Kaspersky
Hybrid Cloud
Security

Security of public cloud instances is your problem. So deal with it.

2019

kaspersky

Learn more on kaspersky.com/hybrid

Security of public cloud instances is your problem. So deal with it.

Intro

Public cloud usage is growing because it provides so many benefits, including instant scalability, automation, configurability and flexibility. All those engaged in managing public cloud deployment, both Information Security teams and those outside this discipline (such as Dev Ops or Web Dev) need to ensure that the security of corporate assets is seen as a planning cornerstone.

This White Paper is designed to arm InfoSec Specialists with the information and evidence needed to ensure that Cloud Workload Security, i.e. the security of the Operating System that the workload is running, is treated as a fundamental consideration in cloud deployment scenarios. It is, after all, your problem – if your security is breached in the cloud, whoever is technically to blame, you're always going to have to handle the consequences.

We aim to counteract the erroneous view that things like software vulnerability exploitation (login bypass, remote code execution, etc), update repo poisoning, network connection exploitation (e.g. DNS hijacking), and account information compromise only happen in physical or virtualized environments, and not in public clouds. Or that, in public cloud environments, damage done to your data or organization through a security incident somehow ceases to be your problem.

In addition, this document draws attention to some specific business risks posed by public clouds – including cloud resource theft. Instant scalability in the cloud means that cybercriminals taking over your cloud controls can spin up and utilize almost infinite volumes of computing power in your name, and at your expense. One way and another, fully securing your public cloud infrastructure makes for a wise investment choice.

Vulnerabilities in public clouds

In addition to most feared (and rightly so) security issues faced in public clouds – account compromise and misconfiguration – there are instance-targeting threat vectors that utilize vulnerabilities in services exposed to the internet – services like RDP and SSH.

RDP is on by default on Amazon instances, and 2 factor authentication is not supported by design. RDP has become the target for many different types of attack. Some attacks use only the most popular logins while bruteforcing the password, while others bruteforce logins using the most common passwords. Some attackers limit and randomize the number of login attempts, with a time-out between sets of attempts, to avoid automated detection. Another method of attack is to bruteforce the password for the service SSM-User login often pre-installed onto AWS instances.

Similar bruteforce attempts target SSH services all the time, and while SSH does offer greater protection than RDP (e.g. 2 factor authentication), a carelessly configured service can readily provide access to a persistent malicious actor. SSH and RDP bruteforce attacks together made up 12% of all attacks on Kaspersky's IoT 'honeypots' during the first half of 2019[1]

Public clouds can and do expose you to vulnerabilities – here are a few examples of how a vulnerability in third party software offers an attacker the chance to execute code on the instance itself.

On June 3rd 2019, a vulnerability was discovered in Exim – a popular email server that's often deployed in public clouds. This vulnerability allowed for remote code execution. If the server was run under root, as is most commonly the case, the malignant code introduced onto the server would then be executed with root privileges. Yet another Exim vulnerability was identified in July of 2019, also allowing remote code execution as root[2].

Another example is the 2016 hack of the official Linux Mint website. This resulted in the distributives being altered to include an IRC-backdoor with DDOS functionality. The Trojan could also be used to drop malicious payloads onto infected machines.

There have been cases of malicious node.js modules, infected containers in the Docker Hub[3] and many more. Cybercriminals are very inventive when it comes to finding entry points into infrastructures, especially where there are many such infrastructures, all very similar and with similar issues, and all conveniently believed to be highly secure by design, and so not requiring any additional protection.

Over half of all workloads in public clouds run under Linux, and there's an unfortunate myth that such workloads are impenetrable to an attacker.

But vulnerabilities, compromised modules and malicious script certainly exist in Linux environments. Here are our anti-malware lab's Linux threat stats:

Threat	**%** of affected users
Exploit	**41%**
Backdoor	**24%**
Trojan	**14%**
Other	**21%**

1 https://securelist.com/it-threat-evolution-q1-2019-statistics/90916/

2 https://www.bleepingcomputer.com/news/security/critical-exim-tls-flaw-lets-attackers-remotely-execute-commands-as-root/

3 https://www.helpnetsecurity.com/2019/04/29/docker-hub-breach/

Security breaches and attacks in public clouds

What can a malevolent actor can do once inside your cloud infrastructure? In addition to getting access to your corporate resources, such as customer data, they can basically leverage the same public cloud benefits as you – instant scalability, automation and configurability. And all at your expense.

Case 0 – We've got a situation

Even if a cybercriminal does nothing at all after gaining access to one of your systems, you still have a problem – you've been breached. In many jurisdictions worldwide, if a system working with sensitive, protected or confidential data is accessed, you're required by law to notify the authorities of the breach, as well as remediating any damage caused.

The indirect costs of data breach remediation can often be higher than the direct costs, while the effects are usually much longer lasting. And if the compromised machine is used as a stepping stone, i.e. a base for lateral movement, surveillance, privileged account information location and extraction or data gathering and extraction - the consequences for you could be dire.

Think about it:
paying for covert mining is essentially transferring money to a cybercriminal's account. Don't fund their business activities at the cost of your own.

Case 1 – SSH/RDP services-based attack

If attackers have SSH access to an infected device, they have far greater scope to monetize the infection. In the overwhelming majority of cases involving intercepted sessions that Kaspersky's investigated, we've found spam mailings, attempts to use our honeytrap as a proxy server, and (last but by no means least) cryptocurrency mining[4]. We actually registered over 50,000 attempts to infect Windows Server environments with miners during the first half of 2019.

So how do you prevent these attacks?

