SANS CLOUD SECURITY

Secure Coding.
Practical steps to defend your web apps.

Top 5 Considerations for Multicloud Security

Author: Brandon Evans

15 April 2020

Abstract

The move to leveraging multiple public cloud providers introduces new challenges and opportunities for security and compliance professionals. As the service offering landscape is constantly evolving, it is far too easy to prescribe security solutions that are not accurate in all cases. This paper will examine five critical considerations for securely using the three biggest public cloud providers: Amazon Web Services, Microsoft Azure, and the Google Cloud Platform. While it is tempting to dismiss the multicloud movement or block it at the enterprise level, this will only make the problem harder to control. By embracing multicloud as inevitable and working to understand it, security and compliance professionals can help move the organization forward safely

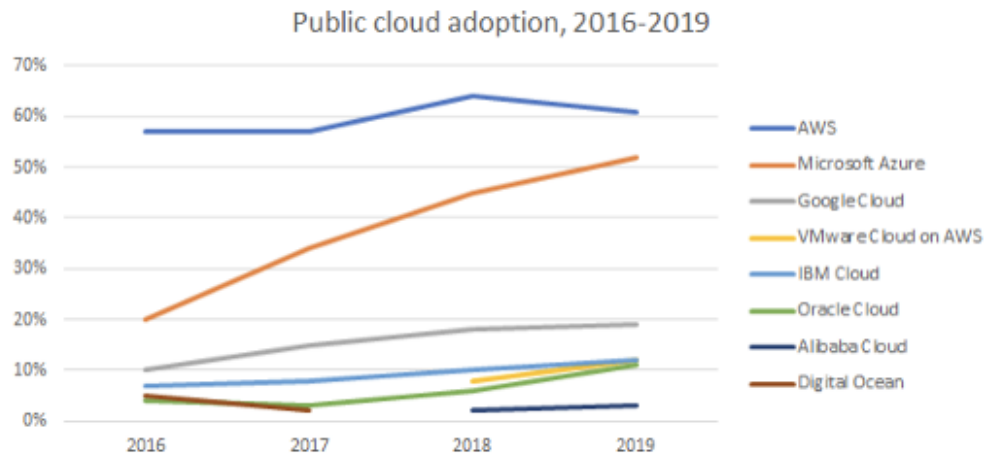
1. Introduction

Despite protest from many security and compliance professionals, public cloud platforms have become an increasingly significant part of nearly every company's technical strategy. This is largely because technical stakeholders have determined that the risks of having a third party manage their data, even if they are significant, are outweighed by the risk of managing the same data internally, or worse, the risk of falling behind the competition due to an inability to quickly innovate.

To make this move responsibly, these companies have carefully vetted and selected the cloud provider that best meets their needs. Historically, the vast majority of companies have selected Amazon Web Services (AWS). Although AWS is certainly still the Infrastructure-as-a-Service leader, it has gained some serious competition. Microsoft Azure has won over many an enterprise with a mature offering with a focus on seamless service integration and security, culminating in them landing the U.S. Department of

Defense's $10 billion JEDI contract in 2019. Despite being in a distant 3rd place, the Google Cloud Platform (GCP) has many fans thanks to Google's development and support of popular open-source technologies, such as Kubernetes, as well as their unique services, several of which came out of their acquisition of Firebase.

Faced with the hard decision of choosing between several viable providers, many organizations opt to take a "multicloud" approach. Instead of going all-in on one provider, multicloud organizations can evaluate analogous offerings and select whichever one serves them best, regardless of who provides it. Nowadays, multicloud is the norm. This allows organizations to save money, avoid vendor lock-in, and to give developers the tools they want to use to deliver cloud-based applications.

