**16**

**Defining Security Policies for Data**

Data is an important asset of any company. Enterprises store their data more and more in multi-cloud setups. How do they secure data? All cloud platforms have technologies to encrypt data but differ in how they apply encryption and store and handle keys. But data will move from one cloud to another or to user devices, so it needs to be secured in transit, next to data at rest. This is done with encryption, using encryption keys. These keys need to be secured as well, preventing non-authorized users from accessing the keys and encrypted data.

Before we discuss data protection itself, we will briefly talk about data models and how we can classify data. We will explore the different storage solutions the major clouds offer. Next, we will learn how data can be protected by defining policies for **data loss prevention** (**DLP**), labeling information to control access, and using encryption.

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

* Storing data in multi-cloud concepts
* Exploring storage solutions
* Understanding data encryption
* Using encryption in Azure, AWS, GCP, OCI, and Alibaba Cloud
* Securing access, encryption, and storage keys
* Securing raw data for big data modeling

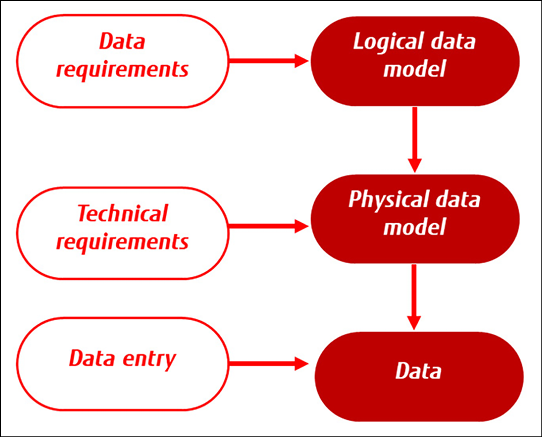
**Storing data in multi-cloud concepts**

If you ask a **chief information officer** (**CIO**) what the most important asset of the business is, the answer will very likely be *data*. The data architecture is therefore a critical part of the entire business and IT architecture. It’s probably also the hardest part of business and IT architecture. In this section, we will briefly discuss the generic principles of data architecture and how this drives data security in the cloud.

Data architecture consists of three layers—or **data architecture processes**—in enterprise architecture:

* **Conceptual**: A conceptual model describes the relationship between business entities. Both products and customers can be entities. A conceptual model connects these two entities: there’s a relationship between a product and the customer. That relationship can be a sale: the business selling a product to a customer. Conceptual data models describe the dependencies between business processes and the entities that are related to these processes.
* **Logical**: The logical model holds more detail than the conceptual model. An enterprise will likely have more than one customer and more than one product. The conceptual model only tells us that there is a relation between the entity customer and the entity product. The next step would be to define the relationship between a specific customer and a product. Customers can be segregated by adding, for instance, a customer number. The conceptual model only holds the structure; the logical model adds information about the customer entity, such that customer X has a specific relationship with product Y within the entity product.
* **Physical**: Neither conceptual nor logical models say anything about the real implementation of a data model in a database. The physical layer holds the blueprint for a specific database, including the architecture for location, data storage, or database technology.

The following diagram shows the relationship between conceptual data modeling and the actual data—data requirements are set on a business level, leading to technical requirements for the storage of the data and, eventually, the data entry itself:



*Figure 16.1: Concept of data modeling*

**Data modeling** is about structuring data. It doesn’t say anything about the data types or data security, for that matter. Those are the next steps.

Each data type needs to be supported by a data model. Each data type drives the choice of the technology used to store and access data. Common data types are integers (numeric), strings, characters, and Booleans. The latter might be better known as *true/false statements* since a Boolean can only have two values. Alongside these, there are abstract types, such as stacks (a data structure where the last entered data is put on top of the stack), lists (countable sequences), and hashes (associative mappings of values).

Data types are not related to the content of the data itself. A numeric type doesn’t say whether data is confidential or public. So, after the model and the definition of data types, there’s also data classification. The most common labels for classification are public, confidential, sensitive, and personal data. For example, personal data needs to be highly secured. There are national and international rules and laws forcing organizations to protect personal data at the highest possible level, meaning that no one should be able to access this data without reasons justified by legal authorities. This data will be stored in strings, arrays, and records, and will likely have a lot of connections to other data sources.