Solution

Effective public cloud security should be able to address cases like this on multiple fronts:

- Application control in default-deny mode would automatically deny permission for the attacker's software to deploy or activate
- Runtime protection would prevent any resource-hijacking software from launching
- Behavior protection would spot mining, spam-generating or any other ill-intentioned software, based on its behavior

Case 2 – Losing a great deal of money in a very short time

Instant scalability and automation means that cybercriminals not concerned about staying below the radar can run wild with your budget. An AWS instance-hour costs anywhere between 5 cents and 5 dollars – that's is up to $125 a day per instance. And there's no limit to the number of instances your attacker can spin up in your name. They can use cloud formation templates to automate the spawning of new cloud compute in order to carry out a task – like cryptocurrency mining or DDOS. The attacker just needs to create new instances a little faster than you can decommission them. It's not impossible to lose $14k in a day[5] or even $50k or $60k[6]. Or you could find yourself apparently launching a DDOS attack, or becoming subject to one[7]...

Solution

An effective security solution could help on multiple fronts:

- Visibility – An alert engine and appropriate reporting systems would immediately highlight the instance sprawl to the administrator
- Internet access control – a policy could be deployed to prevent new instances and communications with C&C servers, as well as preventing outbound network attacks

Case 3 – Supply chain attack

Environments built from many sources offer many possible options when it comes to attacking the software supply chain, including Linux repository poisoning. An example – MeDoc, a Ukrainian business, suffered an attack on its update mechanisms[8]. The compromised updates were used to launch an attack based on ExPetr, the ransomware wiper targeting Windows platforms, which went on to affect many organizations, including pharma giant Merck, shipping firm Maersk and Ukrainian critical infrastructures.

4 https://securelist.com/it-threat-evolution-q1-2019-statistics/90916/

5 https://dev.to/juanmanuelramallo/i-was-billed-for-14k-usd-on-amazon-web-services-17fn

6 https://www.quora.com/My-AWS-account-was-hacked-and-I-have-a-50-000-bill-how-can-I-reduce-the-amount-I-need-to-pay

7 https://www.reddit.com/r/aws/comments/3qt4e0/so_i_was_ddosed_by_35924_amazon_aws_ip_addresses

8 https://securelist.com/schroedingers-petya/78870/

https://securelist.com/in-expetrpetyas-shadow-fakecry-ransomware-wave-hits-ukraine/78973/

https://www.npr.org/sections/thetwo-way/2017/06/27/534560169/large-cyberattack-hits-ukraine-snarling-electric-grids-and-airports

Solution

Even if malware arrives as a part of update to the protected system, it should be dealt with exactly like any other executable or script – analyzed before it executes and monitored when running. Application control in default-deny mode can add a further layer of security.

Case 4 – From DevOps to Secure DevOps

Malware and vulnerabilities embedded into container images can result in a corporate IP leak or production line sabotage. The MeDoc attack above is one example – in another, hackers are believed to have compromised the build environment of CCleaner in order to insert malware[9] into the security product's official distributives.

Solution

DevOp teams moving development into clouds, and employing dynamic environments and tools such as containers, need to recognize the risk and manage it by securing the cloud-based development environment.

Our own cloud-specific security solution, Kaspersky Hybrid Cloud Security, provides runtime protection for Docker hosts to ensure safe development practices. API and other tools are also available for development automation and integration into the CI/CD pipeline.

In the first half of 2019, we fended off over 250,000 attacks on our users' Windows Server platforms. That's not counting AdWare and RiskWare.

Why is it your problem?

Securing a public cloud instance is your responsibility, as any public cloud provider will tell you[10,11]. That's a fact.

Your Public Cloud Provider is also unlikely to be liable for the consequences of any security breach in terms of costs or damage to your brand or public image.

And on top of that – there's no limit to the volumes of computing power an attacker could procure on your behalf without your knowledge. Again, you would be responsible for meeting the costs.

Do you really need an AV in a public cloud?

AWS thinks you do. Their advice is:

"Create a baseline server configuration incorporating up-to-date security patches and host-based protection suites that include anti-virus, anti-malware, intrusion detection/prevention, and file integrity monitoring."

"Each EC2 instance should adhere to organizational security standards. Do not install any Windows roles and features that are not required, and do install software to protect against malicious code (antivirus, antimalware, exploit mitigation), monitor host-integrity, and perform intrusion detection. Configure security software to monitor and maintain OS security settings, protect the integrity of critical OS files, and alert on deviations from the security baseline."[12]

We too believe that you can reduce and manage the risk much more effectively if you protect operating systems on your instances and virtual machines. Basic anti-virus and anti-malware protection is clearly not enough. Industry best practices dictate that every operating system on an infrastructure needs comprehensive, multi-layered protection, and public cloud providers make similar recommendations.

This is where a security solution like Kaspersky Hybrid Cloud Security comes in. Our solution protects different types of workloads running on different platforms, using multiple layers of security technologies, including system hardening, exploit prevention, file integrity monitoring, a network attack blocker, static and behavioral anti-malware, and more.

Ensuring the application of appropriate levels of security to your public cloud environment at all times is essential to avoiding attacks that could prove extremely costly as well as damaging. It's important that security is recognized a fundamental to your ongoing cloud strategy. After all – in the last resort, it's your problem!

9 https://www.wired.com/story/inside-the-unnerving-supply-chain-attack-that-corrupted-ccleaner/

10 https://aws.amazon.com/compliance/shared-responsibility-model/

11 https://docs.microsoft.com/en-us/azure/security/fundamentals/infrastructure

12 https://aws.amazon.com/answers/security/aws-securing-windows-instances/

www.kaspersky.com

Kaspersky Hybrid Cloud Security for AWS: kaspersky.com/aws
Kaspersky Hybrid Cloud Security for Azure: kaspersky.com/azure
Kaspersky Hybrid Cloud Security: kaspersky.com/hybrid

#hybrid
#aws_instance_security
#azure_vm_security