Data breach: Users' data can suffer both from accidental data loss and from malicious intrusive actions. Data loss is out of the scope of this work, since we only consider here data breaches, that is the action of stealing sensitive data (such as personal or credit card information).
- IS3: Account or service traffic hijacking: A user can lose control over its own account. Many attacks that will be described in Section 3 can lead to account or service hijacking. This enables the intruder to get into critical areas of a deployed service and possibly compromise the confidentiality, integrity, and availability of those services.
- IS4: Denial of Service (DoS): One of the most alarming scenarios is when the cloud infrastructure is made unavailable (just think that an outage costs Amazon 66 K dollars per minute). DoS in a cloud context is even more dangerous than in a traditional one, since when the workload increases with respect to a specific service, the cloud environment provides additional computational power to that service. This means that on the one hand the cloud system counters the effects of the attack, but on the other hand it supports the attacker in his evil activity, by providing him with more resources [13].
- IS5: Malicious insiders: This is climbing the list of cloud top threats. The possibility that a malicious insider – e.g. an employee – might try to take advantage of his privileged position to access sensitive information is becoming more and more concrete and worrisome [14].

3. Attack vectors

The aforementioned open issues can be caused by three main vectors of attack (Fig. 1): Network, Hypervisor, and Computing Hardware. Three types of attackers map on these vectors: external users, internal users, and the cloud provider itself (embodied in a malicious employee).

- External users can launch many attacks against the cloud infrastructure through the network. They can affect data confidentiality and integrity by tampering with the communication channels or by landing on the system to subsequently launch an attack. In addition, they can affect the availability of the CP's data centers.

[2] Magic Quadrant for Cloud Infrastructure as a Service Worldwide: http://www.gartner.com/technology/reprints.do?id=1-2G2O5FC&ct=150519&st=sb

Please cite this article as: L. Coppolino et al., Cloud security: Emerging threats and current solutions, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.03.004

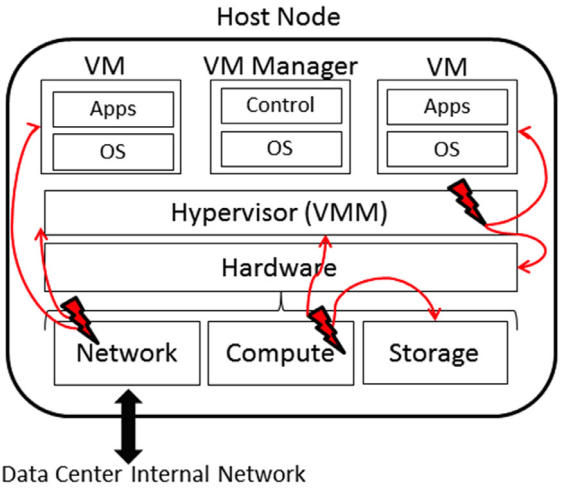

Fig. 1. Cloud platform attack vectors.

- Internal users – i.e., owners of a VM instance – can exploit the Hypervisor to attack another VM instance. The enabling factor of the attacks in this case is Multi-Tenancy, i.e., the fact that both the attacker and the victim share the same host. The inherent vulnerabilities of Multi-Tenant architectures (logical and physical resources are not designed to provide a high degree of isolation among distinct applications) can lead to breaches of confidentiality of sensitive information. These attacks cannot be mitigated by means of traditional security techniques, as in the case of network attacks [18].
- The Cloud provider itself might be an attacker. Employees could exploit their privileged position to steal sensitive users information through either physical or logical manipulation of the hardware platform.

In the rest of this section we provide, for each vector, an overview of the attacks that can be launched against a cloud infrastructure.

3.1. Network-based attacks

The network is one of the main vehicles of attacks against applications running on a cloud platform. Most of these attacks are direct descendants of attacks initially conceived for traditional technology, some were invented after the birth of the cloud computing technology. For completeness, we review them here.

One of the most threatening is the Distributed Denial of Service (DDoS) [10]. Zargar et al. [29] made an exhaustive classification of DDoS flooding attacks, dividing them into two types based on the protocol level that they targeted: network/transport level and application level attacks. A cloud environment could also be the means through which to enforce a DDoS as Badis et al. [31] show in a methodical study of botclouds, providing statistical results on cloud systems exploitation that could lead to an Extrusion Attack.

Examples of attacks directed to data integrity and confidentiality are IP-, ARP- and SOAP- Spoofing- [23] and Sniffing attacks, where the network traffic is eavesdropped in order to steal sensitive information such as username, password and credit card details.

Another common type of attack is Code Injection. In this case hackers insert malicious code into applications to run arbitrary code with the privileges of a vulnerable process and so gain access to user's computing resources, applications (i.e., Cross-Site scripting/HTML injection), or databases (i.e., SQL injection). Malware injection can be network-based, or injections can be done via the hypervisor.

3.2. Hardware-based attacks

Confidential data can be protected from unauthorized access, by storing it in encrypted form and communicating it over encrypted channels, but at some point data must be decrypted in order to carry out the computation. Attackers can take advantage of the multi-tenant environment to access physical resources such as memory bus, disk bus, and data and instruction caches (L1, L2, L3) in which they can find decrypted data and the cryptographic keys of well-known algorithms (AES, DES, RSA) and of other VMs instances. An example of this is the shared memory hierarchy of an Intel Pentium 4 with hyper-threading features. Both the L1 and L2 caches, with the hyper-threading feature turned on, can lead to information leakage from one process to the other [1]. This is called Cache-Based Side-Channel attack and is part of a family known as Cross-VM Side-Channel attacks. In a nutshell, this attack circumvents the logical isolation provided by the hypervisor layer. This threat can come from two enemies: insider attackers (usually cloud employees that take advantage of their privileged position) or malicious customers (that in a first phase must land in the victim server and then launch the attack).

Please cite this article as: L. Coppolino et al., Cloud security: Emerging threats and current solutions, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.03.004

Ristenpart et al. [26] demonstrated the use of cache-based side channel attacks between two VM instances in the Amazon EC2 Cloud. They showed how to achieve co-residency and how to perform load measurement through one of the most famous attacks in the literature, namely the Prime+Trigger+Probe attack.

The theft of sensitive information can be achieved using three possible channels: preemptive scheduling, hyper-threading, and multicore. In the first case the attacker profits of the context switch between his VM and the victim's VM to observe the cache status as the victim had left it. Otherwise, the malicious operation is enforced through a breach of the CPU core sharing. That is, the attacker exploits the multi-tenancy, which is realized with multiple threads running on a single processor with a consequent usage of a common ALU and other resources. The last way through which an attacker can steal information is by reading the L3 cache that is the only one shared when VMs are assigned to multiple cores instead of multiple threads [21].

Three main types of side-channel attacks can be defined based on the information that is leaked: time-, trace- [21] and access-driven [20]. All of them get sensitive information by observing the execution time or power consumption variations generated via cache hits and misses. They mainly differ on the granularity of the captured information. The time-driven attack observes the aggregate profile and so the total number of cache hits and misses. It can be of two types: passive if attackers have no direct access to the victim's machine or active if there is a physical access to the machine. In [8], as an example, it is shown how to carry out a cache timing attack on an embedded ARM-based platform to extract the key of an AES-based authentication protocol.

As in the case of time-driven active, a trace-driven attack requires attackers to access the same machine of the victim. At a finer granularity level, it is observed if a specific access leads to a cache hit or miss. Therefore, unlike the previous one, the study is conducted on individual accesses. Researchers have demonstrated that, especially when multi tenancy is implemented through Hyper-Threading technique, cypher algorithms as RSA and AES can be spied. The last and more recent attack is the access-driven attack, through which the adversary can determine the cache sets that the cipher process modifies. She can understand which elements of the lookup tables have been accessed by the cipher.

