Multi-Cloud Architecture and Governance

Leverage Azure, AWS, GCP, and VMware vSphere to build effective multi-cloud solutions

Jeroen Mulder

Multi-Cloud Architecture and Governance

Leverage Azure, AWS, GCP, and VMware vSphere to build effective multi-cloud solutions

Jeroen Mulder

BIRMINGHAM—MUMBAI

Multi-Cloud Architecture and Governance

Commissioning Editor: Vijin Boricha

Acquisition Editor: Yogesh Deokar

Senior Editor: Arun Nadar

Content Development Editor: Romy Dias

Technical Editor: Sarvesh Jaywant

Copy Editor: Safis Editing

Project Coordinator: Neil Dmello

Proofreader: Safis Editing

Indexer: Manju Arasan

Production Designer: Alishon Mendonca

First published: November 2020

Production reference: 2111220

Published by Packt Publishing Ltd.

Livery Place

35 Livery Street

Birmingham

B3 2PB, UK.

ISBN 978-1-80020-319-8

www.packt.com

14
Defining Security Policies

Whatever we do in the cloud, it needs to be secure. Cloud providers only provide tools. You need to define how to use these tools. In order to determine what these tools should do, you need to think about what type of assets you want to protect and how you need to protect them. There are quite a number of security baselines; for example, the baseline as defined by the **Center for Internet Security** (**CIS**), which provides guidelines.

We will learn what a security framework is and why it's important as a starting point for security policies. We will discover what we need to protect in our cloud environments. Next, we will look at the globally adopted CIS benchmark for Azure, AWS, and GCP and learn how to implement CIS using the security suites of these platforms. Lastly, we will learn what the difference is between security governance and management.

In this chapter, we're going to cover the following main topics:

- Understanding security frameworks
- Learning how to define security policies
- Learning how to implement security policies using the CIS benchmark
- Managing security policies

Understanding security policies

Let's start from our traditional, on-premises data center—a building traditionally used to host physical equipment that runs applications and stores data. The building is very likely secured by a fence and heavy, locked doors that can only be opened by authorized personnel. Access to the computer floors is also secured. There may be guards in the building or CCTV systems watching over equipment 24 hours a day. The next layer of defense is the access to the systems and data. Access to systems is strictly regulated: only authorized and certified engineers may access the systems. It's all common sense when it comes to running systems in a physical data center.

You would be surprised to see what happens if companies move these systems to cloud environments with **Infrastructure as a Service** (**IaaS**), **Platform as a Service** (**PaaS**), and **Software as a Service** (**SaaS**) solutions. For some reason, companies tend to think that by moving systems to the cloud, those systems are secured intrinsically, by default. That is not the case.

Platforms such as Azure, AWS, and GCP are probably the best secured platforms in the world. They have to be since they are hosting thousands of customers globally on them. But this doesn't mean that a company will not have to think about their own security policies anymore. The platforms provide a huge toolbox that enables the securing of workloads in the cloud, but what and how to protect these workloads is still completely up to the companies to implement themselves. We will need to establish and enforce our security policies in the cloud, think them through very carefully, and stick to them. That is what this chapter is about.

As with physical data centers, access needs to be regulated first by defining which identities are authorized to enter systems, and next, by determining what these entities are allowed to do in these systems. This is all part of identity and access management, a topic that we will cover in full in *Chapter 15*, *Designing Identity and Access Management*.

The foundation for security policies is the CIA principle:

- **Confidentiality**: Assets in the IT environment must be protected from unauthorized access.
- **Integrity**: Assets in the IT environment must be protected from changes that are not specified and unauthorized.
- **Availability**: Assets in the IT environment must be accessible for authorized users.

The security policy itself has nothing to do with technology. The policy merely defines the security principles and how these are safeguarded in the organization. The policy does not define what ports must be opened in a firewall or what type of firewall is required. The policy describes the requirement that assets belonging to a certain function in the enterprise must be protected at a certain level. For example, a business-critical functionality that is relying on a specific stack of applications needs to be available at all times and data loss must be zero. That will lead to an architectural design using mirrored systems, continuous backups, disaster recovery options, and a very strict authorization and authentication matrix for people who must be able to access these systems.