An architect will have to think about security on different layers to protect the data, including the data itself, the database where the data is stored, and the infrastructure that hosts the database. If the data model is well-architected, there will be a good overview of what the dependencies and relationships are between data sources.

But what about storing data in multi-cloud concepts, using different providers? This approach can provide a number of benefits, including increased availability, disaster recovery, and cost savings.

Let’s start with availability. By storing data across multiple cloud providers, enterprises will have the possibility to still access data even if one provider experiences an outage or other disruption. But there are more advantages. Spreading data across cloud providers can increase flexibility, by making use of the best options that providers have for specific use cases. An example might be to have certain applications and data in one environment, using specific PaaS database services. At the same time, the enterprise can have a data lake in another cloud, accessed by data scientists and analysts for research purposes.

Some companies have a multi-cloud strategy for data for reasons of business continuity and disaster recovery. By storing data in multiple locations, enterprises can ensure that their data is protected against a wide range of potential threats, including natural disasters, power outages, and cyber-attacks such as ransomware holding data hostage. This can help to minimize the risk of data loss and ensure that critical business operations can continue even in the event of an emergency.

Lastly, storing data in various clouds and, by doing so, making use of the best propositions of the provider, can be a way to lower costs. This can be achieved by using different pricing models, as we have seen in *Chapter 12*, *Cost Modeling in the Cloud*. However, this should never be the first goal. Protecting the data must be priority number one.

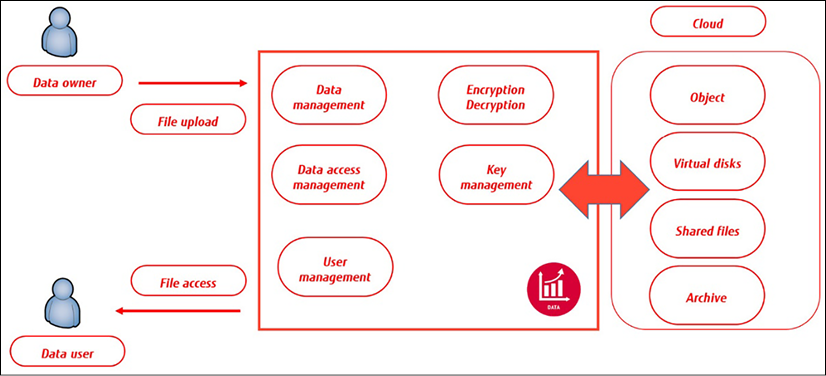
Since we need storage to hold data, we will discuss the different types of storage in the next section.

**Exploring storage technologies**

Aside from data modeling and data types, it’s also important to consider the storage technologies themselves, including data lakes, queues, and other types of data services. All cloud platforms offer services for the following:

* **Object storage**: Object storage is the most used storage type in the cloud; we can use it for applications, content distribution, archiving, and big data. In Azure, we can use Azure Blob Storage; in AWS, **Simple Storage Services** (**S3**); and GCP simply calls it Cloud Storage. The object storage in OCI is Object Storage. Good to know: OCI also supports S3 compatibility, which allows existing S3 applications to work seamlessly with OCI Object Storage. The object storage service in Alibaba Cloud is simply called **Object Storage Service** (**OSS**).
* **Virtual disks**: Virtual machines will either be composed of a virtual disk of block storage or ephemeral. Since every component in the cloud is virtualized and defined as code, the virtual disk is also a separate component that must be specified and configured. There are a lot of choices, but the key differentiators are the required performance of a disk. For I/O-intensive read/write operations to disks, it is advisable to use **solid-state drives**(**SSDs**).
* **Shared files**: To organize data in logical folders that can be shared for usage, filesystems are created. The cloud platforms offer separate services for filesystems—in Azure, it’s simply called **Files**; in AWS, **Elastic File System** (**EFS**); and in GCP, the service is called **Filestore**. GCP does suggest using persistent disks as the most common option. In Alibaba Cloud, the suggested service is called EFS too. OCI provides OCI File Storage to create and manage file systems and mount them as NFS or SMB shares. These protocols are also supported by Azure Files. These shares are then accessible from on-premises or other cloud environments. Alibaba Cloud also provides a file storage service called Apsara File Storage using **NAS** (**network attached storage**) technology.
* **Archiving**: The final storage tier is archiving. For archiving, high-performance SSDs are not required. To lower storage costs, the platforms offer specific solutions to store data that is not frequently accessed but that needs to be stored for a longer period of time, referred to as a retention period. Be aware that the storage costs might be low but the cost of retrieving data from archive vaults will typically be higher than in other storage solutions. In Azure, there’s a storage archive access tier, whereas AWS offers S3 Glacier and Glacier Deep Archive. In GCP, there are Nearline, Coldline, and Archive—basically, different tiers of archive storage. OCI offers Archiving Storage. The archiving service that is recommended in Alibaba Cloud is Cold Storage.