Last but not least are boot integrity attacks. Attackers with either logical or physical access could be able to damage boot integrity with bootkits or particular form of malware that resides outside the OS for example within System Management Mode (SMM). Given its position, this type of malware is particularly dangerous because it may re-infect new OS installations. Vulnerabilities have been found in the BIOS, UEFI, Master Boot Record (MBR), CPU Management Engines and PCI device option ROMs [32].

3.3. Hypervisor-based attacks

The hypervisor is the software layer that lies between virtual machines and the physical hardware, in charge of abstracting the underlying architecture. It is a fundamental part to guarantee the cloud multi-tenancy feature. It allocates physical resources to the guest VMs, such as main memory, CPU and peripherals. In terms of security, hypervisors should be considered as the most important layer to protect in the cloud stack, because they have the highest privilege and thus any command can be run from this space. Attackers can achieve full control of any resource of the host system if they alter or compromise the hypervisor (whose original goal was ensuring VMs isolation).

What make hypervisors extremely vulnerable and difficult to secure is the large (around 100,000 LoC) and complex structure of the software. Usually internal attackers, as shown in Fig. 1, start from a compromised VM's operating system to finally land, through the hypervisor layer, in another VM or in the hardware layer.

The NIST guide to security for full virtualization technologies says: If an attacker can successfully escape a guest OS and gain access to the hypervisor, the attacker might be able to compromise the hypervisor and gain control over its entire guest OSs [10]. That is, a single breach in a guest OS can have a terrible impact on the whole virtualization infrastructure. An OS can be compromised using for example injected malicious code placed in: guest-VM user-level space, guest-VM kernel-level space or in the host OS resident within the VM possessed by the hypervisor (as Xen's Dom0). Once the code has been injected to memory, the attacker – to execute it – can overwrite a subroutine return address with the address of the injected code. An injection can induce a buffer overflow (i.e., return-into-libc attacks).

The Host OS, if compromised, can lead to direct access to I/O and networking devices. Each hypervisor has its implementation of the management domain that does the dirty work running the back-end driver for the multiplexed VM. So attacks can differ a little on this surface. Rocha et al. [22] explain a sophisticated attack that demonstrates how a malicious insider can use the functionalities offered by virtual machine introspection to compromise data from a running process in a Dom0's memory space.

Perez-Botero et al. [17] make an exhaustive characterization of the potential attacks of the hypervisor using as a starting point the vulnerabilities defined in the NIST National Vulnerabilities Database. In a nutshell, there are 11 functionalities usually provided by Hypervisors that can be exploited, some of these are: virtual CPU, Symmetric Multiprocessing (SMP), MMU, I/O and networking, Hypercalls, Virtual Machine Monitor (VMM), Interrupt and Timer Mechanisms. In each category a different attack can be conducted by exploiting breaches and vulnerabilities [19].

Some well-known attacks that are worth mentioning are: BluePill, SubVirt, Vitriol and DKSM attacks. Direct Kernel Structure Manipulation (DKSM) [28] is a type of attack that aims at confusing introspection techniques by changing kernel data structure organization. SubVirt, BluePill and Vitriol are examples of hypervisor-layer rootkit attacks, the worst type of

Please cite this article as: L. Coppolino et al., Cloud security: Emerging threats and current solutions, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.03.004

attacks to face. As an example SubVirt[3] is a rootkit attack, which install, through a boot sequence modification, a virtual machine monitor underneath an existing operating system transforming it into a virtual machine and using that VMM to host arbitrary malicious software and so gaining more control over the system than a usual malware.

4. Related work

Before going into the details of countermeasures, it is worth analyzing other surveys on defense techniques enforced in cloud computing, and in particular mentioning those works that deal with CPs' examples.

Takahashi et al. [1] provide a study of security solutions for cloud computing in multitenant environments. The work proposes a layer-based security analysis of cloud enabling technologies. Analysis focuses on solutions at the following layers: HW&SW, Hypervisor, O.S., Application, and Web. For each of them, different security categories are identified, i.e., aspects of security with common characteristics. The authors mention for Hw&Sw, O.S. and Application layers consolidated security countermeasures, unlike hypervisor and Web layers which are, as they state, still evolving due to novelty of technology and service scenarios. This paper reports solutions to address issues related to resource isolation, which is a noteworthy threat although not the only one. Instead, we identify solutions to multiple issues that are of the same importance in cloud security. Also importantly, our research covers examples of existing cloud architectures of well-known CPs to present not only the state-of-the-art but also real examples.

In [2] Zhou et al. survey security and privacy solutions as implemented by main cloud service providers. In particular, availability, confidentiality, data integrity, controls and audit solutions are discussed. Solutions for availability include virtualization and geographic redundancy. For confidentiality, they include physical isolation (dedicated LANs) or VLANs, VPNs, and network middle-boxes (e.g., firewalls, packet filters), storage encryption or preventive encryption. Data integrity is granted by means of highly dependable hardware, such as cutting edge disk drives and tape technologies. Specifically RAID-like strategies are deployed (e.g., Zetta's RAID-6); also digital signatures are used for integrity check on distributed file systems. Controls refer to the possibility to regulate and manage personal data flows and processing in the cloud. Audit solutions observe what happens in the Cloud system: events, logs and monitoring needs. In contrast to our study, this paper only covers solutions reported by CPs' white papers without describing – as we do – the possible solutions, and the technical aspects behind them, available in the community and which of them are currently used by CPs. Another point that sets us apart is the investigation of research trends and upcoming technologies that could contribute to address the main open issues.

In [3] Panth et al. describe countermeasures taken by most famous cloud providers against well-known security attacks. After a brief review of the encryption schemes available in the community there is for each CP a list of the security mechanisms adopted, mainly focused on the techniques of Identity and Access Management (IAM), encryption and standard used by providers. The paper lack of aspects related to intra-host security and so all the countermeasures taken against Virtual Machine (VM) and Virtual Machine Monitor (VMM) attacks that are particularly important because VMs and VMMs are the Achilles heel in the cloud architecture.

Approximately in the same vein, Joseph et al. [7] mostly referenced white papers and technical reports of leading CPs in order to mention the security solutions of various cloud types (Private, Public, Hybrid, Community) and service delivery models (SaaS, PaaS, IaaS). For what concerns the different cloud types, organizations such as Microsoft, Oracle, Cisco and VMware are mentioned to reference techniques implemented in their architecture. This paper surveys only mechanisms to secure the cloud against attacks enforced through the network. Many of the cited techniques are well-known in the literature, such as SSL, firewall, and AES-256. We go beyond this, since we cover multiple technical domains and consider new strategies (specific to the cloud) that are currently used. For instance, in addition to solutions for network-based attacks, we dwell on mechanisms used against hypervisor-based ones, which are particularly challenging since they are more recent (and thus less known).

In [4] Akinbi et al. define a classification of the security in PaaS cloud architectures. They use the Cloud Security Alliance (CSA) security guidance to define the areas on which to focus the study. These are: operational domain, which highlights guidance with application security, identity and access management, encryption and key management as well as virtualization security of a PaaS public cloud environment. Based on the areas of interest, they evaluate the Windows Azure cloud analyzing the security mechanisms offered and highlight lack of controls. The work is conducted on three main areas: Developer Environment Security, Host/Compute Security and Storage Component Security. While the study of Microsoft Azure is exhaustive, the work fails to analyze solutions other than Microsoft offering. We instead survey and compare more than one platform, since multiple providers could face the same threat in a different way.

