Royal HOLLOWAY SERIES 2012

The Threat of Coexisting With an Unknown Tenant in a Public Cloud

An examination of the vulnerabilities of the cloud, with a focus on the issues of attackers' ability to load malicious programs on to the same virtual machine your organisation is using.

BY JACOBO ROS AND CHEZ CIECHANOWICZ

TechTarget

Royal Holloway University of London

The Threat of Coexisting With an Unknown Tenant in a Public Cloud

IF WE WANT to protect our network, we add firewalls and other network security devices. So far this has worked reasonably well, even though networks and systems get more and more complex every day. But the rise of the cloud has broken the very foundation of this idea. Now an attacker does not need to penetrate several perimeters to reach sensitive data. Instead, he can rent a virtual machine in a cloud service, and from there compromise different targets, thereby avoiding many of the usual security procedures.

VIRTUALISATION AND MULTI-TENANCY

To understand why and how this happens it is necessary to go back to the properties of the cloud that allow this situation: virtualisation and multi-tenancy. Virtualisation enables the resources of a physical machine to be divided or shared through multiple environments which may or may not be aware of the others. These environments are known as *virtual machines* (VMs).

Though multi-tenancy is not defined as an essential characteristic for cloud computing, it is indeed an important part. The main idea behind multi-tenancy is several tenants (or users) coexisting in the same infrastructure. Like the tenants of a building, each one has its own flat but they all live in the same structure.

In *Infrastructure-as-a-Service* (IaaS) cloud environments, multi-tenancy makes use of virtualisation technologies to increase resource utilisation, load balancing, scalability, and reliability. This allows the cloud service providers to maximise use of their infrastructure by multiplexing their physi-

cal machines with virtualisation and then assigning the VMs to different clients when required. This could lead to different users coexisting in the same environment. So, in a nutshell, an attacker can rent one of these VMs and instantly be shoulder to shoulder with several potential targets.

With several users sharing the same physical machine, traditional network security controls become almost useless. The communications between the different VMs in the same machine now go through virtual networks provided by "hypervisors" whose security controls are still relatively immature. Furthermore, in order to reduce its complexity, the hypervisor does not have sufficient capability to monitor the communications.

But there is a bigger problem than the fact that communications between VMs cannot be monitored and analysed. Cloud service providers create a dangerous lack of trust between the different VMs by allowing them to coexist without fully comprehending the risks that this can pose. Tenants indeed have a false sense of isolation and security, not noticing other tenants who could possibly have harmful intentions against them. Of course, the most insecure state is when you totally believe yourself to be completely secure.

The entire situation above gives an attacker a privileged position to perform malicious activities in the cloud, especially in IaaS clouds. That does still not mean the attacker can gain any real profit from it since the coexistent VMs may not be attractive targets. But what if the attacker could select a target and somehow manage to locate its malicious VM in the same physical machine? Of course it is not an easy task, and the concept of the cloud should not allow it, but it can be possible.

ATTACKING THE CLOUD

There are just a few a VM-to-VM documented attacks but they generally consist of three main steps:

1. **Locate** the target VM and place a malicious VM next to it.
2. **Gather** information about the target VM.
3. **Compromise** the target VM.

Locate and place. Mapping the cloud–*cloud cartography*–mainly depends on the characteristics of the cloud service provider, and so there is no general process to do it. There is not yet much research about this, but some is focused on Amazon's EC2 and contains claims that it might also work on other IaaS clouds which provide similar functionality.

The process used to map the cloud consists of two stages. First it is necessary to enumerate the public services using external probes (for example, *nmap*, *hping*, and *wget*). Cloud service providers usually provide an internal DNS service to map public to private IP addresses. So, the second step will be to map the IP addresses of the public services that responded to the first step, to their internal IP addresses.

In Amazon's EC2 scenario, two VMs that have the same creation parameters (e.g. region and instance type) have a high chance of being co-residents; and the creation parameters are also related to the assigned internal IP addresses. So the attacker will first search for the internal IP address using the two steps noted above. Then, because there is a relation between the internal IP address and the VMs' creation parameters, the attacker can narrow down the search for the right parameters, thereby drastically reducing the number of instances needed before a co-resident placement is achieved.