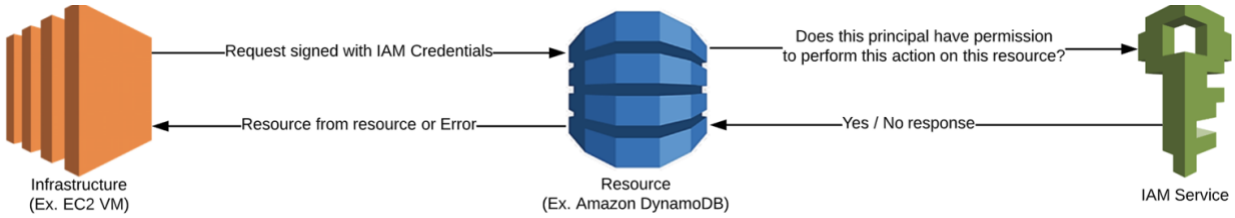


As Azure's adoption has exploded, AWS has largely continued to serve the same percentage of organizations, while Google has increased in use. This would be impossible without the trend towards using multiple cloud providers. https://www.zdnet.com/article/multicloud-everything-you-need-to-know-about-the-biggest-trend-in-cloud-computing

Unfortunately, multicloud means more work for security. It was hard enough to safely move to the cloud with AWS's 175 "fully featured services", but now, security must review and provide guidance for other providers as well. Worse yet, there is a tremendous amount of nuance between the different cloud providers, so it is not enough to learn general cloud concepts; a security team for a multicloud organization must be ready to dive deep into the details. If you find yourself in this challenging position, here are some of the most important security topics to consider.

2. IAM is Hard, and Everywhere

Identity and Access Management (IAM) is one of the most fundamental concepts in the cloud. Proper IAM enables individuals or cloud infrastructure to access the cloud resources they need while keeping everyone else out.



An example workflow of how cloud infrastructure can gain access to other services via IAM.

This ideal is rarely achieved in practice. In a 2017 scan of a sample of over 31,000 EC2 virtual machine instances and workloads, Alert Logic found 30,000 instances of IAM misconfiguration. Attackers are becoming more aware of these weaknesses and are using novel attacks to take advantage of them.

Many developers fail to understand the risks posed by improper IAM policies. From their perspective, they control all of the code that will be accessing their cloud resources. As such, they can merely avoid deploying evil code. This glosses over the very real risk that SSRF, command injection, or supply-chain attacks can enable an attacker to hijack the underlying IAM credentials and write their own evil code that compromises or corrupts your resources. For a live demonstration of the immense damage that can be dealt to an organization with lax policies in a serverless context, check out my SANS@Mic Webcast "Attacking Serverless Servers: Reverse Engineering the AWS, Azure, and GCP Function Runtimes" (spoilers: it is very scary).

Authorizing one cloud to access another further muddies the waters. When operating within a single provider, an organization has the luxury of letting the provider handle the generation, expiration, and automatic rotation of the authentication credentials. By contrast, in order to connect two cloud services in two separate clouds, the organization must export long-living credentials for one provider to the other,

maintaining their own processes for credential rotation. For example, if you wanted to authorize an AWS EC2 instance to access DynamoDB like in the above diagram, you would assign the instance an "instance profile", which would enable the instance to assume an IAM role and obtain temporary credentials to the database. To give Azure access to the same DynamoDB, you would need to create an IAM User, which has permanent credentials, and store those credentials in Azure. AWS considers using IAM roles over users where possible to be best practice, but this kind of multicloud architecture thwarts this ideal.

It does not help that each cloud provider handles IAM differently. Each has its own set of jargon, evaluation logic, and defaults. In particular, GCP breaks IAM conventions. By default, when you launch a virtual machine on AWS or Azure, it does not have access to any other cloud resources. GCP takes nearly the opposite approach: it assigns the VM a default service account with the "Editor" role. Editors have the ability to read and write all in-scope resources as well as create and delete many resources. It is the second most powerful primitive role, right behind "Owner", which can do everything that an editor can do plus manage roles and billing. The same concepts that apply to cloud VMs apply to serverless functions with these providers. Ultimately, it is critical to be aware of these different approaches in order to avoid unnecessarily exposing data and infrastructure to third parties.