In [5] Ristov et al. provide a systematic security evaluation of open source cloud infrastructures, namely: OpenStack, CloudStack, OpenNebula, and Eucalyptus. These have been under study analyzing the compliance with the standard ISO 27001:2005. They evaluate 11 assets of the cloud architecture, giving a score (from –1 to 2) that depend on how much a control objective is met. The assets analyzed are: Back-up, Network Security Management, Monitoring, User Access Management, Network Access Control, OS access Control, Application and Information access control, Cryptographic controls, Security of System files, Reporting Information Security and Weaknesses. Finally, by summing up the scores, they define which CP has the best final score. We believe that this work is too much oriented to a quantitative analysis that wants to

[3] http://docs.openstack.org/security-guide/content/ch047_data-encryption.html

Please cite this article as: L. Coppolino et al., Cloud security: Emerging threats and current solutions, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.03.004

Table 1
Contribution with respect to related work.

	Takashi et al. [1]	Zhou et al. [2]	Panth et al. [3]	Akinbi et al. [4]	Ristov et al. [5]	Booth et al. [6]	Joseph et al. [7]	This Work
Issues	IS1, IS3, IS4	IS2, IS3	IS1,IS2, IS3, IS4	IS1, IS3	IS2, IS3, IS4	IS2, IS3, IS4	IS3, IS5	IS1, IS2, IS3, IS4, IS5
Attack Vectors	Network, Hypervisor, Hardware	Network, Hypervisor	Network, Hypervisor	Network, Hypervisor	Network, Hypervisor	Network, Hypervisor	Network	Network, Hypervisor, Hardware
Cloud Platforms	Amazon	Amazon	Amazon, Azure, IBM SmartCloud RackSpace	Azure	OpenStack OpenNebula Eucalyptus CloudStack	–	Amazon, Google	Amazon, VMWare vCloud, Azure, OpenStack,
Hypervisors	Hyper-V Xen	Xen	–	Hyper-V	–	Xen	–	Xen, KVM, vSphere

measure the level of compliance to the standard. Main weaknesses we found are: security mechanisms superficially touched, lack of association between attacks and solutions, lack of a clear division among the multiple cloud layers, and last but not least lack of techniques to address hypervisor- and hardware-based attacks.

Finally, Booth et al. [6] present a high-level level classification of which are the defences used in the community and how these defences can counter multiple attacks. The analysis is done in five areas: collocation DoS, collocation breaches of confidentiality, data integrity and availability, data confidentiality, infrastructure compromise. The main added value of our work, as compared to this paper, is certainly the coverage of solutions and upcoming technologies that address the open issues left open by Booth's paper about physical attacks and infrastructure compromise which are as they say The most unexplored area in cloud security because have not yet been suitable efforts.

In conclusion, previous work has addressed important issues, but more investigation is needed. In particular, to deal in one paper with more than one issue, to analyze the problems at multiple architectural levels, and to report on solutions provided my more than one vendor makes a good addition to the research in the field. Table 1 summarizes and correlates the main concepts of Sections 2–4. We report, for each paper: the type of open issues treated, the attack vectors covered, the cloud platforms and the hypervisors analyzed.

5. Countermeasures against network-based attacks

Cloud Providers must protect traffic inside the infrastructure to securely connect virtual machines one to the other and the traffic coming from outside trying to leave the smallest number of access points. Many cloud providers implement almost the same solutions in the field of network. They use common techniques such as Firewalls, Intrusion Detection Systems (IDS), and Anti-Virus Gateway that are now widely deployed in edge networks to protect end-systems from attacks and to monitor the incoming and outgoing traffic. First we survey techniques to guarantee: systems availability, data confidentiality and data integrity both at channel and system's application level. Then, we give an overview of the way cloud providers protect their infrastructures against network-based attacks.

5.1. System availability

To guarantee the system availability, providers must counter DoS/DDoS attacks. Zargar et al. [29] classify defense techniques to mitigate DoS/DDoS based on the time in which the defense takes place: before (prevention), during (detection) or after (identification) the attack. In turn, the defense mechanisms can be placed on three different positions: source, destination or network.

For what concern prevention techniques, community makes use of filtrations. These can be applied in different places: close to the destination, in the routers, or close to the source. An example is ingress filtering useful to prevent IP spoofing by dropping traffic with IP addresses that do not match a domain prefix connected to the ingress router. Egress filtering, instead, is an outbound filter, which ensures that only assigned or allocated IP address space leaves the network. Finally, router based packet-filtering uses the route information to filter out spoofed IP. Other techniques usually provided: history based filtering, Secure Overlay Service (SOS), remote-triggered black hole and Firewalls.

Against DDoS other possibilities in terms of prevention are Intrusion Detection and Prevention Systems (IDPS), which are placed in-line and are able to actively prevent/block intrusions that are detected, or mechanisms such as load balancing and flow control, widely deployed by cloud providers.

Detection-based countermeasures include Intrusion Detection Systems (IDS) that monitor and detect malicious activity in a system. IDS can be classified into: Host based Intrusion Detection Systems (HIDS), Network based Intrusion Detection Systems (NIDS) and Hybrid Intrusion Detection Systems, which are a combination of previous two. Techniques such as wavelet, spectral analysis, statistical methods and machine learning can be used to detect DDoS attacks. However because of the complexity of cloud environment traditional IDS are not sufficient anymore and in the last years it is in vogue a Distributed Intrusion Detection (DIDS) [33].

As Identification techniques an example is the mechanism of IP traceback (packet marking and link testing). In fact, after a DDoS is detected, the defense system should, firstly, identify the source of attack and then try to block the attack. To do

Please cite this article as: L. Coppolino et al., Cloud security: Emerging threats and current solutions, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.03.004

that, providers can filter the traffic. Usually they use Rate-Limiting that expects usage of tokens issued by destination to sources known as capabilities to identify and prioritize approved flows.

5.2. Data confidentiality and integrity

To guarantee data confidentiality and integrity on the channels and so prevent Sniffing and Spoofing Attacks the basic solution is using an encrypted network protocol that encrypts all the traffic from the source to the destination over the whole trip. SSL and TLS can be used to prevent leakage of sensitive information through communication encryption. Another standard commonly used by CPs is IPsec, a protocol suite for securing IP communications implementing network-level authentication and encryption for each IP packet. Usually these mechanisms protect network traffic to the edge of the cloud network, Virtual Private Network (VPN) and its techniques as SSH and IPsec tunnels are used to defend traffic between servers within the cloud network.

To enforce security in terms of data integrity and confidentiality in applications running in the host against attacks enforced through the network, the most established techniques are: data encryption useful to keep data at rest protected from unauthorized access; key management which deals with the generation, exchange, storage, use and replacement of keys; data tokenization which replace sensitive data with randomly generated values with no mathematical correlation using a secure tokenization system; data backup and replication to prevent accidental or malicious data deletion; mechanisms of authentication and authorization to enforce data access control; proof of retrievability and third party auditors which assume responsibility of monitoring data integrity. Some of these are provided by default, others can be included in a Security as a Service extra deployment.

5.3. Commercial solutions

Cloud Providers offer built-in capabilities or products developed by third party vendors (i.e. TrendMicro, Symantec, Macfee, KasperSky, CloudLink, BitDefender) specialized in security of cloud systems especially when security mechanisms against network-based attacks are implemented at application level. In some cases vendors establish a partnership with specific Cloud Providers; in others they guarantee compatibility with a list of providers and hypervisors. TrendMicro SecureCloud is recommended by Amazon AWS, Microsoft Azure and VMWare vCloud as a third party solution useful to guarantee application data security. Its security toolset contains AES-256 Block-level encryption, policy-driven key management, key management interoperability protocol, and role-based access control to ensure separation of duties and credential rotation for AWS instances.

A remarkable recently released product for cloud data security is HP Atalla Cloud Encryption that supports Amazon AWS and VMWare. It is an encryption solution based on AES (256-bit key) algorithm. What makes Atalla very interesting is the usage of Split-Key Encryption technology that consists in a division in two parts of the key: one part, different for each data object, is hosted by HP Atalla by the Key management Service; a second part, the master key, unique for all the data objects, is held by the application owner and is unknown to HP. Furthermore, HP Atalla Cloud Encryption is one of the first commercial realization (even if partially) of the homomorphic encryption (described in Section 8) , which is used for the mathematical operation that combine and split the encryption keys. In this way the keys will be encrypted before and during their use and will be never decrypted when they reside in the cloud.

