WHIOCE PUBLISHING PTE. LTD.

RESEARCH ARTICLE

Cloud computing security: protecting cloud-based smart city applications

Alkiviadis Giannakoulias*

European Dynamics SA, 209, Kifissias Av. & Arkadiou Str., 15124 Maroussi, Athens, Greece

Abstract: Data security is a major concern in cloud computing environments as they provide much scope for intruders to attack. Data centres in cloud environments hold valid information that end-users would conventionally have stored on their computers. Moving information towards centralized services may have an adverse effect on the security of users' interactions with files kept in cloud cupboard spaces[1], for example accidental or deliberate alterations or deletions of information from the cloud server by the Cloud Service Provider (CSP). This necessitates the deployment of some sort of mechanism to ensure the safety of information integrity[2]. Public sector organizations have much to gain by adopting a cloud computing approach to service delivery in their ICT environments. However, these benefits must be reaped without compromising core requirements and institutional values.

This paper focuses on the security issues that may arise when public sector organizations consider transitioning to an Open Source Software (OSS) Infrastructure as a Service (IaaS) Cloud Infrastructure (OpenStack), although the same issues are likely to be found in other OSS cloud computing software like Apache CloudStack[3], Eucalyptus[4], and OpenNebula[5]. We examine legal implications, regulatory and standards compliance, new attack vectors resulting from vulnerabilities coming from virtualization technologies, data integrity issues such as encryption and access controls, and security checks to be performed on the services prior to their movement to the cloud. In addition, some of the most important security threats in cloud computing are presented, followed by key recommendations on how to address them, namely security standards and certifications, service provider auditing, secure APIs, transport layer protection, authentication and encryption key management, and cloud service agreements.

Keywords: STORM CLOUDS, OpenStack, cloud security, certification, auditing, security by exception and group management, introspection, host-based security, firewall, network security architecture, transport layer protection, access right management, configuration management, patch management, cryptography

*Correspondence to: Alkiviadis Giannakoulias, European Dynamics SA, 209, Kifissias Av. & Arkadiou Str., 15124 Maroussi, Athens, Greece; Email: Alkiviadis.Giannakoulias@eurodyn.com

Received: February 4, 2016; **Accepted:** March 11, 2016; **Published Online:** May 9, 2016

Citation: Giannakoulias A, 2016, Cloud computing security: protecting cloud-based smart city applications. *Journal of Smart Cities*, vol.2(1): 66–77. http://dx.doi.org/10.18063/JSC.2016.01.007.

1. Introduction: Security Definition

Cloud computing security is an evolving subdomain of information security and refers to a broad set of policies, technologies, and controls deployed to protect data, applications, and the associated infrastructure[6].

There is a number of security concerns associated with cloud computing; they can be broadly classified into two categories, namely issues faced by Cloud Service Providers (CSP) and those faced by Cloud Service Consumers (CSC). Providers must ensure that their infrastructure is secure and clients' data and applications are protected; consumers, on the other hand, must ensure that their provider has taken appropriate security measures to protect their information[6].

Current Cloud delivery models (whether implemented on an Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), or Software-as-a-Service (SaaS) model) are ruled by Service Level Agreements (SLA) that normally define mutual supplier and user expectations and obligations. The central idea behind such models is that the consumer ought to trust the supplier[7].

1.1 Delineation of Responsibility

Venturing into a public cloud environment, especially via an IaaS model, security becomes a shared responsibility. Although there are certain measures which a cloud provider will apply to ensure that Virtual Machines (VM) stay secure, a considerable number of tasks are left in the hands of the tenant (cloud consumer). Figure 1 shows the cloud stack[8], presented as an OSI model for the cloud. As can be seen, responsibility for security is equally split between tenant and provider.

To make our notions more precise, some definitions and explanations on the IaaS model are in order:

- **Provider** is the person/organization that has built the cloud and offers the relevant service.
- **Tenants** are those asking the provider for access to that service.
- The **red line** identifies where the delineation of responsibility is depending on the cloud service model used.
- **Facility**, found at the bottom of the stack, refers to installations, such as buildings, doors that lock and other related objects. As tenants have no control here, responsibility for those lies with the provider.
- **Network**, refers to connected physical entities such as wires, cables, switches, routers, located inside the Facility.

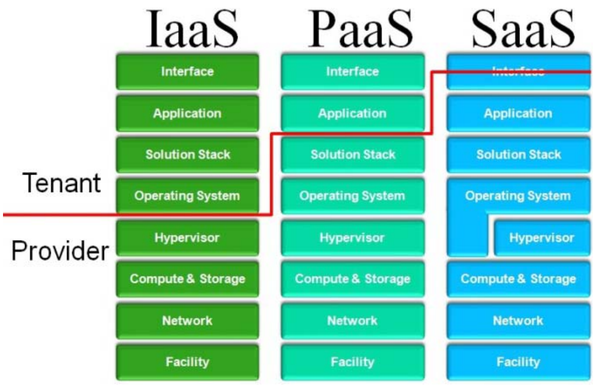

Figure 1. Cloud service models delineation of responsibility.[9]

- **Compute & Storage** refers to CPUs, motherboards, hard drives, etc.
- **Server virtualization** is the technology for creating and managing VMs, implemented by hypervisors.
- **Virtual Machine** refers to the software container used to emulate hardware for the Operating System (OS) running inside of it.
- **Solution stack** refers to any type of application language running, such as .Net, Perl, Python, or others.
- **Applications** refer to specific for purpose software, for example web applications.
- **Interface** refers to implemented GUIs, graphic web interfaces, or even sets of RESTful APIs.

1.2 Cloud Computing Security Threats

Some of the most important security threats in cloud computing include[10–12]:

1. **Ease of use**. The simplicity of cloud services (i.e., any resource that is provided over the Internet[13]) is appealing to attackers for malicious purposes like spamming, malware distribution, command-and-control servers, distributed denial-of-service (DDoS) attacks, password/hash cracking, etc.