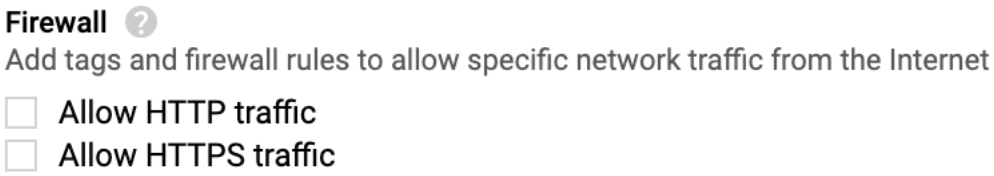
3. Network Security is Still Important

If an organization was capable of shipping software without ever introducing a security bug, maintaining infrastructure without any misconfigurations, and perfectly vetting its vendors, it would not be too dangerous for them to expose all of their assets to the public internet. Alas, we live in reality, and securing our network at the perimeter is our first line of defense against a myriad of threats.

This principle does not change when moving to the cloud. Although we have entrusted our private assets to our provider of choice, we certainly want to keep them from the strangers on the internet perusing Shodan. Each provider tries to make it easy for you to logically isolate your resources from the outside world. To have resources from multiple clouds interact with one another, you have to poke holes into these configurations. This is such an arduous topic that GCP has put out a three-part series on Hybrid

and Multi-Cloud. This complexity is a necessary element of using multiple clouds, even if temporarily incurred while transitioning from on-premises to the cloud or from one provider to another. As the series laments, “a hybrid or multi-cloud setup is rarely a goal in itself, but rather a means of meeting business requirements.”

Just like with IAM, network security is handled drastically differently across the different cloud providers, and again, GCP is the outlier. For example, when launching a virtual machine in AWS and Azure, the providers explicitly ask you to configure which ports to allow inbound traffic and from which sources. Although their automatically recommended configuration allows inbound SSH traffic from anywhere in the world, it is the customer’s part of the Shared Responsibility Model to avoid inappropriately using that setting. Meanwhile, the Google Compute Engine just presents you with this:



A seemingly reasonable set of firewall rules you can allow for the Google Compute Engine.

However, after launching the VM, you will find that you can SSH to it. This is because, for your convenience, GCP has pre-populated the default network with some fairly eyebrow-raising rules, allowing SSH and RDP traffic from anywhere. They even allow inbound ICMP; I know it is common to ping Google in order to troubleshoot a network, but not like this!

Name	Type	Description	Targets	Filters	Protocols / ports	Action	Priority
default-allow-icmp	Ingress	Allow ICMP from anywhere	Apply to all	IP ranges: 0.0.0.0/0	icmp	Allow	65534
default-allow-internal	Ingress	Allow internal traf fic on the default network	Apply to all	IP ranges: 10.128.0.0/9	tcp:0-65535 udp:0-65535 icmp	Allow	65534
default-allow-rdp	Ingress	Allow RDP from a nywhere	Apply to all	IP ranges: 0.0.0.0/0	tcp:3389	Allow	65534
default-allow-ssh	Ingress	Allow SSH from a nywhere	Apply to all	IP ranges: 0.0.0.0/0	tcp:22	Allow	65534

The Google Cloud Platform's highly permissive default network rules.
https://cloud.google.com/vpc/docs/firewalls#more_rules_default_vpc