An additional third-party security system that is worth mentioning is CipherCloud, which provides granular field-level control over encryption and tokenization for sensitive data in the cloud. The platform is deployed as an on-premises gateway that gives users the possibility to encrypt data in real time before sending it to the cloud environment, which means direct control over the encryption or tokenization process, as well as exclusive ownership and control over the encryption keys which are stored locally and are not shared with the cloud provider.

OpenStack network (known as Neutron) is an architecture organization.[4] In a standard deployment there are four distinct physical data center networks where each one represents a security domain: management, guest, external and API networks. Depending on the choice of the architect, there are different possibilities that can be used to protect the internal- and external-source traffic. OpenStack offers two different mechanisms for traffic segregation: VLANs or L2 tunnels using GRE encapsulation.

VLAN networks are isolated from each other at L2 by sharing the same physical network, and can even have overlapping IP address spaces. Each distinct physical network supporting VLAN networks is treated as a separate VLAN trunk. L2 tunnels permit to reduce the visibility of individual tenant traffic from a monitoring point of view by adding a layer of obfuscation to network data traffic. The tunnel is realized taking advantage of IPsec. However, it is insufficient to rely only on security domain separation to protect internal traffic. Because of this, OpenStack recommends to secure all domains with TLS that must be preferred to SSL due to some vulnerabilities of this protocol.

Moreover, other mechanisms supported by OpenStack to protect the cloud network are Access Control lists applied to security groups, Load Balancing based on the High Availability (HA) proxy implementation, Firewalls, VPN. Host- and Network-based IDSs are provided in OpenStack via Security Onion, a Linux distribution for intrusion detection, network security monitoring, and log management. NIDS is composed by Rule-driven NIDS, which looks at network traffic for fingerprints that

[4] http://docs.openstack.org/security-guide/security-guide.pdf

Please cite this article as: L. Coppolino et al., Cloud security: Emerging threats and current solutions, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.03.004

Table 2
Commercial solutions to secure network vector.

Attacks	Amazon AWS	OpenStack	Azure	VMWare Cloud
Malware Injection	–Barracuda Web App Firewall –TrendMicro DeepSecurity –Symantec Endpoint Protection –Deep Packet Inspection –AWS Security Groups	–OpenStack Neutron Network Security –OpenStack Security Onion (Rule- Driven NIDS Analysis-Driven NIDS)	–Microsoft Advanced Threat Analytics –Barracuda Web Application Firewall –TrendMicro Instant- On-Cloud Security	–WebApplicationFirewall (WAF) –VXLAN-extended networks –KasperSky Sec for Virtualization –VMWare vShield Bundle
DoS	–Elastic Load Balancer –NGINX –Incapsula DDoS protection –Neustar DDoS protection	–F5 DDoS protection –DefenseFlow (SDN-based protection) –OpenDayLight (SDN-based protection)	–Azure CDN –Azure Load Balancer –Barracuda NG Firewall	–VMWare vShield Edge –Barracuda NG Firewall
Spoofing	–AWS IAM –AWS MFA	–OpenStack Keystone	–Azure Active Directory IAM	–VMWare Horizon Application Manager
Sniffing	–AWS DirectConnect –HP Atalla Cloud Encryption	–OpenStack Neutron Network Security		–VMWare vShield Edge –CipherCloud gateway –HP Atalla Cloud Encryption

match known anomalous traffic; and by Analysis-driven NIDS that monitors network activity and logs any connections, DNS requests, SSL certificates, HTTP, FTP, IRC SMTP, SSH, SSL, and Syslog activity that it sees. The HIDS performs log analysis, file integrity checking, policy monitoring, rootkit detection, real-time alerting and active response.

Microsoft Azure provides a topology in which each customer can have multiple VMs in one or more deployments that are organized in a virtual network. VMs inside a deployment can communicate with each other using private IP addresses.

The internal communication is protected using SSL encryption. Moreover, a virtual network and the IPsec-based security are used to secure the communication among multiple deployments. Not all the VMs can be accessed directly from Internet, only those that are defined as input endpoint by opening the related ports. IP access control lists are defined on the endpoint to control which entity is allowed to communicate. The ACL is applied on a Load Balanced Set (LBS). Top-of-rack switches are used to defend against spoofing attacks on internal networks to restrict which IP and MAC addresses the VMs use [4].

For what concerns DoS/DDoS attacks, Microsoft makes available standard detection and mitigation techniques such as SYN cookies, rate limiting, and connection limits. Another mechanism put in place to mitigate DoS is to route all the traffic through a Load Balancer infrastructure. However, as Microsoft itself admits, this solution can only partially mitigate the problem.

Azure uses Network Packet Filters (NPF) to prevent spoofing. Network Packet Filters ensure that untrusted VMs do not generate spoofed traffic, cannot receive traffic not addressed to them, cannot direct traffic to protected infrastructure endpoints, and cannot send or receive inappropriate broadcast traffic. Network packet filtering is done by the hypervisor and the root operating system.

VMWare secures the network via VMWare vShield Edge.[5],[6] VMWare vShield Edge provides a perimeter security for each tenant using network security gateway services as firewall, VPN, Web load balancer, IPsec, NAT and DHCP services to monitor packet headers for source and destination IP addresses in order to prevent IP spoofing and DDoS.

The vShield Edge firewall provides network perimeter security and services to a tenant. It isolates the tenant's stub network from the shared (uplink) networks. The isolation is realized with two approaches: VLAN isolation and Cloud network or port group isolation. The first design approach is used to protect against higher-layer network attacks, each tenant has its own VLAN composed by multiple VMs and one acting as a firewall.

This has two network interfaces: one connected to the uplink port that provides accesses to the external world, the other interface is connected to the internal port group. All VMs of a tenant are connected to the internal port group and these VMs are allowed to communicate with each other without going through the vShield Edge firewall VM. In the cloud network (port group) isolation there is an additional entity, the virtual distributed switch (vDS), which keeps track of where the traffic is coming from and where it is headed for. It determines if the traffic can flow directly or if it must go through an edge firewall. The vDS has an important role when a VM connected to a port group attempts to communicate with a VM connected to another port group. In that case the traffic is not allowed. It must be noticed that the vShield Edge firewall virtual machine is protected through the VMware DRS and VMware HA features that ensure availability of the firewall VM.

Table 2 reports previous and others examples of commercial solutions useful to secure the network vector of attack.

6. Countermeasures against hardware-based attacks

A first line of defense, against some hardware attacks shown in Section 3.2 is obviously ensuring a high level of physical security of the data center that is, providing a strict surveillance of the server rooms that support the cloud architecture.

[5] http://www.vmware.com/files/pdf/techpaper/vShield-Edge-Design-Guide-WP.pdf
[6] http://www.vmware.com/files/pdf/products/vcns/vmware-vcloud-networking-and-security-overview.pdf

Please cite this article as: L. Coppolino et al., Cloud security: Emerging threats and current solutions, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.03.004

However, at a certain point if there is a malicious behavior by cloud provider administrators, which are certainly in a position to violate customer confidentiality or integrity bypassing physical security controls, things get much more difficult. Somehow this problem can be solved. In fact new emerging technologies as Intel SGX and ARM TrustZone may hinder side-channel attacks (see Section 8)

Other countermeasures to prevent the loss of information when the host is under side-channel attacks start from the assumption that the adversary had access to the host and so in some way it is necessary to mislead the adversary for what concerns the cache access time. This can be done by making changes to cypher algorithms phases that result in modifications to cache and memory access patterns.