The following diagram shows the relationship between the data owner, the actual storage of data in different solutions, and the data user:



*Figure 16.2: Conceptualized data model showing the relationship between the data owner, data usage, and the data user*

All mentioned solutions are ways to store data in cloud environments. In the next section, the principles of data protection are discussed.

**Understanding data protection in the cloud**

In a more traditional data center setup, an enterprise would probably have physical machines hosting the databases of a business. Over the last two decades, we’ve seen tremendous growth in data, up to the point where it has almost become impossible to store this data in on-premises environments—something that is often referred to as **big data**. That’s where public clouds entered the market.

With storing data in external cloud environments, businesses were confronted with new challenges to protect this data. First of all, the data was no longer in on-premises systems but in systems that were all handled by third-party companies. This means that data security has become a shared responsibility—the cloud provider needs to offer tools and technologies to be able to protect the data on their systems, but the companies themselves still need to define what data they need to protect and to what extent. It’s still largely the responsibility of the enterprise to adhere to compliance standards, laws, and other regulations.

There’s more to consider about data than its current or live state. An enterprise should be equally concerned about historical data that is archived. Too often, data protection is limited to live data in systems, but should also focus on archived data. Security policies for data must include live and historical data. In the architecture, there must be a mapping of data classification and there must be policies in place for DLP.

Data classification enables companies to apply labels to data. DLP prevents sensitive data from being transferred outside an organization. For that, it uses business rules and data classification. DLP software prevents classified data from being accessed and transferred outside the organization. To set these rules, data is usually grouped, based on the classification. Next, definitions of how data may be accessed and by whom are established. This is particularly important for data that can be accessed through APIs, for instance, by applications that connect to business environments. Business data in a **customer relationship management** (**CRM**) system might be accessed by an application that is also used for the sales staff of a company, but the company wouldn’t want the data to be accessed by Twitter. A company needs to prevent business data from being leaked to other platforms and users than those authorized.

To establish a policy for data protection, companies need to execute a **data protection impact analysis** (**DPIA**). In a DPIA, an enterprise assesses what data it has, what the purpose of that data is, and what the risk is when the data is breached. The outcome of the DPIA will determine how data is handled, who or what should be able to access it, and how it must be protected. This can be translated into DLP policies. The following table shows an example of a very simple DLP matrix. It shows that business data may be accessed by a business application, but not from an email client. Communication with social media—in this example, Twitter—is blocked in all cases. In a full matrix, this needs to be detailed:

|  |  |  |  |
| --- | --- | --- | --- |
| Connection | Data Source | | |
| Business Data | Email | Twitter |
| Business Application | Allowed | Denied | Blocked |
| Email | Denied | Allowed | Blocked |
| Twitter | Blocked | Blocked | N/A |

*Table 16.1: Example of data mapping*

Labeling and DLP are about policies: they define what must be protected and to what extent. The next consideration is the technologies to protect the data—the how.

**Understanding data encryption**

One of the first, if not *the* first, encryption devices to be created was the Enigma machine. It was invented in the 1920s and was mostly known for its usage in World War II to encrypt messages. The British scientist Alan Turing and his team managed to crack the encryption code after 6 months of hard work.

The encryption that Enigma used, in those days, was very advanced. The principle is still the same: we translate data into something that can’t be read without knowing how the data was translated. To be able to read the data, we need a way to decipher or decrypt the data. There are two ways to encrypt data—asymmetric, or public key, and symmetric. In the next section, we will briefly explain these encryption technologies, before diving into the services that the leading cloud providers offer in terms of securing data.