Understanding security frameworks

Security policies and forthcoming principles do not stand on their own. Typically, they are defined by industry or public frameworks to which a company must adhere. There are two types of frameworks: mandatory industry frameworks and best practices.

Examples of industry frameworks are the **Health Insurance Portability and Accountability Act** (**HIPAA**) for health care and the **Payment Card Industry** (**PCI**) data security standard for financial institutions. These frameworks were created to protect consumers by setting standards to avoid personal data – health status or bank accounts – being compromised. The cloud architect must have a deep understanding of these frameworks since they define how systems must be designed.

Next to these industry frameworks, there are some overall security standards that come from best practices. Examples are the standards of the **International Organization of Standardization** (**ISO**) and the U.S. **National Institute of Standards and Technology** (**NIST**). Specific to the cloud, we have the framework of the CIS.

Cloud providers have adopted the CIS framework as a benchmark for their platforms, as the internationally accepted standard for cybersecurity. The reason is that CIS maps to the most important industry and overall security frameworks such as ISO, NIST, PCI, and HIPAA. The controls of CIS take the principles from these frameworks into account, but it doesn't mean that by implementing CIS controls, a company is automatically compliant to PCI or HIPAA. CIS controls need to be evaluated per company and sometimes per environment.

Basically, there are two levels of CIS controls:

1. Essential basic security requirements that will not impact the functionality of the workloads or service.
2. Recommended settings for environments that require greater security but may impact workloads or services through reduced functionality.

In summary, CIS provides a security framework based on best practices. These are translated into benchmarks that can be adopted for specific platforms and systems: Azure, AWS, and GCP, and the instances in those clouds using operating systems such as Windows Server or various Linux distributions. These benchmarks lead to settings in hardening.

CIS offers recommendations for the following:

- Identity and access management
- Storage accounts
- Database services
- Logging and monitoring
- Networking
- Virtual machines
- Application services

In the next section, we will learn how to define the baseline for security policies.

Defining the baseline for security policies

It just takes a few mouse clicks to get a server up and running on any cloud platform. But in an enterprise that's migrating or creating systems in the cloud, there's a lot for an architect to think about – securing environments being the top priority. It is likely that IaaS, PaaS, and SaaS solutions will be used to build our environment. It could grow in complexity where a lack of visibility could lead to vulnerabilities. So, with every service enrolled in the cloud environment, we really need to consider how best to secure each service. Every service needs to be compliant with the security baseline and the policies defined in that baseline.

What are the steps for creating policies and the baseline?

1. **Check regulations**: Every company is subject to regulations. These can be legal regulations such as privacy laws or industry compliance standards. Make sure the regulations and compliance frameworks your company needs to adhere to are understood. Be sure to involve internal legal departments and auditors. This is the starting point in all cases.

   Also, check which security frameworks cloud providers have adopted. The major platforms – Azure, AWS, and GCP – are compliant with most of the leading compliance and security frameworks, but this may not be the case for smaller providers, for instance, specific SaaS solutions. Be aware that with SaaS, the provider controls the full stack: operating systems, hardware, network infrastructure, application upgrades, and patches. You have to be sure that this is done in a compliant way for your company.

2. **Restrict access**: This is what is often referred to as zero trust, although the term is even more related to network segmentation. But zero trust is also tightly connected to access management. We will have to design a clear **RBAC** model: **Role-Based Access Control**. Users have specific roles granting authorization to execute certain actions in cloud environments. If they don't have the appropriate role or the right authorization, they will not be able to execute actions other than the ones that have been explicitly assigned to that particular role. One term that is important in this context is least privilege: users only get the role and associated authorizations to perform the minimum number of actions that are really required for the daily job – and nothing more.

3. **Secure connections**: Cloud environments will be connected to the wide area network (WAN) of a company and to the outside world, the internet. The network is the route into cloud environments and must be very well secured. What connections are allowed, how are they monitored, what protocols are allowed, and are these connections encrypted? But also: how are environments in the cloud tenant segmented and how do systems in the tenant communicate with each other? Are direct connections between workloads in the cloud tenant allowed or does all traffic need to go through a centralized hub?