Rani et al. [25] show software and hardware countermeasures in this sense. Noteworthy software methods are: 1) Turn-off Cache S-Box access, that is, loading data directly from memory when there are S-Box accesses and so eliminating potential cache-hits and cache-misses. This kind of defense, on the one hand, has the advantage of constant access time but, on the other hand, reduces significantly the performances. 2) Avoid Lookup Table. 3) Insert Dummy Operation, a technique used in particular when attacks operate on behavior traces rather than timing information: with enough dummy loads inserted the adversary cannot be sure if a given cache-hit or cache-miss is produced by real or faked execution. It is not really efficient since the randomization is simply noise that can be statistically removed. 4) Perform Cache Warming, or rather, preparing the cache before the execution begins giving it all the data necessary to carry out the encryption in order to avoid cache misses. This solution is good only in theory, in practice it is impossible that other processes do not evict the data copied in the cache during the warm up phase. Moreover, it is quite difficult that the S-box fix entirely into the cache.

Hardware countermeasures are: Partitioned Cache (PC) and Partition-Locked Cache (PLC). In the first one the cache is dynamically split into protected regions that can be specifically configured for an application. In partitioned caches, a part of the cache is allocated exclusively to the protected process in order to prevent information leakage. This may cause inefficient cache sharing since the cache partition is fixed statically. The second one proposes a fine-grained locking control so that only the cache lines, which contain the critical data, are isolated. In PLC, creating private partitions locks interested cache lines. These cache lines cannot be used by other cache accesses that are not belonging to private partitions. PLC is a flexible cache partitioning mechanism that achieves less performance degradation.

The other possibility to protect the HW is through cryptography. One of the most accepted by CPs is Intel Advanced Encryption Standard New Instructions (AES-NI), which may counteract cache-based software side-channel attacks. It consists in a hardware implementation of some sub-steps of the AES algorithm. This results not only in a significant speed up execution of the AES encryption/decryption algorithms but also in a higher security. In fact, software cache-based attacks, as we have seen in the previous section, exploit the fact that encryption block, keys and lookup tables are held in memory. Since AES-NI is hardware-based, it has no need for lookup tables and the encryption blocks are executed in hardware within the microprocessor. Moreover, the instructions latency is data independent; this means a major difficulty for an attacker to carry out time-driven cache attacks. AES-NI is transparently available to the end-user and is supported by some provider servers as Amazon AWS, Microsoft Azure and by software cloud platforms as OpenStack.

To fight boot integrity and more in general platform integrity attacks a common solution is using chain of signatures to realize measurements and attestation of software and platform components. The platform key used to verify signatures usually resides in a Trusted Platform Module (TPM), a dedicated microprocessor well described by the Trusted Computing Group (TCG). Intel TXT[7] is a product that makes use of TPM to realize a Dynamic Root of Trust Measurement able to protect the BIOS and to guarantee firmware cleanliness. It is used by leading cloud infrastructure as Amazon AWS, VMWare vCloud and OpenStack. However Intel TXT is not the panacea, Wojtczuk et al. [35], i.e., attacked TXT by taking advantage of a bug in the VMM. Other bugs were found during the years. For this reason Intel is going to release SGX that seems to address weaknesses of TXT.

7. Countermeasures against hypervisor-based attacks

Multiple hardware and software isolation techniques can provide a secure separation of resources. Some hardware techniques shown in the previous part (i.e., AES-NI) can influence security also at this level and thus they could be considered as part of hardware-based hypervisor protection.

As the OpenStack guidelines suggest[8] an evaluation on which hypervisor platform can be chosen must take into account HW-based and SW-based security technologies provided or supported. Remarkable HW technologies supported almost by all hypervisors are Intel Virtualization Technology (VT)[9] and AMD Virtualization (V), which are efficient hardware architectures designed for virtualization in what is known as Hardware-assisted Virtualization (HaV) rather than software-based virtualization.

As well as optimizing usage of resources, one of HaV main goal is to prevent threats to hypervisor integrity and enhance the isolation in system hardware resources. Intel and AMD equip servers with a new ISA in which privileged instructions are handled differently. A new type of processing mode has a separation between hardware privilege level, which is VMM execution mode, and virtualized privilege level, which is guest OSs execution mode.

[7] Intel TXT White Paper http://www.intel.com/content/dam/www/public/us/en/documents/white-papers/trusted-execution-technology-security-paper.pdf
[8] http://docs.openstack.org/security-guide/content/hypervisor-selection.html
[9] https://software.intel.com/en-us/articles/intel-virtualization-technology-for-directed-io-vt-d-enhancing-intel-platforms-for-efficient-virtualization-of-io-devices

Please cite this article as: L. Coppolino et al., Cloud security: Emerging threats and current solutions, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.03.004

The mechanism of virtual memory management have been revised. HaV introduces the concept of hardware physical memory virtualization where memory addresses are translated from guest virtual to guest physical and finally to system physical. Furthermore, HaV provides a secure IOMMU unit, responsible of the PCI, which allows virtual machines to directly access peripheral devices. In this way the hypervisor is protected against malicious devices that may attempt DMA to hypervisor memory regions, potentially compromising its integrity.

Intel TXT (described in Section 6 is an additional hardware solution in charge of realizing, first, integrity checks of hypervisors and VMs at start and, second, a runtime integrity validation to discover uncontrolled changes to processes in execution. TXT can be considered a good solution to secure zero-day vulnerabilities as the Trusted Computing Group affirms.[10]

SW-based security techniques usually are enforced at multiple levels to protect VMs where tenants reside on top of VMM. Some protect VM's state from inside (enforced at the OS level), others from outside (enforced at the Hyp level). At OS level used techniques are: access controls (DAC, MAC), monitoring the Guest-OSs using Introspection techniques and Host-based Intrusion Detection Systems (HIDS).

Discretionary Access Control (DAC) models are those typically applied on conventional operating systems, when it is the owner of the data object to define permissions and policies for user accesses that will be checked by the OSs. What makes DAC less used than MAC in the cloud infrastructure is its vulnerability to viruses programs that can easily change the policies and copy information.

Mandatory Access Control (MAC) is more stringent and fits well for the cloud infrastructure case where the level of security must be higher. With this type of access control a system administrator defines rules and policies that will be enforced by the OS and end users cannot change it [30]. Real examples of MAC access controls are: the Mandatory Integrity Control implemented in Windows; AppArmor, a Linux security module that provide the mechanisms of MAC; and finally TrustedBSD, the one used by Mac OSx.

MAC solutions not only operate at Guest-OS level but also at Hypervisor-level. While a traditional OS-layer MAC specifies what actions a process can perform on files, Hyp-Level MAC specifies what actions a VM can perform on another VM. An administrator defines a specific set of allowed operations and the hypervisor enforces those policies. Real examples are Xen Security Modules (XSM) and sVirt.

For what concern HIDS, it consists of a software built to monitor OS behavior through an analysis of events information collected during the system activity by host-based agents (known as sensors), system logs, or audit trails. HIDS traces the access to sensitive information in terms of who accessed what. However HIDS are not free from attacks. Because they belong to the Guest OSs, HIDS are vulnerable to usual attacks to hosts machine such as viruses. New solutions reported in Section 8 try to find a solution to HIDS vulnerabilities.

A technique used in the community to externally monitor the runtime state of a system-level virtual machine is Virtual Machine Introspection (VMI). Monitors can be placed in another virtual machine, within the hypervisor, or within any other part of the virtualization architecture. They can detect abnormal behavior, which may be due to an intrusion. An example is LibVMI,[11] an introspection library focused on reading and writing from VMs, accessing CPU registers, pausing and unpausing a VM. Lib VMI is supported by KVM and Xen hypervisors.

Following we describe security mechanisms deployed in the three principal hypervisors used by three major CPs: Xen, VMWare vSphere, KVM.