2. **Vulnerable data transmission**. Since data can be intercepted by man-in-the-middle attacks, data transferred from clients to the cloud needs to be properly encrypted by using Secure Socket Layer (SSL)/Transport Layer Security (TLS).

3. **Insecure APIs**. Since cloud services are exposed by APIs, it is imperative for the CSPs to secure them. The reason is that attackers can manipulate data, with the right authentication/authorization token.

4. **Malicious insiders**. CSP personnel having complete access to enterprise data and resources can, undetected, gather confidential information. Hence, CSPs should employ security measures in place to track employee actions such as data viewing.

5. **Shared technology issues**. Shared infrastructure resources amongst various tenants can lead to vulnerabilities, such as hypervisor exploitation, VM sandbox break-out, unauthorized access to shared data through side-channel attacks, and others.

6. **Virtualization technology issues**. Virtualization is a critical part of cloud computing with virtualized operating systems being the backbone of IaaS[14]. They provide an important layer of abstraction from physical hardware, thus enabling the elasticity and

resource pooling commonly associated with cloud. Given its importance, it is only natural to ask: “what constitutes our biggest concern regarding security in virtualization?” The answer is the hypervisor, which ties together all our operating systems. Traditionally, in the past, machines were plugged to a switch; the only way for machine A to interact with machine B was via the switch itself, in other words via the IP stack, i.e., via network-based communication. In virtualisation, a hypervisor manages all our VMs containing operating systems, thus creating a software bridge among them. As shown in Figure 2, this means that we have to secure the two connection points (depicted in red).

From a theoretical perspective, the hypervisor appears as a less secure solution. However, from a practical perspective, a hypervisor can augment our security.

7. **Data loss**. Data stored in the cloud could be lost due to a number of reasons, namely hard drive failure, CSP going out of business, accidental data deletion by a CSP employee, data-theft by an attacker, etc. The best way to protect against such threats is via data backups for subsequent restore.

8. **Data breach**. Side-channel attacks refer to a situation where VMs running on the same physical host can access the data of another VM, leading to a data breach.

9. **Insecure or incomplete data deletion**. Requests to delete cloud resources may not result in actual data wiping. Data deletion could be incomplete either because extra copies are being stored but not deleted, or because the physical resources are being used by other clients.

10. **Security incident handling**. CSCs rely on CSPs to handle detection, reporting and management of security breaches.

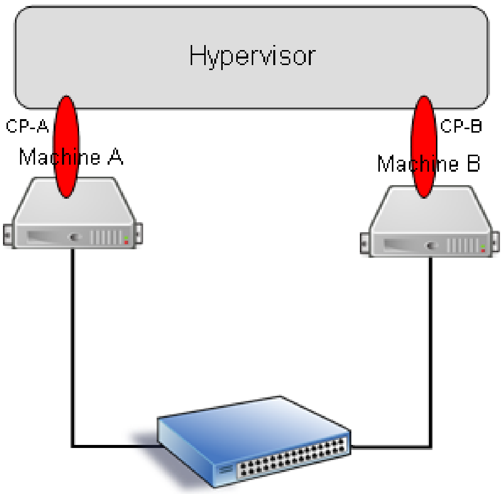

Figure 2. The lack of an air gap.

11. **Data protection**. Issues like exposure or release of sensitive data as well as loss or unavailability of data are major data protection risks for both cloud consumers and providers. Cloud consumers have no effective information about the data handling practices of the CSP so as to ensure that data is handled in a lawful way.

12. **Account/service hijacking**. If cloud access is only password protected, an attacker who knows the password will have equally easy access. It is therefore better to use two-factor authentication when available.

13. **Unknown risk profile**. Lack of knowledge of a cloud provider’s security protocols and policies can contribute to making it harder to “calculate” a risk profile. Security by obscurity may be a low effort but it can result in unknown exposures[15].

14. **Denial of service (DoS)**. Cloud services can get disrupted by attackers issuing a DoS attack against the cloud service rendering it inaccessible.

15. **Lack of understanding**. Users should understand the cloud security threats in order to properly defend against them. This means that organizations should invest time and resources in education and training before moving to the cloud.

16. **Access privileges**. CSPs should be able to demonstrate that they enforce adequate hiring processes, oversight, and access controls to enforce administrative delegation.

17. **Regulatory compliance**. Cloud consumers are accountable for their own data even when this is in a public cloud; they should ensure that CSPs are ready and willing to undergo audits. Cloud consumers should use cloud computing services in a responsible way and should ensure that the CSP has appropriate certifications in place.

18. **Data segregation**. Most public clouds are shared environments; it is critical to ensure that hosting providers can guarantee complete data segregation for secure multi-tenancy.

19. **Loss of governance**. Public cloud consumers unavoidably tend to handle control to CSPs over a number of issues affecting security without at the same time being able to impose strict SLA commitments on the part of the CSP. This creates gaps in their security defences.

20. **Responsibility ambiguity**. Responsibility for security aspects is spread across both the cloud con-

sumers and CSPs, potentially resulting in vital parts being left unattained in case of failure to allocate responsibility clearly.

21. **Vendor lock-in**. Use of proprietary services from a specific CSP that does not support portability of applications and data to other CSPs can lead to vendor lock-in and higher risks of data and service unavailability.

22. **Authentication and Authorization**. Assurance regarding the identity of users (employees, contractors, partners, and customers) is important as resources are accessed from anywhere. Strong authentication and authorization becomes a critical concern.

2. Security Recommendations

According to David S. Linthicum, senior vice president of Cloud Technology Partners, CSCs are responsible for securing their data and for providing the security requirements that the CSPs should meet via appropriate technical means[16]. CSCs must determine their security requirements and must map those to the appropriate technology.