The security baseline should contain strict policies for all connectivity: direct connections, VPNs, in-transit encryption, traffic scanning, and network component monitoring. Again, we should think from the zero trust principle: network segmentation is crucial. The architecture must be designed in such a way that users can't simply hop from one segment of the environment to another. Segments must be contained and workloads inside the segments must be protected. A zero trust architecture typically has zones defined: for instance, a private zone where only inbound traffic is allowed or a public zone that has connections to the outside world. These zones are strictly separated from each other by means of a variety of security elements, firewalls, security groups, or access control lists.

4. **Protect the perimeter**: This is about protecting the outside of the cloud environment, the boundary. Typically, the boundary is where the connections terminate in the cloud environment. This can be a hub and that's where the gateways, proxy servers, and firewalls will be hosted. Typically, it also hosts the bastion host or jump server as a single point of entry where a user is allowed to gain access to the workloads in the environment.
5. **Protect the inside**: There will be workloads in our cloud: servers, applications, containers, and functions. Although there is boundary protection with gateways and firewalls, we must also protect our workloads, especially – but not limited to – the critical ones. These workloads must be hardened, reducing the vulnerability of systems with mandatorily applied security settings such as removing software components or disabling services that are not required to run on the system.
6. **Perform frequent audits**: This is a step that falls within managing security policies, which will be covered in the last section of this chapter. Security policies need to be constantly assessed. Hackers don't sit on their hands and will constantly think of ways to look for vulnerabilities. Therefore, it's necessary to continuously assess and audit policies and evaluate identified vulnerabilities. How critical are those vulnerabilities and what are the odds that our environments will get breached? Are we protected well enough? But also, how fast can action be taken if a vulnerability gets exploited and mitigate the consequences? This is not something that should be discussed once a month, but must be at the front of our minds at all times, for everyone developing or managing cloud environments.

We will need to define the scope of our security policies. One way to do that is by thinking in layers, derived from defense-in-depth as a common methodology in designing security architectures. Each layer comprises protective measures against specific threats. These layers are as follows:

- **Network layer**: As already stated in the previous section, the network is the entrance into our cloud environment. Networks need to be protected from unauthorized people getting in. Technologies to protect a network from threats are firewalls, **Intrusion Detection Systems** and **Intrusion Prevention Systems** (**IDSes/IPSes**), public key infrastructure, and network segmentation, preferably adhering to zero trust principles.
- **Platform layer**: Typically, this is the layer of the operating system. Systems should be fully patched with the latest fixes for (possible) vulnerabilities and hardened. Also, pay attention to ports that are opened on a system. Any port that is not required should be disabled.
- **Application layer**: This layer is not only about the application but also about middleware and APIs communicating directly with the application. Application code must be secured. Static code analysis can be very helpful and is strongly advised. Static program analysis is performed without actually executing software, validating the integrity of source code so that any attempt to change code or software parameters is detected.
- **Data layer**: This is the holy grail for hackers, the very target of almost every hacker. If a hacker succeeds to get through the first three layers – network, platform, and application – then the next layer is the data itself. We will extensively discuss data security in *Chapter 16*, *Defining Security Policies for Data*. All critical data should be encrypted, in transit and at rest.
- **Response layer**: This is the layer for all security monitoring, typically the layer for **Security Information and Event Management** systems (**SIEM**). This is the layer where all suspicious activity is captured, analyzed, and translated into triggers to execute mitigating actions.

Security policies must be defined and applied at each layer. Now, let's look at some best practices for security policies:

- **Access**: Only use named accounts to allow access to systems. Be extremely selective when granting global admin rights, implement role-based access, and use multi-factor authentication. In the next chapter, we will go into this subject in more detail.
- **Perimeter or boundary protection**: Implement firewalls or use the native firewalls from the cloud platforms. A recommended practice is to have the firewall set to "block all" as the default and then open up ports as per the requirements of a certain workload or functionality. Only have ports open when there's a validated reason.
- **Public Key Infrastructure (PKI)**: Public and private keys are used to verify the identity of a user before data is transferred. Breached passwords are still the number one root cause for compromised systems and data leaks. Therefore, it's recommended not to use passwords, but instead use keys, securely stored in a key vault. All major cloud providers offer PKI services and key vault solutions.
- **Logging and audit trail**: Be sure that you know what happens in your cloud environment, at all times. Even with the most rigid security policy, a company should never fully rely on security measures alone. Monitoring and an audit trail are highly recommended (or required, even) best practices.