7.1. Solutions adopted by commercial hypervisors

VMware proposes vShield Endpoint[12] as a solution to protect guest VMs from viruses and malware. It is an agentless antivirus/anti-malware solution, which results in an improvement of security and performance. The difference with traditional solutions is that antivirus and anti-malware functions, usually handled by agents in each VM, are moved to a central security virtual machine. This architecture frees up system resources, eliminates the risk of attack directed to antivirus and solves the storm issue which is a performance degradation that occurs when antivirus software simultaneously scans multiple VMs on a single physical host.

Security of VMWare's Hypervisor is well deepened.[13] This is known as vSphere based and it is a bare-metal hypervisor. The security infrastructure provides mechanisms to enforce virtualization layer security, which includes:

Memory isolation: The VMM gives the illusion to each virtual machine that is using a zero-based address space, as it would be in a real hardware. This is done using an extra level of address translation and so there will be three different types of addresses, from high above: virtual, physical and machine addresses. VM instructions with the aim of manipulate guest OS page tables contents are intercepted in order to prevent unwanted updates in the hardware memory. The VMM, in fact, is interposed in the physical-machine translation. To protect against buffer overflow attacks, vSphere uses two techniques: 1) Address Space Layout Randomization (ASLR) that randomizes where core kernel modules are loaded into memory

[10] http://www.trustedcomputinggroup.org
[11] https://code.google.com/p/vmitools/wiki/LibVMIIntroduction
[12] http://www.vmware.com/files/pdf/techpaper/vShield-Edge-Design-Guide-WP.pdf
[13] http://www.vmware.com/files/pdf/techpaper/vmw-wp-secrty-vsphr-hyprvsr-uslet-101.pdf

Please cite this article as: L. Coppolino et al., Cloud security: Emerging threats and current solutions, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.03.004

2) CPU NX/XD that marks specific areas of memory as non-executable where only data and not instructions reside. The processor will then refuse to execute any code residing in this area. In this way it is possible to protect the system from code injections.

Device isolation: At the hardware level all DMA transfers and device interrupts are virtualized and isolated from other VM in order to prevent unwanted memory accesses. VMWare takes advantage of a mechanism provided by Intel-VT: DMA-Remapping, a hardware solution to improve security of devices isolation located between DMA capable peripheral I/O devices and the computer's physical memory. DMA-remapping translates the address of the incoming DMA request to the correct physical memory address and perform checks for permissions to access that physical address, based on the information provided by the system software. This permits to restrict access to protection domain's physical memory from I/O devices not assigned to it.

Network isolation: Virtual networks are necessary to create a communication between VMs and the rest of the network. The isolation of virtual switch, which is the medium through which the VMs on the same host can communicate, is implicitly granted thanks to its type of operations. In fact, first, ESXi does not provide a path for network data between virtual switches, second, the virtual switches cannot share physical Ethernet adapters and, third, each virtual switch has its own forwarding table with the impossibility of port pointing among different tables. When the virtual switch is shared among multiple VMs who care for the isolation is the virtualized network controller (vNIC)-level firewall.

A remarkable example comes from Xen Hypervisor used by many Cloud Providers as Amazon AWS. It is a bare-metal hypervisor equipped with a particular VM with the most privileged domain known as Dom0, which is assigned to manage the hardware resources and to launch unprivileged domains.

A brief description of the security features provided in the Xen system as the Xen Security Module (XSM) is described below.[14] Security mechanisms can protect four sensitive surfaces of Xen: network, boot manager, qemu, and hypercall layer.

Network: The network surface embraces the hardware driver for the network interface card, the bridging software, and the network backend. An attacker may exploit driver bugs and gain access to the Dom0 kernel. The solution is to make use of an unprivileged VM that provides only driver access to guest VMs, the driver domain; this can limit the range of action of the adversary thanks to its limited scope.

Boot manager: In Xen the PyGrub is used as boot loader. It starts user's domains (domU) reading the kernels in the root directory of a guest's VM file system and passing the selected kernel to the domain builder, which resides in Dom0. Bugs in file system parser, in domain builder, or in menu parser can open breaches in the system. One obvious solution could be using fixed kernels and so to disable the possibility for users to select among multiple kernels. Another possibility is to make use of PVGrub, a version of grub ported to run on minios, which, unlike PyGrub, it runs an adapted version of the grub boot loader inside the created domain itself rather than in the Dom0 domain. Because PVgrub runs inside the guest context, it has no more privileges than the guest already has; there is therefore no benefit to attacking it.

Qemu: Qemu aims at providing an emulated hardware for user's virtual machines. It resides in Dom0 and has the permission of reading and writing from any VM, a clear lack of security. Just think about a situation of security breaching through hardware devices interfaces. This could open many other attack surfaces to an attacker. The solution provided in the Xen project is using device model stub domains that is a distinct domain with a minimal OS used only to execute Qemu.

Hypercall layer: Xen project tries to counter attack that use hypercalls with a security feature called FLASK (FLux Advanced Security Kernel) provided in the Xen Security Module (XSM).[15] It is the Xen implementation of Hypervisor-level MAC controls. Its aim is to restrict hypercalls with specific policies to those needed by a particular guest and allows more fine-grained granting of privileges. The XSM permits to define rules of interaction among VMs, the hypervisor itself and other resources as memory and devices.

Lastly, Kernel-based Virtual Machine (KVM), a Bare-Metal hypervisor, is a virtualization solution based on Linux on x86 hardware containing virtualization extensions (Intel VT or AMD-V) used in Google Cloud. It incorporates the Secure Virtualization (sVirt) technology, which is the application of SELinux controls to provide processes (and so VMs) isolation through an implementation of Mandatory Access Controls at the hypervisor-level (similar to XSM). SELinux marks each virtual machine process with a label and define for this VM a security boundary that will be monitored by the Linux kernel in order to restrict accesses to resources outside the boundary.[16] KVM supports also the introspection libraries LibVMI.

8. Research trends and upcoming technologies

One of the most investigated drawbacks of cloud technology is the lack of control over data that is handed over to the cloud provider. This is the main hazard leading to issues IS3, IS4, and IS5, and possibly the main impairment to massive cloud adoption. In particular, since the user has no access to the physical systems, he/she has to rely solely on the infrastructure provider for what concerns data security. In the past years, a promising approach seemed to be Trusted Computing (TC). TC exploited the features of a Trusted Platform Module (TPM) to provide integrity of the software stack (particularly,

[14] http://wiki.xenproject.org/wiki/Securing_Xen
[15] http://wiki.xen.org/wiki/Xen_Security_Modules_:_XSM-FLASK
[16] http://libvirt.org/drvqemu.html

Please cite this article as: L. Coppolino et al., Cloud security: Emerging threats and current solutions, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.03.004

the VM layer). An example of this approach is Intel Trusted Execution Technology (TXT) technology. While Intel TXT provided a trusted way of loading and executing the Virtual Machine Monitor (VMM) or the OS kernel, it had serious limitations [35]. Most importantly, it was found to be useless against physical attacks to the data center perpetrated by the infrastructure owner. Today the key challenge for cloud security research is overcoming such limitations, i.e., finding mechanisms that can effectively protect users against attacks by the infrastructure owner/operator.

Again, one of the most promising approaches comes from hardware manufacturers, and most notably Intel Software Guard Extensions (SGX). SGX is an Instruction Set Architecture (ISA) extension that enables the execution of instructions in a protected memory area, called secure enclave. By doing so, SGX empowers users to enforce the security of their applications and data without having to trust the cloud operator. SGX can be considered as an evolution of Intel TXT. Although SGX has not been released yet, expectations are very high, as demonstrated in [15,34]. Since an enclave is a protected area in the application's address space, it provides confidentiality and integrity even in the presence of privileged malware. Attempts of accessing the enclave memory area by software that is not resident in the enclave itself – including privileged software, such as virtual machine monitors, BIOS, or the operating system – are prevented. An important feature of this new architecture is hardware-based attestation of the software running inside an enclave. This can prove its identity to a remote party and be securely provisioned with key and credentials. A possible trend for future research is to extend the enclave concept from a local to a distributed setup, where multiple enclaves would interact across distinct cloud data centers using secure communication channels. An important advantage of the secure enclaves approach is that it enables end-to-end encryption of data, that is never transmitted in the clear and thus cannot be accessed nor modified by attackers or even system administrators or cloud providers.