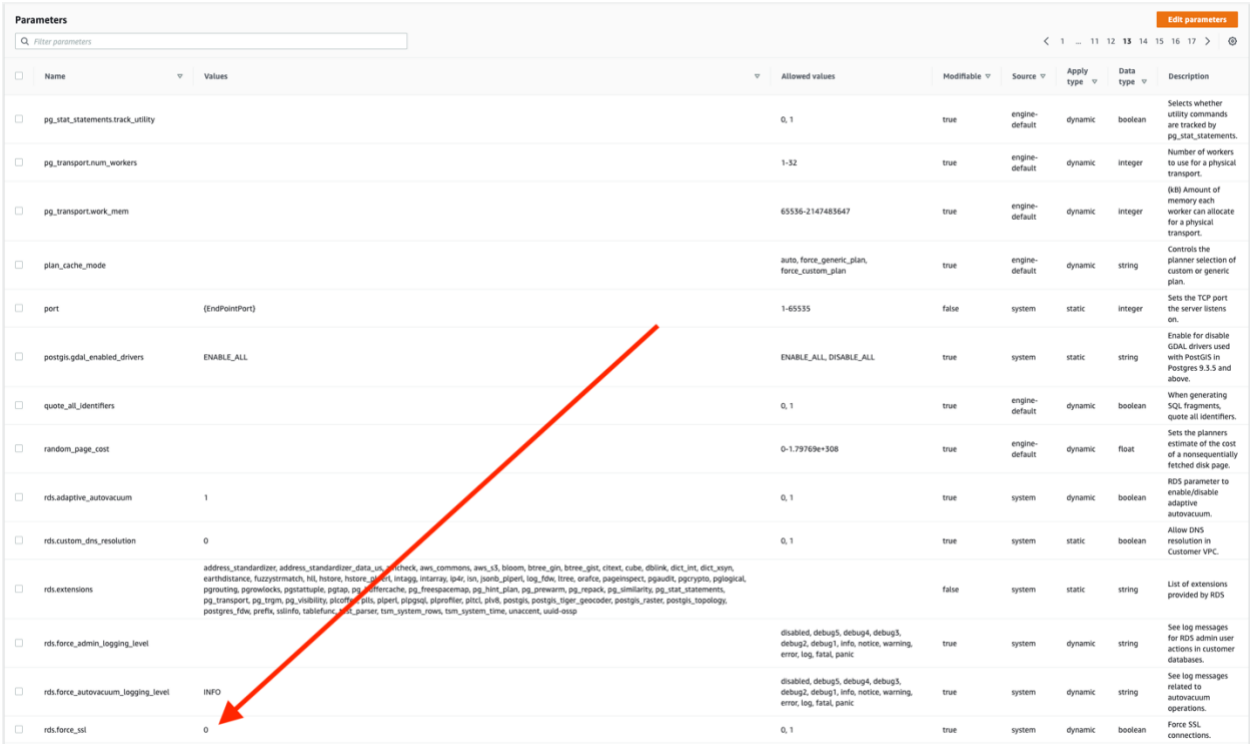
Needless to say, when adopting new cloud providers, it is important to recognize these gotchas, lest you find yourself on Shodan's Search Engine of Shame.

4. Encryption Inconsistencies can Lead to Compliance Headaches

Every cloud provider and security organization agrees that encryption is a critical thing to get right. The only problem is that each might disagree on the details. This is a very broad topic, but we will focus on two of the major categories: encryption at rest and encryption in-transit.

The former subject refers to encrypting data where it is stored. For example, you can configure your managed databases in the cloud to have their underlying disk encrypted. This way, if the machine hosting the database was stolen, the data would be inaccessible to the thief. Azure and GCP have the advantage of always encrypting their database volumes, but this comes at the expense of customizability. Customers cannot turn off encryption, but they also cannot control how the encryption is performed or with what cryptographic keys. This can be a serious compliance concern if your organization has contractual agreements that prevent the use of key material generated in the cloud. By contrast, AWS allows you to encrypt your database at rest with any Key Management Service (KMS) Customer Master Key (CMK) of your choice, be it managed by AWS or by the customer. These are both valid philosophies, but it is important to know which are compatible with your organization.

Encryption in-transit means securing a communication channel such that no third parties can observe or modify the traffic as it travels from the sender to the receiver. Sticking with the managed database example from before, each provider enables you to mandate encryption in-transit. Some providers make this easier than others, however. Here, Azure and GCP by making this as simple as turning on a single setting that is clearly visible in the web console. In order to accomplish the same effect in AWS, you need to create a configuration set called a "Parameter Group", enable the obscure "rds.force_ssl" setting on page 13, and associate the parameter group to your database instance. Given the legal liability you might face from improperly transmitting unencrypted data, you will have to take a lot of steps to ensure everything is above bar across all of the providers you utilize.



It can be extremely difficult to sort out the settings that matter from those that do not in the cloud.

5. Multicloud can Help with Availability, but only Slightly

When the cloud landscape was less mature and more geographically concentrated, you might have considered leveraging multiple clouds in order to improve your availability globally. Today, this is

arguably unnecessary. To demonstrate this, I put together a map of the approximate locations of the cloud infrastructure for Amazon Web Service, Azure, and GCP:

AWS's cloud infrastructure is represented with orange markers, while Azure's are blue and GCP's are green. https://www.google.com/maps/d/viewer?mid=1WKYlcUMJKnOnlvmGkn82Ek0TZAgKmFj3

AWS and Azure have fairly comparable and robust coverage, so you can host your resources in most regions of the world, barring Russia, with each. It is hard to make the same argument for GCP, which has zero presence in both Africa and China. Although Google claimed in 2018 that it will "offer mainland services", they have yet to deliver on this promise.