Now it's time to discover how these policies should be implemented using the native security suites in Azure, AWS, and GCP.

Implementing security policies

We have studied the compliance and security frameworks and we've defined our security baseline. Now we need to implement it in our cloud environments. In this section, we will explore implementations in the major clouds, using the native security platforms. Since CIS is widely and globally adopted as the baseline for security policies, all sections will explore specific settings that CIS benchmarks recommend for the different platforms. Links to the benchmarks are provided in the *Further reading* section of this chapter. CIS provides recommendations, but also documents how policies should be implemented.

For example, in GCP there is a recommendation to "ensure Cloud Audit Logging is configured properly across all services and all users from a project." CIS benchmarks also guide users to find where the setting needs to be configured and how; in this example, by going to audit logs at `https://console.cloud.google.com/iam-admin/audit` or by configuring it from the command line:

```
gcloud organizations get-iam-policy ORGANIZATION_ID
gcloud resource-manager folders get-iam-policy FOLDER_ID
gcloud projects get-iam-policy PROJECT_ID
```

The format in the CIS benchmarks is always the same, for all cloud platforms.

Implementing security policies in Azure Security Center

Azure Security Center is a native service of Azure. In other words, you don't need to install or configure anything. From the Azure console, Security Center can be accessed immediately by simply enabling it. It then starts monitoring workloads that you have deployed in Azure: virtual machines, databases, storage accounts, networking components, and other Azure services.

However, policies will need to be configured in Security Center. CIS lists some recommendations specific to Azure Security Center. The most important one is to activate the standard pricing tier in Security Center: this enables threat detection for all networks and VMs in the Azure tenant. Every CIS recommendation to implement a policy comes with an explanation. In the case of enabling the standard pricing tier, the rationale is that it allows greater defense-in-depth, with threat detection provided by the **Microsoft Security Response Center** (**MSRC**).

Enabling the standard pricing tier and adjusting settings is done through the **Security Center** blade in the portal at `https://portal.azure.com/#home`, as shown in the following screenshot:

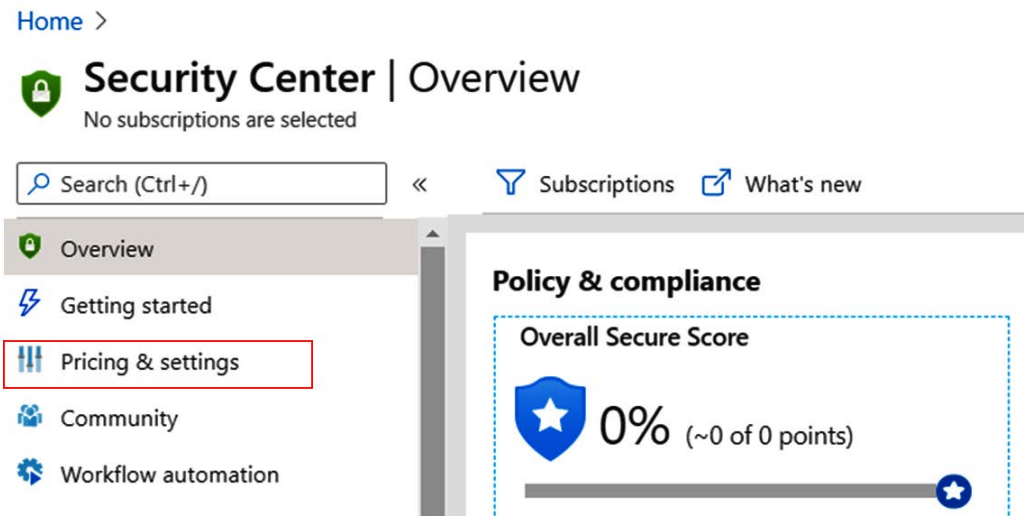

Figure 14.1 – Overview of the Security Center blade in the Azure portal

The next action is to enable the monitoring agent to actually collect the data and make sure that the default policy setting, **Monitor system updates**, is not set to **Disabled**. Enabling this setting retrieves a daily list of available security and critical updates from Microsoft, both for Windows systems and for systems that run Linux distributions. These are the basic configuration settings to get Security Center started.