What we should keep in mind that the level of security provided in the cloud environment should be equal to or better than the security provided by traditional IT environments, otherwise we could be facing higher costs and potential loss of information. This would eliminate any potential benefits of cloud computing.

2.1 Service Provider Certification and Auditing

Certification of cloud computing services allows CSPs to show their customers that they meet certain standards, for example on network and information security.

Certification provides assurance to CSCs that their critical security requirements are being met. Therefore, they should identify which security certifications are important to them and to insist that their CSPs demonstrate their conformance.

Regrettably, many providers have taken a **security through obscurity** approach—they don't want to talk about what security controls they have put in place. The idea is that if they don't talk about their controls, then it becomes harder for an attacker to break in.

Moreover, as there is no single accepted security assessment program for CSCs that can help them evaluate the types of controls CSPs have in place, making risk assessments can be difficult. Programs like the United States federal government's **FedRAMP**[17] and the United Kingdom government's **G-Cloud**[18] are only focused on government cloud use. One effort currently underway is the **Cloud Security Alliance's Open Certification Framework** (CSA OCF). CSA is the world's leading organization dedicated to defining and raising awareness of best practices to help ensure a secure cloud computing environment[19]. It comprises of the following programs[20]:

- **CSA Security Trust and Assurance Registry (STAR) Certification**[21]: The STAR Certification relies on an independent third-party assessment of a cloud provider against the ISO 27001 standard, as well as the CSA Cloud Controls Matrix (CCM).
- **CSA STAR Attestation**[22]: The STAR Attestation phase provides a report via the audit-reporting standard for customer consumption known as the Statement on Standards for Attestation Engagements Service Organization Control (SSAE SOC) 2 Report[23].
- **CSA STAR Continuous**: STAR Continuous was originally planned for release in 2015. This service will provide a scanning and monitoring console for users to remotely assess cloud providers' control statements via the CloudAudit XML-based tag format and the Cloud Trust Protocol (CTP) for data transmission and retention.

Additional frameworks are available from the **Shared Assessments Program** and the **European Union Agency for Network and Information Security** (ENISA).

Hence, CSPs should state the security validations they have obtained or they should identify and describe the security they have implemented inside their environment using the CSA STAR program. This will provide more detailed information and a better understanding of the CSP's security posture.

Even in cases where the provider under consideration is not listed in the STAR registry, there is a solution, namely the **Consensus Assessments Initiative Questionnaire** (CAIQ), downloadable from the CSA website, which the provider can be asked to complete[24].

Additionally, the Cloud Standards Customer Council (CSCC) "Security for Cloud Computing: Ten Steps to Ensure Success"[12] provides a prescriptive series of 10 steps that should be taken by CSCs to evaluate and manage the security of their cloud environment with the goal of mitigating risk and delivering an appropriate level of support.

2.2 Security by Exception and Group Management

To assist public sector organisations in deploying their servers and services to an OpenStack IaaS cloud model, we leverage automation. The automatic deployment is obtained using Heat[25]. The automation process includes: (a) The configuration of the VM hosting the application and (b) The installation and configuration of the application and its dependencies. Validation of the automation process is made in collaboration with the public sector organizations and includes functional tests in order to ensure that the deployed application performs as designed. Once the automation process is validated it results in a master VM.

When it comes to security, we need not focus as much on the instances that get generated but on that master VM, since all VMs should be identical. Even if we deploy multiple VMs serving out slightly different content, then:

- OS is going to be the same
- Patches are going to be the same
- Configurations should be the same
- Same processes will be running in memory

Rather than having to look at them as a one-off, we relate them all back to the master VM. In this way, we can change the way we go about security from the one-off security mode to **security by exception**.

Security by exception is similar to the "spot the difference game". Should one of the VMs deployed appear different compared to the master configuration, we then have a strong indication of compromise. Using this way to look at the data and managing it as a **collective group**, makes things easier. For example, we need to know if it is normal for a Web Server to serve content without HTTPS in a secure site, or whether directory listing is enabled, or whether the X-Frame-Options HTTP response header is set on all web pages returned by the application. Not having this **intimate knowledge** may not lead us to the compromised server.

Another great example of the power of doing management via collective groups is shown in Figure 4. Any patching and configuration management should not be performed at each deployed VM. If a VM is missing a patch, it is because it is also missing from the scripts implementing the automation process and thus from the master. Update the scripts, re-launch out another copy and we are good to go! Once more, rather than having to do one-off security management, we are doing security management by exception.

Now, there are two ways we can look at this. We can look at our deployed VMs and ask "Does any of them differ from the master?" But there is another way we can parse this data too. Let us say that we are missing three patches on our master; we did build the master a month ago, three patches have come out since then, and all our VMs running in the cloud are currently missing three patches. A look at Figure 4 reveals that the VM on the right is fully patched and up-to-date, despite the fact that the master is not. The conclusion is that somebody has done something to that VM.

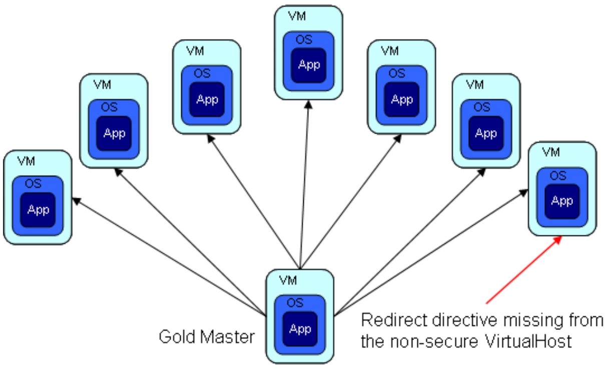

Figure 3. Spot the difference – case 1.