First, let’s get into two forms of encryption: at rest and in transit. Data that is stored on a (virtual) device or in a database can be protected using encryption at rest. If the data is accessed by unauthorized users, these users won’t be able to read the data unless they have the appropriate decryption key. A typical method of encryption is **full disk encryption** (**FDE**), commonly implemented on end user devices using, for instance, BitLocker. This works the same in the cloud. For example, FDE can be implemented on VMs in Azure using Azure Disk Encryption, and in AWS, with the encryption feature of EBS.

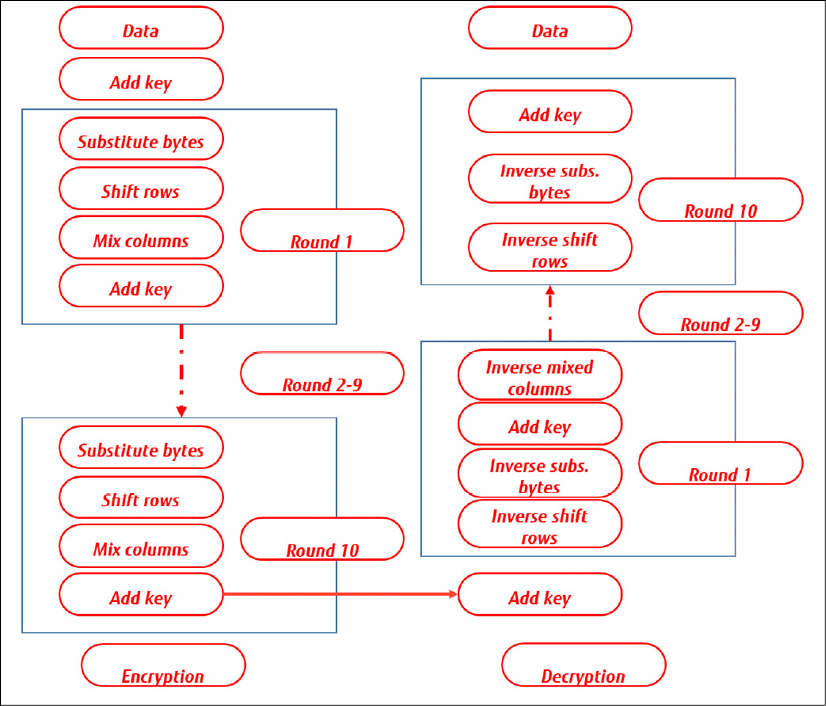
Another method is the use of database encryption, which encrypts the data stored within a database. This can be done through the use of built-in encryption features of the database management system or through the use of third-party encryption software.

Encryption in transit, on the other hand, refers to the process of protecting data as it is transmitted over a network. This type of encryption is used to protect data from being intercepted and read by an attacker while it is being transmitted from one device to another. There are multiple ways of implementing encryption in transit. **Transport Layer Security** (**TLS**) is used as a protocol to encrypt data between a web server and a browser. Another method is using VPNs.

There’s one really important note that must be made: all efforts to encrypt data are useless if encryption keys are not secured too. Encryption uses an encryption algorithm and an encryption key—symmetric or asymmetric. With **symmetric**, the same key is used for both encrypting and decrypting. The problem with that is that the entity that encrypts a file needs to send the key to the recipient of the file so it can be decrypted. The key needs to be transferred. Since enterprises use a lot of (different types of) data, there will be a massive number of keys. The distribution of keys needs to be managed well and be absolutely secure.

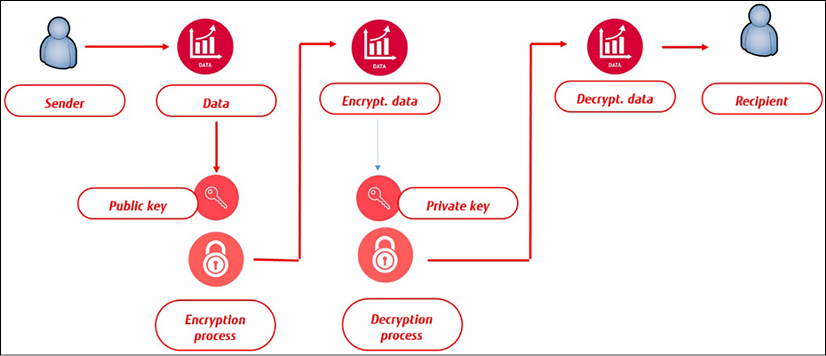
An alternative is **asymmetric** encryption, which uses a private and a public key. In this case, the company only needs to protect the private key, since the public key is commonly available.