A similar feature is brought about by another promising research area, namely: Homomorphic Cryptography [24]. This represents a branch of cryptography that allows computation on encrypted data without knowledge of the private key and without any data decryption. Homomorphic encryption systems are classified as full schemes (that support both addition and multiplication) and partial schemes (that only support one of the two operations). While extremely promising, homomorphic encryption has major drawbacks, and in particular: i) the set of operations that can be implemented is very limited, and ii) the computation on encrypted data has a non-negligible overhead. Some researches focus on the possibility of parallelizing the operations, to improve the performance [27]. However, this area of research is still in an embryonic stage.

Another research trend that is worth mentioning comes from the usage of containers. Containers are used to abstract applications from the underlying OS, thus enabling faster development and easier deployment. One of the most famous and appreciated containers is Docker. The literature about containers is still poor, however their diffusion is demonstrated by the support provided by big players such as Amazon,[17] Google,[18] and OpenStack.[19] At least in principle, it is possible to use containers to improve the security of user applications. It should be noted however that we are still in the early stage of containers, and it is thus very possible that they could introduce new security issues (especially because of their limited isolation capabilities).

9. Summary and conclusions

The paper surveyed the most widely used mechanisms that are currently available for protecting the different layers of a cloud stack. The study was conducted analyzing one by one three distinct attack vectors. For each vector, we discussed: implementable attacks, related solutions, and new security technologies being developed by the research community and/or the industry for addressing open issues. Issues are classified in five categories, as listed in Table 1. The paper also reviewed examples of features provided by commercial products and measures taken by CPs. Tables 2–4 summarize the approaches taken by: Amazon AWS, VMWare vCloud, Microsoft Azure, and OpenStack. We provided an exhaustive reference on technical maturity of security mechanisms. The analysis shows that Amazon AWS and VMWare vCloud provide a number of products and mechanisms, that are offered either directly by the CPs or by third-parties. It turns to that while some protection techniques have already reached a certain level of technical maturity, others are still in their infancy and are not suitable for deployment in an operational setup. Many techniques are available to address IS1 and IS2, but they very much need to be improved. Not surprisingly (since defense mechanisms in this field have already been researched for years, even before the birth of the cloud itself), countermeasures at the network-level against attacks launched through the network are the most mature. Instead, approaches at the application-level – to address IS3 and IS4 – are still being improved, and many researchers are focusing on them. There are many works dedicated to investigating new solutions that could protect against hypervisor-based attacks. One of the most unexplored issue is IS5. The research community is now realizing this, and more and more attention is being paid to hardware solutions to specifically address IS5. Products such as SGX can bring about a leap forward in this direction.

[17] https://aws.amazon.com/it/blogs/aws/cloud-container-management/
[18] https://cloud.google.com/container-engine/
[19] https://www.openstack.org/assets/pdf-downloads/Containers-and-OpenStack.pdf

Please cite this article as: L. Coppolino et al., Cloud security: Emerging threats and current solutions, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.03.004

Table 3
Commercial solutions to secure hardware vector.

Attacks	Amazon AWS	OpenStack	Azure	VMWare cloud
Time-, Trace- and, Access-driven Cache-Based Side-Channel Attacks, Prime+Trigger+Probe Attack	–Cisco Physical Security –AWS CloudHSM –Intel AES-NI –KeyNexus –SafeNetHSM –HP ESKM -HP Atalla Cloud Encryption	–Barbican HSM support –Intel AES-NI	–Azure Trust Center –Intel AES-NI –Azure Key Vault	–VMWare vCloud Air Data Center Security –VMWare Air Physical Infrastructure Security –Intel AES-NI –HP ESKM –HP Atalla Cloud Encryption
Boot Integrity Attacks	–Intel TXT –HyTrust	–OpenAttestation with Intel TXT		–Intel TXT –HyTrust
Data Directed Attacks	–SafeNet ProtectV –CipherCloud –KeyNexus –HyTrust DataControl			
		–OpenStack Raksha –OpenStack server side encryption	–CloudLink –Secure VM –HyTrust DataControl	–CipherCloud gateway –vSphere Data Protection

Table 4
Commercial solutions to secure hypervisor vector.

Attacks	Amazon AWS (Xen)	OpenStack (KVM)	Azure (Hyper-V)	VMWare Cloud(vSphere)
Code Injection	–Xen Security Module (XSM) FLASK –Virtual,Gateway –IntelVT and AMD V –AlertLogic HIDS	–AppArmor MAC –Secure Virtualization (sVirt) –IntelVT and AMD V –OSSEC HIDS	–McAfee MOVE AV –Hyper-V Secure Boot –5nine Cloud Security –BitDefender Gravity Zones –BitDefender SVE –Intel VT and AMD V	–VMWare vSphere security (ASLR, CPU NX/XD, DMA-Remapping, vNIC-level firewall,VMI) –McAfee MOVE AV –KasperSky Sec for Virtualization IntelVT and AMD
Rootkit (BluePill, SubVirt, Vitriol, DKSM)	–Intel TXT –BitDefender SVE –Zero-Footprint GuestMemory introspection	–Lib VMI –Intel TXT	–Live CloudKd Memory Introspection MoonSol	–Juniper vGW –Sophos AV –IBM IPS

Acknowledgments

The research leading to these results has received funding from the European Commission within the context of the Horizon2020 Programme under Grant Agreement No. 645011 (Secure Enclaves for REactive Cloud Applications, SERECA project).

References

[1] Takahashi T, Blanc G, Kadobayashi Y, Fall D, Hazeyama H, Matsuo S. Enabling secure multitenancy in cloud computing: challenges and approaches. In: Future internet communications (BCFIC), 2012 2nd Baltic congress on, Vilnius; 2012. p. 72–9. doi:10.1109/BCFIC.2012.6217983.
[2] Zhou M, Zhang R, Xie W, Qian W, Zhou A. Security and privacy in cloud computing: a survey. In: Semantics knowledge and grid (SKG). 2010 sixth international conference on, Beijing; 2010. p. 105–12. doi:10.1109/SKG.2010.19.
[3] Panth D, Mehta D, Shelgaonkar R. A survey on security mechanisms of leading cloud service providers. Int J Comput Appl July 2014;98(1):34–7.
[4] Akinbi A, Pereira E, Beaumont C. Evaluating security mechanisms implemented on public platform-as-a-service cloud environments case study: Windows azure. In: Internet technology and secured transactions (ICITST), 2013 8th international conference for, London; 2013. p. 162–7. doi:10.1109/ICITST.2013.6750183.
[5] Ristov S, Gusev M. Security evaluation of open source clouds. EUROCON, 2013. Zagreb: IEEE; 2013. p. 73–80. doi:101109/EUROCON20136624968.
[6] Booth G, Soknacki A, Somayaji A. Cloud security: attacks and current defenses. In: 2013 8th annual symposium on information assurance, Albany, NY; June 2013.
[7] Joseph AO, Jaspher G, Kethrine W. Security of real time cloud service providers: a survey. In: Electronics and communication systems (ICECS). 2014 international conference on, Coimbatore; 2014. p. 1–5. doi:10.1109/ECS.2014.6892555.
[8] Weiss M, Heinz B, Stumpf F. A cache timing attack on AES in virtualization environments. In: Financial cryptography and data security: 16th international conference, FC 2012, Kralendijk, Bonaire; Februray 27-March 2 2012.Revised Selected Papers
[9] Goel R, Garuba M, Girma A. Cloud computing vulnerability: DDos as its main security threat, and analysis of IDS as solution model. In: Information technology: new generations (ITNG). 2014 11th international conference on, Las Vegas, NV; 2014. p. 307–12. doi:10.1109/ITNG.2014.77.
[10] Grance T., Jansen W.. Guidelines on security and privacy in public cloud computing. NIST.
[11] Fernandes DA, Soares LF, Gomes JV, Freire MM, Inácio PR. 2014. Security issues in cloud environments: a survey. Int J Inf Secur April 2014;13(2):113–70.
[12] Nanavati M, Colp P, Aiello B, Warfield A. 2014. Cloud security: a gathering storm. Commun ACM May 2014;57(5):70–9. http://dx.doi.org/10.1145/2593686.
[13] Deshmukh RV, Devadkar KK. Understanding DDos attack & its effect in cloud environment. Procedia Comput Sci 2015;49:202–10.ISSN 1877-0509
[14] Kandias M, Virvilis N, Gritzalis D. The insider threat in cloud computing. Crit Inf Infrastruct Secur 2013;vol. 6983:93–103. doi:10.1007/978-3-642-41476-3_8.