The next step is to implement the security settings. In Security Center, enable settings for the following:

- Scanning vulnerabilities in operating systems
- Enforcing endpoint protection
- Monitoring disk encryption
- Monitoring network security groups
- Monitoring web application firewalls
- Monitoring next-generation firewalls
- Vulnerability assessment
- Monitoring blob storage encryption

- Monitoring **just-in-time** (**JIT**) network access
- Monitoring adaptive application whitelisting
- Monitoring SQL auditing
- Monitoring SQL encryption

Lastly, there are a few settings that enable communication in case of high-severity alerts, by sending email notifications or text messages.

> **Tip**
>
> Azure has something more than just Azure Security Center: Azure Sentinel, a native SIEM solution. Sentinel is an intelligent defense-in-depth solution, especially when activating the security framework of MITRE ATT&CK® in Sentinel. ATT&CK is a knowledge base that is constantly updated with the latest threats and known attack strategies. A group of developers under the name of BlueTeamLabs have published templates and code to implement ATT&CK in Sentinel. It's worthwhile taking a look at this at `https://github.com/BlueTeamLabs/sentinel-attack`.

Implementing security policies in AWS Security Hub

AWS offers a single security dashboard with AWS Security Hub. The solution aggregates monitoring alerts from various security solutions, such as CloudWatch and CloudTrail, but also collects findings from Amazon GuardDuty, Amazon Inspector, Amazon Macie, AWS **Identity and Access Management** (**IAM**) Access Analyzer, and AWS Firewall Manager. CloudTrail, however, is the key element in Security Hub. CloudTrail constantly monitors the compliance of accounts that are used in the AWS environment. It also performs operational auditing and risk auditing, meaning it keeps track of all activity that is started from the console in your environment, enables analysis of changes to resources, and detects unusual activity. It's fair to say that CloudTrail is the engine underneath Security Hub.

Security Hub makes it easy to start monitoring all activity in your AWS environment. It's accessible from the AWS console, as shown in the following screenshot:

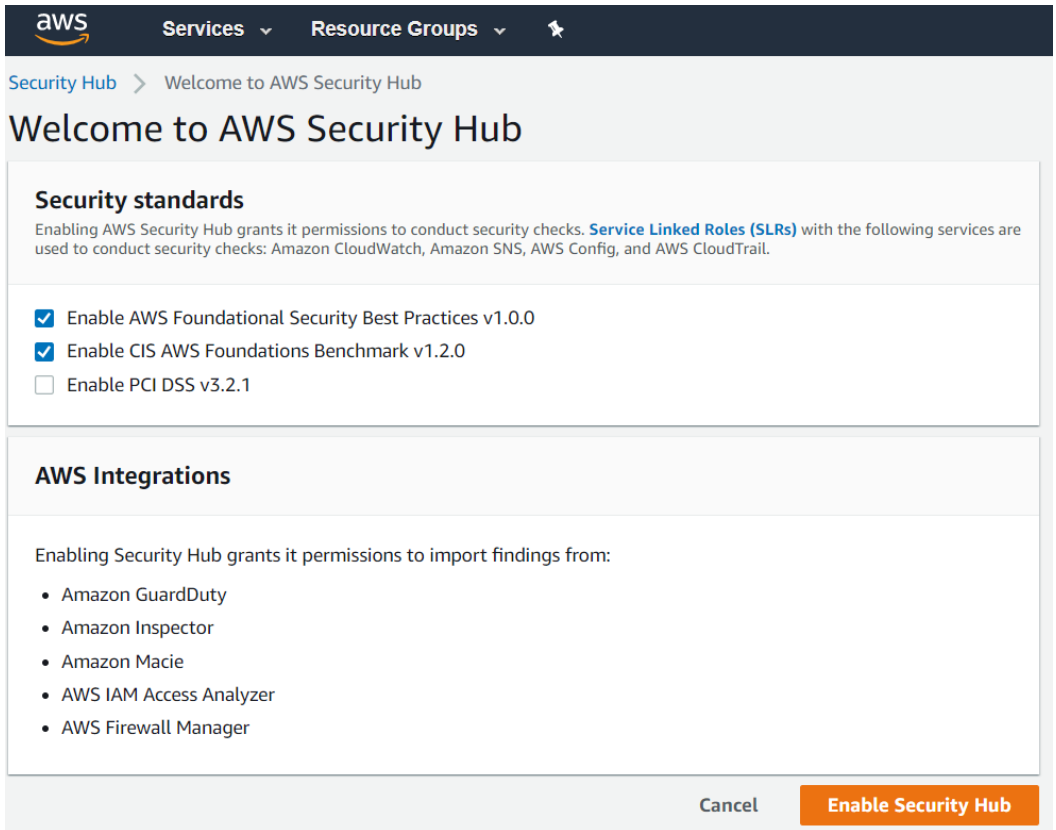

Figure 14.2 – Accessing Security Hub in the AWS console

There are a couple of things that need explaining in the preceding screenshot. The top part of the screen shows the security baselines that can be enrolled by default: **Enable AWS Foundational Security Best Practices v1.0.0** and **Enable CIS AWS Foundations Benchmark v1.2.0** have been ticked by default. The third one is the PCI DSS framework. **PCI DSS** stands for **Payment Card Industry Data Security Standard** and is specific to financial institutions.

In the lower part of the screen, we see all the integrations that Security Hub offers:

- **GuardDuty**: Amazon's solution for threat detection.
- **Inspector**: This tool assesses applications for exposure, vulnerabilities, and deviations from best practices valid for these applications.

- **Macie**: This solution monitors the data security and data privacy of your data stored in Amazon S3 storage.
- **IAM Access Analyzer**: This tool keeps track of accounts accessing environments in AWS and whether these accounts are still compliant with security policies.
- **Firewall Manager**: This tool enables centralized management of all firewalls in the AWS environment.

By clicking the **Enable Security Hub** button, the mentioned baselines with the named integrations will be enrolled.

The CIS baseline should definitively be implemented as the worldwide accepted standard for securing online environments. Specific to AWS, CIS includes the following recommendations for settings to control security policies:

- Ensure CloudTrail is enabled in all regions.
- Ensure CloudTrail log file validation is enabled.
- Ensure that an S3 (storage) bucket used to store CloudTrail logs is not publicly accessible.
- Ensure CloudTrail logs are integrated with CloudWatch logs.
- Ensure AWS Config is enabled in all regions.
- Ensure S3 bucket access logging is enabled on CloudTrail S3 bucket.
- Ensure CloudTrail logs are encrypted at rest using **KMS CMKs** (**Key Management Services – Customer Master Keys**).
- Ensure rotation for customer-created CMKs is enabled.
- Ensure **Virtual Private Cloud** (**VPC**) flow logging in all VPCs.

Obviously, these are not all the settings: these are the most important settings for getting the logging and monitoring of security policies right. In the *Further reading* section, we include links to the various CIS benchmarks for the major clouds.

Implementing security policies in GCP Security Command Center

In GCP, we will have to work with Security Command Center. You can manage all security settings in Security Command Center and view the compliancy status from one dashboard. The concept is the same as AWS Security Hub – Security Command Center in GCP comprises a lot of different tools to manage security in GCP environments. In the GCP cloud console, we'll see **Security** in the main menu. Hovering over the **Security** subheading will pop up the products and tools that are addressed in **Security Command Center**, as shown in the following screenshot:

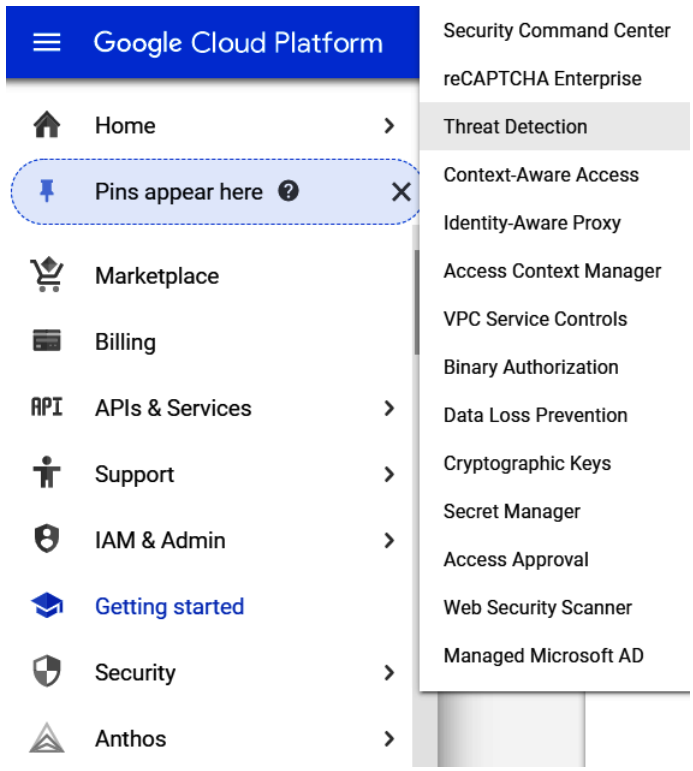

Figure 14.3 – Launching Security Command Center in the cloud console of GCP

Security Command Center does an inventory and discovery of all assets in the GCP environments and, next, starts monitoring them in terms of threat detection and prevention. One special feature that needs to be discussed here is Google Cloud Armor. Cloud Armor started as a defense layer to protect environments in GCP from **Distributed Denial of Services** (**DDoS**) and targeted web attacks. Cloud Armor has since been developed to a full security suite in GCP to protect applications using the functionality of **Web Application Firewalls** (**WAFs**).

Cloud Armor can be launched from the GCP console at `https://console.cloud.google.com/`. You won't find it under **Security Command Center**, but under **Network Security**, as shown in the following screenshot:

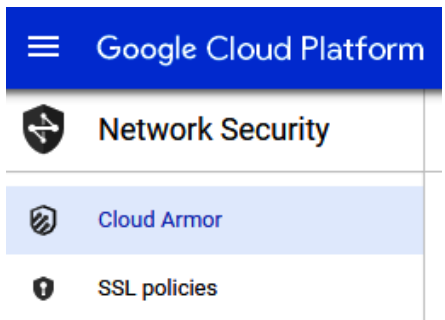

Figure 14.4 – Menu of Cloud Armor in GCP

We can specify security policies in Cloud Armor, but GCP already included a list of policies that can be evaluated. These preconfigured policies are based on **OWASP** CRS—the **Open Web Application Security Project**, a community that strives to find methodologies and practices to constantly improve the protection of online applications. **CRS** stands for **Core Rule Set**. Cloud Armor includes the top 10 OWASP threats in rule sets. The number one threat is the injection of hostile code in order to breach the application and get access to data. In *Chapter 16*, *Defining Security Policies for Data*, we will explore OWASP in more detail since this is all about securing applications and data.

However, OWASP does overlap with CIS, but OWASP merely identifies the threats, whereas CIS makes recommendations to avoid vulnerabilities and the chances of threats really being exploited. Misconfigured security, for example, is number 6 in the top 10 of OWASP. Insufficient logging and monitoring concludes the top 10. Both are heavily addressed in CIS.

The CIS 1.1.0 benchmark for GCP was released in March 2020. Specifically, for logging and monitoring, CIS recommends the following settings to audit security policies:

- Ensure Cloud Audit Logging is configured properly across all services and users in a project.
- Ensure sinks are configured for all log entries.

> **Note**
> A sink will export copies of all the log entries.

- Retention policies on log buckets must be configured using Bucket Lock.