To determine the location of a target, an attacker could use network-based checks, such as performing a route trace to the target and checking the number of hops, or using side-channel vulnerabilities to analyse possible co-residence.

The only step left in this phase is to create a VM on the desired machine. The simplest way to do this is by trial and error: creating several VMs during a period of time, using the appropriate parameters, and checking for co-residence. If the VM instance is not created on the target VM's physical machine, then the VM is terminated and another one is launched. This takes advantage of the placement algorithms that tend to launch new VMs with similar parameters on the same physical machine. Because the idea of cloud computing is to maximise resources, if a VM is not being used, it is suspended, and re-launched when needed. So an attacker monitors for the activity of the target VM until it is instanced and then launches a VM fast enough to be assigned the same physical environment.

Gathering and compromising. Once the malicious VM has been placed near the target, information about the target is gathered through different side-channels such as measuring the cache usage, or estimating the traffic rate. This could provide valuable information in order to accomplish the last phase of the attack, namely compromising the target, but it is not always the case. If, for example, several VMs are placed in the same physical machine then measuring cache usage may not reveal enough about the target VM since other tenants are accessing the cache too.

Finally, the target VM can be compromised using either of two different

vectors. The attacker might decide to attack the VM directly through the compromise of side-channels like memory or virtual networks. This kind of attack is known as a *VM-to-VM attack*, and the goal is to break the isolation of the VM.

On the other hand, instead trying to attack a VM directly, an attacker could compromise the hypervisor, thereby gaining access to all the resources and obtaining full control over the physical machine. In theory, securing the hypervisor against takeover should be feasible since it has much less code than an operating system, but the truth is that the requirements of cloud services force the hypervisor to provide more functionality than ideally it should, thereby increasing its complexity and the chances of it being compromised.

MAKING IT SAFER

The most effective way to protect against a threat is to prevent it; so, even though it is important to harden all the elements involved in cloud computing to achieve total isolation between VMs, special attention should be taken to ensure first of all that an attacker cannot select where to place a malicious VM. Of course, the risk of random VMs being compromised will still be present and that is why all the elements must still be hardened, but the attacker would no longer be able to select a specific target.

This would mean that if an attacker starts a VM, the chances are high that he will barely know anything about the other VMs coexisting on the same physical machine. Those VMs may have nothing of interest to the attacker. This situation will leave the attacker blindly searching throughout the cloud service. Even if there is an interesting target, it will be complicated for the attacker to recognise its importance. Lack of knowledge about that potential target will further reduce the probability of the VM being compromised.

To ensure the safety of their clients, and to prepare a good defence strategy plan, the cloud service provider must develop a framework to map the network topology of its cloud. Furthermore, randomly assigning internal IP addresses and restricting the use of the internal DNS would prevent a malicious VM from scanning the inside of the cloud. Monitoring the creation of VMs is also a good practice to prevent brute-force attacks.

FINAL THOUGHTS

A shift to the cloud is inevitable, and for many it has to be done. But, as with everything nowadays related to technology, security must be kept

in mind at all times. Cloud computing adds new vectors for an attacker, and because of its immaturity, every aspect of the cloud must be carefully investigated for weakness.

Virtualisation multiplies the available resources, but needs to properly isolate each VM. Multi-tenancy maximises the use of resources, but VMs with different security needs must be separated to minimise their exposure.

The ease of an attacker to target a specific VM is what might scare users most, so those cloud service providers that want to highlight security as their insignia should start by hardening their methods for assigning VMs in the cloud. ■

ABOUT THE AUTHORS:

Jacobo Ros successfully pursued his MSc Information Security at Royal Holloway, completing it in 2011. Currently he works as a security consultant for the London-based security company Context Information Security. He is mainly involved in security assessment activities, performing web application and infrastructure penetration tests.

Chez Ciechanowicz is a course director and supervisor at Royal Holloway's information security group. His special interests are in risk analysis and security management.