Please cite this article as: L. Coppolino et al., Cloud security: Emerging threats and current solutions, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.03.004

[15] Kim S, Shin Y, Ha J, Kim T, Han D. A first step towards leveraging commodity trusted execution environments for network applications. Proceedings of the 14th ACM workshop on hot topics in networks (HotNets-XIV). New York, NY, USA: ACM; 2015. p. 7 http://dxdoiorg/101145/28340502834100. Article 7
[16] Awadhi EA, Salah K, Martin T. Assessing the security of the cloud environment. GCC conference and exhibition (GCC), 2013 7th. Doha: IEEE; 2013. p. 251–6. doi:101109/IEEEGCC20136705785.
[17] Perez-Botero D, Szefer J, Lee RB. Characterizing hypervisor vulnerabilities in cloud computing servers. In: Proceedings of the 2013 international workshop on security in cloud computing (Cloud Computing '13). New York, NY, USA: ACM; 2013. p. 3–10.
[18] AlJahdali H, Albatli A, Garraghan P, Townend P, Lau L, Xu J. Multi-tenancy in cloud computing. In: Service Oriented System Engineering (SOSE). 2014 IEEE 8th international symposium on, Oxford; 2014. p. 344–51. doi:10.1109/SOSE.2014.50.
[19] Szefer J, Lee RB. A case for hardware protection of guest VMs from compromised hypervisor in cloud computing. In: Distributed computing systems workshops (ICDCSW). 2011 31st international conference on, Minneapolis, MN; 2011. p. 248–52. doi:10.1109/ICDCSW.2011.51.
[20] Zhang Y, Juels A, Reiter M, Ristenpart T. Cross-VM side channels and their use to extract private keys. Proceedings of the 2012 ACM conference on computer and communications security (CCS '12). New York, NY, USA: ACM; 2012. p. 305–16 http://dxdoiorg/101145/23821962382230.
[21] Kim T, Peinado M, Mainar-Ruiz G. Stealthmem: system-level protection against cache-based side channel attacks in the cloud. In: Proceedings of the 21st USENIX conference on security symposium (Security'12). USENIX association, Berkeley, CA, USA; 2012.11-11
[22] Rocha F, Gross T, van Moorsel A. Defense-in-depth against malicious insiders in the cloud. In: Cloud Engineering (IC2E). 2013 IEEE international conference on, Redwood city, CA; 2013. p. 88–97. doi:10.1109/IC2E.2013.20.
[23] Jensen M, Schwenk J, Gruschka N, Iacono LL. On technical security issues in cloud computing. Proceedings of the 2009 IEEE international conference on cloud computing (CLOUD '09). Washington, DC, USA: IEEE Comput Soc; 2009. p. 109–16 http://dxdoiorg/101109/CLOUD200960.
[24] Moore C, O'Neil M, O'Sullivan E, Doroz Y, Sunar B. Practical homomorphic encryption: a survey. In: Circuits and systems (ISCAS). 2014 IEEE international symposium on, Melbourne VIC; 2014. p. 2792–5. doi:10.1109/ISCAS.2014.6865753.
[25] Rani DR, Venkateswarlu S. Software and hardware defence methods against cache-based side channel attacks. Int J Manage Inform Technol Eng May 2014;Vol 2(Issue 5).
[26] Ristenpart T, Tromer E, Shachem H, Savage S. Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds. Proceedings of the 16th ACM conference on computer and communications security (CCS '09). New York, NY, USA: ACM; 2009. p. 199–212 http://dxdoiorg/101145/16536621653687.
[27] Hayward R, Chiang C-C. Parallelizing fully homomorphic encryption. Proceedings of the 2014 international symposium on computer, consumer and control (IS3C '14). Washington, DC, USA: IEEE Computer Society; 2014. p. 721–4 http://dxdoiorg/101109/IS3C2014192.
[28] Bahram S, Jiang X, Wang Z, Grace M, Li J, Srinivasan D, et al. DKSM: subverting virtual machine introspection for fun and profit. Proceedings of the 2010 29th IEEE symposium on reliable distributed systems (SRDS '10). Washington, DC, USA: IEEE Computer Society; 2010. p. 82–91 http://dxdoiorg/101109/SRDS201039.
[29] Zargar ST, Joshi J, Tipper D. A survey of defense mechanisms against distributed denial of service flooding attacks. IEEE Commun Surveys Tutor Fourth Quarter 2013;vol. 15(no. 4):2046–69. doi:10.1109/SURV.2013.031413.00127.
[30] Punithasurya K, Jeba Priya S. Analysis of different access control mechanism in cloud. Int J Appl Inf Syst (IJAIS) 2012.
[31] Badis H, Doyen G, Khatoun R. Understanding botclouds from a system perspective: a principal component analysis,. In: Network operations and management symposium (NOMS), 2014 IEEE, 1; May 2014. p. 5–9.
[32] Weis S. Protecting data in-use from firmware and physical attacks. Palo Alto, CA: BlackHat; 2014.
[33] Neelima S, Lakshmi Y. A review on distributed cloud intrusion detection system. Int J Adv Technol Eng Res (IJATER) 2013.
[34] Baumann A, Peinado M, Hunt G. (microsoft research) Shielding applications from an untrusted cloud with Haven. ACM Trans Comput Syst August 2015;33(3):26. http://dx.doi.org/10.1145/2799647.Article 8
[35] Wojtczuk R., Rutkowska J.. Attacking intel trusted execution technology. 2009.

Please cite this article as: L. Coppolino et al., Cloud security: Emerging threats and current solutions, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.03.004

Luigi Coppolino is an assistant professor at the University Parthenope, Italy. His research activity mainly focuses on dependability of computing systems, critical infrastructure protection, and information security. He has several scientific publications and was/is principal investigator of many European Commission funded research projects.

Salvatore D'Antonio is currently an assistant professor at the University of Naples "Parthenope", Italy. He is an expert in network monitoring, intrusion detection systems, and critical infrastructure protection. He was the technical coordinator of the FP7 EU INTERSECTION project and the project coordinator of the INSPIRE and the INSPIRE-INCO projects. He actively participates to IETF standardization activities.

Giovanni Mazzeo is currently a PhD student at the Department of Engineering of the University of Naples "Parthenope". His research activity mainly focuses on security and dependability of computer systems. He is actively involved in a European project on cloud security namely SERECA.

Luigi Romano is a full professor at the University Parthenope in Italy. His research focuses on system security and dependability. He has worked extensively as a consultant for industry leaders in the field of safety critical computer systems in Europe and the US. He cooperates with ENISA (In particular the PROCENT and the Research Policy Recommendations expert groups).

Please cite this article as: L. Coppolino et al., Cloud security: Emerging threats and current solutions, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.03.004