The idea is to compare the VMs to the master so as to investigate possible inconsistencies. In our example above, we had three missing patches on the master but would we really have cared had three missing patches appeared on all of our VM instances? The answer is yes and no. Yes, because we would want to install these three patches and burst back out again—this is an administrator's stand point. We wouldn't necessarily care from a security stand point; there is no indication that somebody has broken into the VM. If we now normalise out the three missing patches across the system and something appears to have been patched differently, then this is an indication of a serious problem. What is interesting here is that old school one-off server management would never catch such a problem—what we would have looked for would be for a patched and up-to-date machine. A one-off server, patched and up-to-date would not make us suspect that something is wrong. However, if we look at it in the context of Figure 4, then we have **an exception**. It is common practice for hackers once they break into a machine to pullout on their toolkit and patch the machine so some other script can break in exactly the same way. In other words, when all our other systems have not been patched, it is not uncommon for a single system to appear patched because of it being actually compromised. This is not something which needs

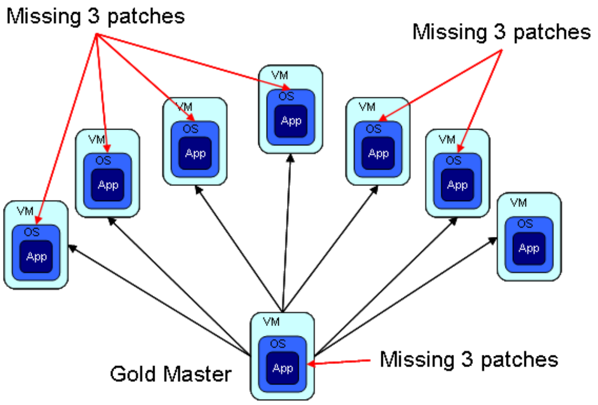

Figure 4. Spot the difference – case 3.

intimate knowledge of each VM. It just shows right up.

To summarize, the process of managing by exception:

(1) simplifies the work,

(2) offers new security controls which were not previously available.

For IaaS deployments:

- The CSP is responsible for their (physical) machines and for taking care of updating the software running on them (host OS, OpenStack, virtualization technology, etc.). As such, the selected public CSP has to confirm that they support this security feature.
- The CSC is responsible for the (virtual) machines deployed and for taking care of updating the software running on them (OS, applications, frameworks, etc.). As such we should periodically review our VMs and apply updates as soon as they are available.

2.3 Operating System Firewall

When we deploy a firewall to protect a corporate network, we typically install it as an appliance on the perimeter, as shown in Figure 5. Here, each server is connected to a unique switch, which is then routed into the firewall. The servers are:

- Logically segregated from each other, as the only routed path between networks is through the firewall
- Physically segregated from each other, as the only point of connectivity between them is through the firewall.

In other words, pull the firewall out of the mix and there is no possible way for the servers to talk to each other or to the Internet. This is one of the security benefits of physical segregation[26].

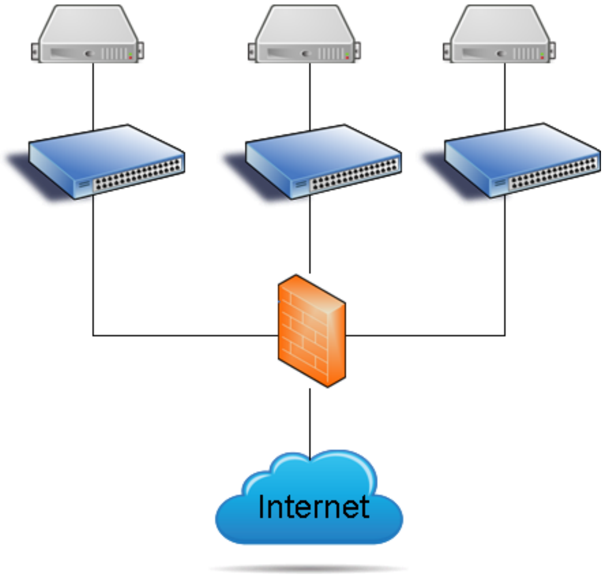

Figure 5. Classic firewall installation.

Can a virtual firewall appliance provide the same level of risk mitigation as a physical firewall appliance?

Figure 6 shows an identical configuration, only applied to an IaaS cloud with a virtual firewall appliance.

When virtual switches are deployed, we tend to think in terms of having multiple virtual switches; only one virtual switch is deployed, separated into logical partitions. Hence, apart from the risk that comes from the hypervisor (a software connection between the VMs), an additional identifiable risk comes from the virtual switch being another connection point among all the VMs. Should the virtual switch be compromised, access to the VMs (which normally would not be possible due to them being petitioned off) could be gained. If two VMs figure out a way to get connectivity

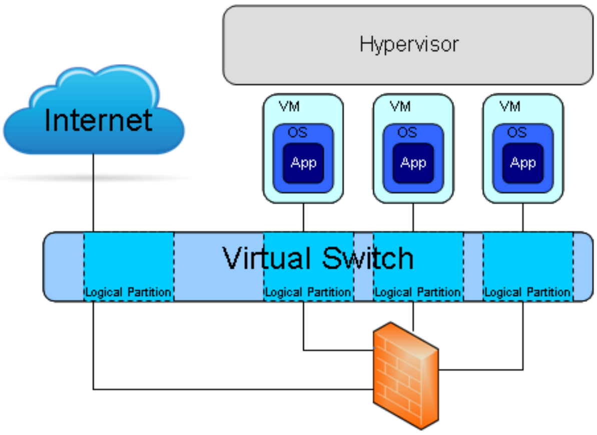

Figure 6. Virtual switch installation.

through the virtual switch, any risks that may exist within the virtual switch itself are propagated to the VMs. Here, firewall protection is not effective as these acts outside the virtual switch. By deploying a virtual appliance, we are protected from the Internet but not necessarily from other users operating in the same cloud environment.

In order to mitigate any risks coming from the virtual switch, we need to move the firewall to the other side of the switch.