The parity between AWS and Azure can be perplexing as Azure claims it "has more global regions than any other cloud provider". While they are technically correct, with 58 regions vs. AWS's 23 and GCP's

22, it is important to consider that Azure defines a region in a dramatically different way than AWS does. AWS's regions consist of "Availability Zones" (AZs), which are a cluster of data centers within a region. Therefore, if you deploy your infrastructure across multiple AZs and one goes down, your infrastructure will still be available in the region. Until 2018, Azure did not have Availability Zones, which meant if you wanted High Availability, you would need to replicate your infrastructure to a different region or cloud provider. Although Azure is rolling out Availiability Zones to more regions and services, they still only cover 10 of their regions. With multiple AZs for each of its regions (except Osaka) and AZ support built into all of its services, AWS truly has the largest global footprint, and it is reflected by the massive amount of orange on the above map. That said, if you have the highly specific requirement of hosting your infrastructure in either Mexico or Qatar, Azure is the only way to go.

6. Multicloud is Coming to Your Organization, Sooner or Later

With some of the security and operational concerns above, you might be inclined to prevent your organization from going multicloud if it has not done so already. For better or for worse, this is a nearly impossible battle. Even if your organization is completely committed to one provider, you cannot predict what technical stacks you will inherit through mergers and acquisitions. If you obtain infrastructure on a cloud provider that is new to you, your organization has two choices: learn to support the new technology or migrate the acquired assets to your incumbent provider. The latter is ideal but could be tremendously costly. If the business determines that this cost is too great to justify the initiative, the organization will have no choice but to support the new cloud provider.

Even if you work in a smaller company that does not delve into M&A, it is key to remember why teams are wanting to adopt additional cloud solutions in the first place: they make their jobs easier or more enjoyable. At the end of the day, developers are creating the products that make the business money, not the central security team. If a team discovers a service offering that can help them get their product to

market quicker than the competition, they can and should use it, barring a catastrophic security issue or massive incursion of technical debt.

If development teams are prevented from being productive, many teams will create "Shadow IT" cloud accounts that are not managed or sanctioned by the enterprise. This is the worst possible outcome for an organization looking to implement some basic level of standardization in their cloud accounts. Not only does a shadow cloud account allow for "Wild West" development, but it eliminates the central organization's ability to monitor, scan, and get situational awareness of this account. Beyond this, these accounts pose legal and policy concerns as they are often procured outside of the official process. This is unsustainable. Eventually, the shadow cloud account's assets will need to be consolidated into an enterprise account, which is more costly and time-consuming than if the organization prevented the splinter in the first place.

Instead of futilely blocking the multicloud movement, security should embrace its inevitability and work to implement guardrails throughout the various cloud providers. If you'd like to learn more about how to do this, check out my upcoming SANS course, SEC510: Multicloud Security Assessment and Defense. The class prepares students to perform multicloud security assessments across AWS, Azure, and GCP clouds, identify key weaknesses in core cloud services, and apply hardened configurations. Vast swaths of the cloud are insufficiently documented, undocumented, or even incorrectly documented. To assess our risk in the cloud, we must look underneath the hood, and in SEC510, instead of merely citing best practices from each provider's documentation, we will validate that these recommendations work first-hand in a realistic multicloud lab environment. The class will cover the topics discussed in this article in greater detail and much more. Look out for the beta of the class, which is expected to launch this June!

About the Author:

Brandon Evans is an Instructor for the SANS Institute. He teaches SEC540: Cloud Security and DevOps Automation and is the lead author of the upcoming course SEC510: Multicloud Security Assessment and

Defense. His full-time role is as a Senior Application Security Engineer at Asurion, where he provides security services for thousands of his coworkers in product development across several global sites responsible for hundreds of web applications. This includes performing secure code reviews, conducting penetration tests, developing secure coding patterns, and evangelizing the importance of creating secure products.

Last Updated: June 4th, 2020

Upcoming SANS App Sec Training