- Ensure log metric filters and alerts exist for project ownership assignments and changes.
- Ensure log metric filters and alerts exist for audit configuration changes.
- Ensure log metric filters and alerts exist for custom role changes.
- Ensure log metric filters and alerts exist for VPC Network Firewall rule changes.
- Ensure log metric filters and alerts exist for VPC Network Route changes.
- Ensure log metric filters and alerts exist for VPC Network changes.
- Ensure log metric filters and alerts exist for cloud storage IAM permission changes.
- Ensure log metric filters and alerts exist for SQL instance configuration changes.

As with Azure and AWS, these are the settings to audit the security policies against the CIS benchmark. In the *Further reading* section, we included links to the various CIS benchmarks for the major clouds.

Managing security policies

It doesn't stop with implementing security policies. We need to have governance in place to manage the policies. Governance is required on two levels:

1. The security policies themselves, auditing these to the compliancy frameworks that a business has to adhere to.
2. The technical implementation of the security policies, keeping the monitoring up to date, making sure that all assets are indeed tracked against the policies.

The first level is the domain of people concerned with the security governance in a business, typically, a **Chief Information Security Officer** (**CISO**) or **Chief Information Officer** (**CIO**). They need to set directions for security policies and make sure that the business is compliant with the security strategy, industry, and company frameworks. The CISO or CIO is also responsible for assurance from internal and external auditing.

Level two is more about security management, concerning how to deal with security risks in the IT landscape, including the cloud environments. To make it simple: security governance is about making policies; security management is about (technically) implementing and enforcing policies. So, security engineers should worry about the management of security monitoring tools that were covered in this chapter. They will need to understand how to implement rule sets in Azure Security Center, AWS Security Hub, and Google Security Command Center. They will also need to know what to do in the event of an alert being raised in these systems, who should follow up, and what actions need to be taken. Those will be technical actions, such as isolating an environment when it's breached. The configuration of rules in the security suites is also in their hands.

However, the security policies themselves need to be defined from a higher level in the business. The CISO or CIO will hardly ever completely understand how to program the security console, but they will know what needs to be protected from a business perspective. Obviously, the strategic level of CISO/CIO can't do anything without input from the tactical level – the security architects and engineers. They will all have to work closely together.

Summary

In this chapter, we discussed the basics of security frameworks as a starting point to define policies for cloud environments. We have learned that there are different frameworks and that it depends on the industry to determine the compliance requirements of a business. Next, we must decide which security controls to set to ensure that our cloud environments are compliant too.

One framework that is globally accepted and commonly used for clouds is CIS. For Azure, AWS, and GCP, we studied the CIS benchmarks. We learned that the CIS benchmarks for these cloud platforms greatly overlap, but also have specific settings that need to be implemented in the respective security suites – Azure Security Center, AWS Security Hub, and Google's Security Command Center.

In the last section, we learned the difference between security governance and security management, but also that one can't live without the other.

In the next chapter, we will dive into identity and access management, since that's where security typically starts: who is allowed to do what, how, and maybe even when in our cloud environments? In *Chapter 17*, *Using Security Monitoring and Management Tools*, we will further explore the use of the monitoring tools that we discussed briefly in this chapter.

Questions

1. We've discussed the CIA principle. What does it stand for?
2. What are public and private keys used for in terms of PKI?
3. All major cloud platforms have adopted a certain security baseline. Name the framework that comprises this baseline.
4. True or false: the CIO should be concerned with security management.

Further reading

You can refer to the following links for more information on the topics covered in this chapter:

- The CIS framework: `https://www.cisecurity.org/`
- The download page for CIS Benchmark for Azure: `https://azure.microsoft.com/en-us/resources/cis-microsoft-azure-foundations-security-benchmark/`
- The CIS baseline for AWS: `https://d0.awsstatic.com/whitepapers/compliance/AWS_CIS_Foundations_Benchmark.pdf`
- The CIS benchmark for GCP: `https://www.cisecurity.org/benchmark/google_cloud_computing_platform/`
- Link to the OWASP community pages: `https://owasp.org/www-project-top-ten/#:~:text=The%20OWASP%20Top%2010%20is%20the%20reference%20standard,software%20development%20culture%20focused%20on%20producing%20secure%20code`
- *Enterprise Cloud Security and Governance*, by Zeal Vora, Packt Publishing