One solution is to leverage a **hypervisor-based firewall**. Such firewalls are not only vendor specific but in some cases, CSPs do not even support them. Moreover, they require the use of introspection which introduces additional risks in public cloud environments as we analyse below.

Alternatively, we could leverage the **VM operating system firewall** software. Since such a solution controls traffic as it passes in and out of the VM, it is more than capable of negating any potential risks within the virtual switch. This option presents two advantages:

(a) It is the least expensive; the firewall is already present within the VM operating system.

(b) It is the most portable—if the VM is moved to another CSP, host-based firewall protection obviously moves with it.

The main disadvantage is that it introduces an additional point of management.

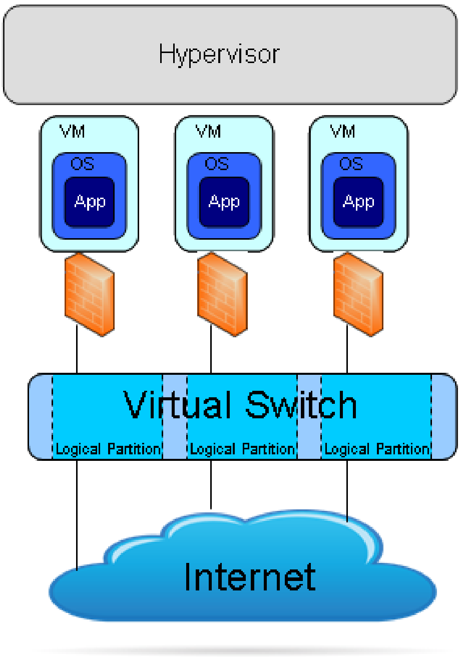

Figure 7. Hypervisor based firewall.

2.4 Introspection

Introspection is a security tool that we can leverage via the hypervisor in order to implement security on each VM. The capability of introspection can be leveraged for a wide range of security applications such as malware control, data loss prevention, firewalling between VMs, network intrusion detection between VMs, and forensics. If a hypervisor can see and interact with all the VMs running within a certain platform, it almost makes sense to ask whether it can be leveraged to implement security. With that being said, a hypervisor based solution secures ONLY a specific cloud group; if the VM is to be migrated to another, CSP protection is lost, resulting in vendor lock-in. Introspection is thus similar to a firewall—if a machine stays on the one side of the firewall, it can be protected; if it is moved, protection is not there anymore.

Despite some advantages, additional security issues introduced by introspection can elevate risk against an environment. The first is hypervisor bloat. Much of hypervisor security is predicated on keeping the code as small as possible. The fewer the lines of code, the less likely an attacker will find a problem which can be leveraged for malicious gain. By adding introspection capabilities to the hypervisor we increased the amount of code being processed[27].

Introspection also enlarges the attack surface of the hypervisor; as the hypervisor is made to interact more with each VM, increased interaction with an untrusted source results in increased probability of compromise.

A good example is Network-based Intrusion Detection Systems (NIDS) that sits on the wire and passively monitor traffic by processing packets and matching them against pre-defined malicious patterns. The problem here lies in the fact that NIDS interacts with the passing packets. Attackers can figure out a way to create a packet that would not hurt the targeted system but the NIDS sitting between the attacker and the target system. Such a packet can make the NIDS blow up and fall off-line. Hence, while using the hypervisor to monitor security within the VMs, hackers may find a way to create files or processes that don’t necessarily harm the VMs themselves but instead attack the introspection system. In this way, a hacker can gain full access to the hypervisor which in turn has access to everything. The result will be a cascade effect leading to VMs themselves becoming compromised.

Another concern is that introspection can potentially break segregation of duties[27], leading to uncontrolled access to VM data. Other similar arguments state that introspection generally creates a backdoor access to every VM.

For public deployment models, introspection is not suitable because any access to tenant data is unlogged. When the hypervisor monitors file processing within each VM, the VM itself does not monitor any activity. When the provider deploys introspection to monitor processes inside a VM, the tenant has no available audit trail of what took place. Two problems arise here:

- Possible uncontrolled access that cannot be audited
- Bulls-eye on the provider's back—if attacking the VM directly to access tenant's data is not possible, a provider (having access to all data running on the VMs) who uses introspection can be attacked

In contrast to public deployment, introspection makes a good architecture for private deployment models. It should be used when focus is placed in protecting the network, when risks to the hypervisor itself can be controlled, and where segregation of duties is not an issue. Moreover, introspection secures the infrastructure; by migrating a VM to another virtualization infrastructure, risk mitigation provided via introspection may be lost. Fear of this happening can result in vendor lock-in.

2.5 Data Misplacement (Resource Allocation)

Data leaks are a risk when physical memory or data storage used by one virtual machine is reallocated to another[28]. Leaks occur when a VM shrinks or is no longer needed and freed resources are allocated to another VM. It is possible for the new VM receiving the additional resources to use forensic investigation techniques to acquire an image of the whole physical memory and data storage. It is not clear whether we can leverage this. It is certainly hard to argue that attackers keep on waiting until we make our partition smaller in order to try and get our data. It is entirely possible; however, that this is opportunistic—attackers may be able to find valuable information, not necessarily because they have a specific target but simply because the data is there to be had.

Hence, when reallocating resources from one VM to another, both must be properly secured. The old data present in the physical memory and in the data storage should be nullified so as to prevent other VMs from pulling data out of them and gaining access to important information still contained there.

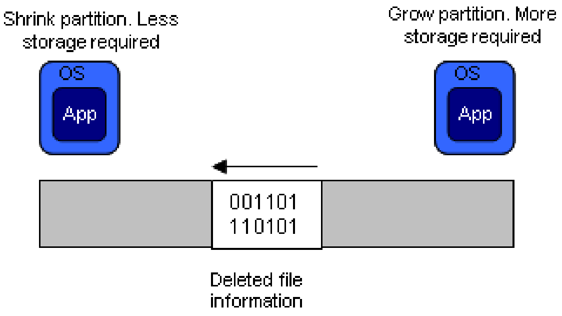
Shrink partition. Less storage required
Grow partition. More storage required
OS
App
OS
App
001101
110101
Deleted file information

Figure 8. Data misplacement.

By design, OpenStack allows zeroing of all the data used by a virtual resource (VM or virtual volume) once the resource is released. This happens via internal KVM procedures without any intervention needed from the CSP's side. However, in order to clean the used memory, CSPs should ensure that they have updated the Cinder configuration (available in /etc/cinder/cinder.conf file) in order to delete blocks that have some written data.

2.6 Configure Access and Security

According to the principle of defence-in-depth[29], layered security mechanisms increase the security of the system as a whole. We should therefore use the additional **OpenStack** hardening mechanism related to the network ports, namely the **security groups**[30]. These are used to define a number of IP firewalling rules, describing the kind of network traffic allowed to go to or come from the VMs[31].

As these rules are project/application specific, CSCs should carefully review which protocol to use and the port range. However, at a minimum, SSH ONLY access should be allowed.

To support this operation, CSCs should import their public keys (**SSH credentials**) into OpenStack that will be **injected into the VM** when it is launched.

Here we should note that even when a VM is compromised, the security group rules still provide the required level of security. This is because they are implemented in the host operating system, i.e., the operating system of the physical machine where the VM is hosted.

2.7 Continuous System Management

Arguably, one of the most important elements of cloud security is configuration management, including patch management[32]. Since the web will always be prone to bugs, it is of essence to be prepared to apply security updates and general software updates using configuration management and deployment tools.

2.7.1 Vulnerability Management

Vulnerability management depends on the cloud delivery model and is a shared activity between the CSP and CSC. However, for IaaS and PaaS models it is up to the CSP to perform vulnerability scanning and penetration testing activities. This should be done on a regular basis in order to evaluate the security posture of systems exposed to the Internet.

These activities should be performed using some kind of automated tools. Security auditing tools help automate the process of verifying that a large number of security controls are satisfied for a given system configuration. Combining configuration management and security auditing tools create a powerful combination—auditing tools will highlight deployment concerns while configuration management tools will simplify the process of changing each system to address the concerns highlighted in the audit. Penetration testing tools have been developed to assist in the automation of the process. These are:

1. Zed Attack Proxy (ZAP)[33]
2. OpenVAS[34]
3. SQL Inject Me[35]
4. HTTP Directory Traversal Scanner[36]
5. Burp Suite[37]
6. Qualys SSL Server Test[38]
7. Tamper Data (Samurai WTF)[39]
8. Vega[40]

The testing tools above can be used to conduct a security assessment on own VMs only. However, one should **always** check the terms of service of the selected CSP to determine whether running security tests on the CSP infrastructure is allowed, even if their own machines are the target. If this is not so, either a CSP that allows penetration tests on own VMs must be chosen, or have tests run on a development or testing environment before deployment to the production environment.

2.7.2 Configuration Management

Configuration management allows avoiding the many pitfalls inherent in building, managing, and maintaining complex infrastructures. We should always use tools to automate configuration and deployment. This eliminates human error and allows the cloud to scale much more rapidly[41].

For IaaS deployments we should lock down VMs as securely as possible. Ultimately, we should manage our own VMs—given the exposure level within the cloud, both the application code and the underlying software stack are our responsibility. We should decommission all unnecessary services and applications, remove any unneeded codes, limit user and group access to the bare minimum, and consistently keep systems patched.

For PaaS deployments, CSPs offer a computing platform that could include an operating system, programming languages, an execution environment, a database, and a web server. The management of the components of a PaaS deployment is left to the CSP, who must be able to meet service level agreements (SLA).

2.7.3 Patch Management

Patch management refers to controlling the implementation of fixes so as to resolve the defects/problems identified[42]. One of the reasons why a patch should be deployed can be due to vulnerability management.

A patch manager should keep track of deployed patches while securing the necessary approvals before every patch is deployed. Although automatic updates for operating systems and applications represent a good approach, they can lead to unexpected behaviour and problems. It is therefore recommended to first test these patches in the development/test environment before applying them to the production environment. In addition, as patch deployment can lead to outages one should deploy them in batches to the possible extent, choose lean usage time periods to run the deployment process, and inform users in advance.

Basic steps governing the upgrades and/or patches process, a necessary part of any IT system, are as follows:

1. Identify the responsible partner to lead the implementation of the requested changes
2. Identify the required changes and outline a “chain of command”, a project plan and a timeline, to test and implement those changes
3. Identify the process for resolving issues introduced by an upgrade, including a clearly defined set of responsibilities and methods for resolving those issues
4. Define a rollback process to restore an upgrade to its initial state should the changes cause unexpected and major failures

2.8 Decommissioning of Unnecessary Services

An important principle in network security is to only run the services that are absolutely necessary, thus reducing the ways an attacker might compromise systems. It's important to periodically review the necessity of the services provided and to completely decommission any unnecessary services.

Moreover, insecure services should be avoided wherever possible as they can be exploited by an attacker. Such services include:

- Telnet (use SSH instead)
- Plain FTP
- Open mail (SMTP) relays

Additionally, periodic port-scanning should be used to check for unnecessary services enabled inadvertently, as well as to ensure that services intended for local use only are not made publicly-available, such as:

- File- and print-sharing services (SAMBA, NFS, CUPS)
- Memcached
- Direct database access
- Universal Plug 'n' Play (UPnP)
- Remote Procedure Calls (RPC)
- Simple Network Management Protocol (SNMP)

Finally, one should maintain a list of which services should be made available, while periodically reviewing the necessity of the services provided and restricting or decommissioning those which are not necessary. In addition, one should record any temporarily installed services which will eventually be disabled/decommissioned. Attention should be paid in ensuring (for example by using port scanners) that the decommissioning procedure has actually succeeded.

2.9 Cryptographic Operations

In an IaaS cloud deployment, the CSC deploys computing resources (VMs) from a shared pool of configurable computing resources. This involves the following operations:

(a) Authentication of the offered pre-built images to ensure that they are from authorised sources and have not been tampered with

(b) Authentication via the management interface of the hypervisor, needed to launch the VM and to perform subsequent lifecycle operations on that VM (Stop, Pause, Restart, Kill, etc.)

(c) Secure interaction with the running VM instances

To assure the integrity of the VM templates the CSP should either[43]:

(a) Digitally sign the templates (with the private key being securely stored at the CSP's premises and protected while in use) using strong (e.g., FIPS 140-2 compliant) algorithms, while the corresponding public key is made available to the CSC in an authenticated manner

(b) Use strong cryptographic algorithms, such as AES, RSA public key cryptography, and SHA-256 or better, computed over the VM code, with the corresponding cryptographic hash made available to the CSC in an authenticated manner

(c) Use a Hash-based Message Authentication Code (HMAC), using a cryptographic algorithm and a secret key that both the CSC and CSP share

(d) Regenerate images on a daily basis

(e) Allow CSCs to upload their own image templates on the cloud image repository

To assure the integrity of hypervisor API calls the CSP should implement functionality whereby the VM Management Interface of the hypervisor only accepts and executes authenticated API calls. This is possible via a private/public key pair generation, used for signing the calls submitted to the VM Management Interface. A third-party trusted authority can be used to sign the public key certificate. The certificate is then made available to the VM Management Interface of the hypervisor to verify the signature of the calls submitted by the consumer to the VM instance[43].

To assure the integrity of communication with running VM instances[43]:

(a) A CSC should upload their public keys and inject them in the activated VMs

(b) A CSP should deploy a Secure Shell (SSH) protocol. This strong cryptographic authentication prevents anonymous connection attempts to the VM instance, as well as authentication attacks. When SSH is used, not only is the administrator authenticated but all commands, responses, and payloads are protected in both directions from eavesdropping and undetected modifications, in addition to being cryptographically authenticated

3. Conclusion

When considering a move to cloud computing, we must have a clear understanding of the associated potential security benefits and risks so as to set realistic expectations on our CSPs. Attention must be paid to the different service models (IaaS, PaaS or SaaS) as each model carries different security requirements and responsibilities.

Cloud computing services certification is an important aspect—it provides assurance that our critical security requirements are being met. We should therefore identify which security certifications are important to us and insist that our CSP demonstrates their conformance.

Our analysis of the security issues associated with virtualization lead us to the following principles:

- The VM operating system firewall software should be leveraged to secure the VM
- Introspection should not be used in public deployment models—host-based security should
- Regularly check for new updates and apply them accordingly

Finally, securing our cloud infrastructure means not only implementing controls for the layers we are able to but also auditing our CSP regarding actions taken to lock-down the tenant instances. We must conduct our own analysis of our needs—assess, select, engage, and oversee the cloud services that can best fulfil those needs.

When we talk about securing our cloud infrastructure, part of it is what we are going to do to implement controls for the layers that we have control over and how we can audit the CSP regarding the actions taken to lock-down the tenant instances.

References

1. Wang C, Wang Q, Ren K, *et al*. 2009, *Privacy-preserving public auditing for data storage security in cloud computing*, Cryptology ePrint Archive, viewed February 1, 2016, <https://eprint.iacr.org/2009/579.pdf>
2. Zhu Y, Wang H, Hu Z, *et al*. 2010, *Efficient provable data possession for hybrid clouds*, Cryptology ePrint Archive, viewed February 1, 2016, <http://eprint.iacr.org/2010/234.pdf>
3. *Apache CloudStack*, viewed January 28, 2016, <http://cloudstack.apache.org/>
4. *Eucalyptus*, viewed January 24, 2016, <https://www.eucalyptus.com>
5. *OpenNebula*, viewed January 25, 2016, <http://opennebula.org/>
6. European Commission, 2013, *What does the Commission mean by secure cloud computing services in Europe?*, viewed January 12, 2016, <http://europa.eu/rapid/press-release_MEMO-13-898_en.htm>
7. Raju R V, Vasanth V and Udaykumar P, 2013, Data integrity using encryption in cloud computing, *Journal of Global Research in Computer Science*, vol.4(5): 40–43, viewed January 14, 2016, <http://www.rroij.com/open-access/data-integrity-using-encryption-in-cloud-computing-40-43.pdf>
8. Ali M, 2014, *What is cloud computing stack (SaaS, PaaS, IaaS)*, Maizkglobal Global IT Solutions, viewed January 18, 2016, <http://www.mazikglobal.com/blog/cloud-computing-stack-saas-paas-iaas/>
9. Brenton C, 2012, *Delineation of cloud responsibility*, SANS Information Security Training, viewed October 17, 2015, <https://www.sans.org/cloud/2012/07/02/delineation-of-cloud-responsibility>
10. Lukan D, 2014, *Addressing the most critical cloud security threats*, TechTarget SearchCloudSecurity, viewed January 21, 2016, <http://searchcloudsecurity.techtarget.com/tip/Addressing-the-most-critical-cloud-security-threats>
11. *Choosing a Cloud Provider with Confidence*, n.d., viewed January 4, 2015, <https://www.geotrust.com/resources/whitepapers/choosing-cloud-provider.pdf>
12. Cloud Standards Customer Council, 2015, *Security for cloud computing – 10 steps to ensure success, version 2.0*, viewed May 14, 2015, <http://www.cloud-council.org/deliverables/CSCC-Security-for-Cloud-Computing-10-Steps-to-Ensure-Success.pdf>
13. Rouse M, 2011, *Definition of cloud services*, TechTarget SearchCloudProvider, viewed January 29, 2016, <http://searchcloudprovider.techtarget.com/definition/cloud-services>
14. Cloud Security Alliance, n.d., *Virtualization Working Group*, viewed January 30, 2016, <https://cloudsecurityalliance.org/group/virtualization>
15. Adams D, 2010, *Top 7 threats to cloud computing — part 1*, Patriot Technologies Inc., viewed December 3, 2015, <http://patriot-tech.com/top-7-threats-to-cloud-computing-part-1>
16. Linthicum D, 2015, *Minimize threats through public cloud security testing*, TechTarget SearchCloudComputing, viewed January 20, 2016, <http://searchcloudcomputing.techtarget.com/tip/Minimize-threats-through-public-cloud-security-testing>
17. *FedRAMP*, n.d., viewed September 21, 2015, <https://www.fedramp.gov/>
18. U.K. Government Digital Service, 2013, *The G-cloud framework on the Digital Marketplace*, viewed September 21, 2015, <https://www.digitalmarketplace.service.gov.uk/g-cloud/framework>

19. *Cloud Security Alliance*, n.d., viewed September 13, 2015, <https://cloudsecurityalliance.org/about/>
20. Horrigan B L, 2015, *Securing your cloud deployment*, Information Security – Insider Edition, viewed September 30, 2015, <http://docs.media.bitpipe.com/io_12x/io_121990/item_1101004/ISM_InsideEdition_Securing%20Your%20Cloud%20Deployment_final.pdf>
21. Cloud Security Alliance, n.d., *STAR Certification*, viewed October 12, 2015, <https://cloudsecurityalliance.org/star/certification/>
22. Cloud Security Alliance, *STAR Attestation*, viewed October 12, 2015, <https://cloudsecurityalliance.org/star/attestation/>
23. SSAE-16, n.d., *SSAE SOC 2 report — Trust Services Principles*, viewed November 12, 2015, <http://www.ssae-16.com/soc-2/>
24. Cloud Security Alliance, 2015, *Consensus Assessments Initiative Questionnaire v3.0.1 Info Sheet*, viewed November 15, 2015, <https://cloudsecurityalliance.org/download/consensus-assessments-initiative-questionnaire-v3-0-1/>
25. OpenStack, n.d., *Heat*, viewed May 12, 2015, <https://wiki.openstack.org/wiki/Heat>
26. Brenton C, 2012, *Virtual firewall appliances — trust misplaced?*, Cloud Security Alliance, viewed December 1, 2015, <https://cloudsecurityalliance.org/wp-content/uploads/2012/02/VirtualFirewallAppliances-TrustMisplaced.pdf>
27. Brenton C, 2011, *Hypervisor versus host based security*, Cloud Security Alliance, viewed November 16, 2015, <https://cloudsecurityalliance.org/wp-content/uploads/2011/11/hypervisor-vs-hostbased-security.pdf>
28. Lukan D, 2014, *How to limit security risks during cloud computing virtualization*, TechTarget SearchCloudSecurity, viewed December 2, 2015, <http://searchcloudsecurity.techtarget.com/tip/How-to-limit-security-risks-during-cloud-computing-virtualization>
29. OWASP Open Software Security Community, 2015, *Defense in depth*, viewed November 10, 2015, <https://www.owasp.org/index.php/Defense_in_depth>
30. OpenStack, n.d., *Security Groups*, viewed August 10, 2015, <http://docs.openstack.org/openstack-ops/content/security_groups.html>
31. OpenStack, n.d., *Configure access and security for instances*, viewed May 12, 2015, <http://docs.openstack.org/user-guide/cli_nova_configure_access_security_for_instances.html>
32. Shackleford D, 2014, *Assessing cloud security controls key in repelling cloud attacks*, TechTarget SearchCloudSecurity, viewed November 20, 2015, <http://searchcloudsecurity.techtarget.com/tip/Assessing-cloud-security-controls-key-in-repelling-cloud-attacks>
33. OWASP Open Software Security Community, n.d., *Zed Attack Proxy Project*, viewed March 8, 2016, <https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project>
34. *OpenVAS*, n.d., viewed March 8, 2015, <http://www.openvas.org/download.html>
35. *SQL Inject Me 0.4.5 end-user license agreement*, n.d., viewed March 8, 2015, <https://addons.mozilla.org/en-US/firefox/addon/sql-inject-me/eula/88410?src=collection&collection_id=203cc10a-26b3-5921-12ef-6ba80b06fe07>
36. AutoSec Tools, 2016, *HTTP directory traversal scanner*, viewed March 10, 2016, <http://www.autosectools.com/Page/HTTP-Directory-Traversal-Scanner>
37. PostWigger Web Security, n.d., *Burp Suite*, viewed March 8, 2016, <https://portswigger.net/burp/>
38. Qualys SSL Labs, n.d., *SSL server test*, viewed March 8, 2015, <https://www.ssllabs.com/ssltest/index.html>
39. *Samurai Web Testing Framework*, n.d., viewed March 8, 2015, <https://addons.mozilla.org/el/firefox/collections/rsiles/samurai/>
40. Subgraph, n.d., *Vega vulnerability scanner*, viewed March 8, 2015, <https://subgraph.com/vega/>
41. OpenStack, n.d., *Openstack Security Guide*, viewed February 5, 2016, <http://docs.openstack.org/sec/>
42. Sidana H S, 2012, Cloud computing: managing security. *Infosys Labs Briefings*, vol.10(1).
43. Chandramouli R, Iorga M and Chokhani S, 2013, *cryptographic key management issues & challenges in cloud services*, National Institute of Standards and Technology, viewed January 27, 2016, <http://dx.doi.org/10.6028/NIST.IR.7956>