Both encryption methods are used. A lot of financial and governmental institutions use **AES**, the **Advanced Encryption Standard**. AES works with data blocks. The encryption of these blocks is performed in rounds. In each round, a unique code is included in the key. The length of the key eventually determines how strong the encryption is. That length can be 128-, 192-, or 256-bit. Recent studies have proven that AES-256 is even quantum-ready. The diagram shows the principle of AES encryption:



*Figure 16.3: Simple representation of AES encryption principle*

**RSA**, named after its inventors, **Rivest, Shamir, and Adleman**, is the most popular asymmetric encryption method. With RSA, the data is treated as one big number that is encrypted with a specific mathematical sequence called **integer factorization**. In RSA, the encryption key is public; decryption is done with a highly secured private key. The principle of RSA encryption is shown in the following diagram:



*Figure 16.4: Concept of RSA encryption*

Both in AES and RSA, the length of the keys is crucial. Even today, the most common way to execute an attack on systems and retrieve data is by brute force, where the attacker will fire random keys on a system until one of the keys matches. With high-performance computers or computer networks executing the attack, the chance of success is still quite high. So, companies have to think about protecting the data itself, but also protecting the keys used to encrypt data.

In the next section, we will explore the different solutions in public clouds for storage and secure keys, and finally, draw our plan and create the data security principles.

**Securing access, encryption, and storage keys**

Cloud platforms provide customers with technology and tools to protect their assets, including the most important one—data. At the time of writing, there’s a lot of debate about who’s responsible for protecting data, but generally, the company that is the legal owner of the data has to make sure that it’s compliant with (international) laws and standards. In the UK, companies have to adhere to the Data Protection Act, and in the European Union, all companies have to be compliant with the **General Data Protection Regulation** (**GDPR**).

Both the Data Protection Act and GDPR deal with privacy. International standards ISO/IEC 27001:2022 and ISO/IEC 27002:2022 are security frameworks that cover data protection. These standards determine that all data must have an owner so that it’s clear who’s responsible for protecting the data. In short, the company that stores data on a cloud platform still owns that data and is therefore responsible for data protection.

To secure data on cloud platforms, companies have to focus on two aspects:

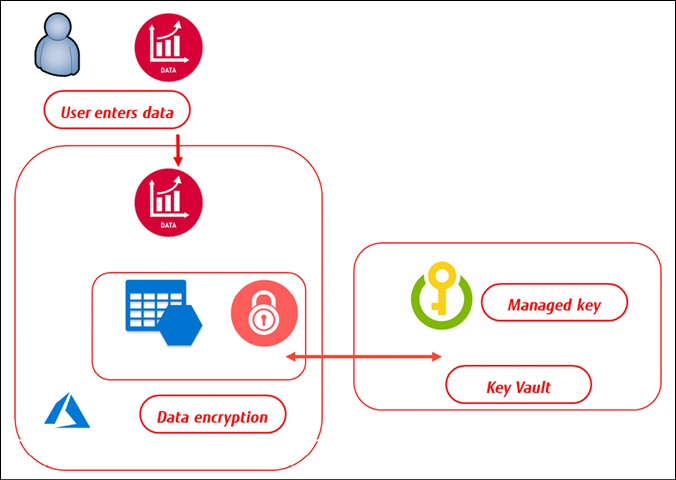
* Encryption
* Access, using authentication and authorization

These are just the security concerns. Enterprises also need to be able to ensure reliability. They need to be sure that, for instance, keys are kept in a separate place from the data itself and that even if a key vault is not accessible for technical reasons, there’s still a way to access the data in a secure way. An engineer can’t simply drive to the data center with a disk or a USB device to retrieve the data. How do Azure, AWS, and GCP take care of this? We will explore this in the next section.

**Using encryption and keys in Azure**

In Azure, the user writes data to Azure Blob Storage. The storage is protected with a storage key that is automatically generated. Data in a new storage account is encrypted with Microsoft-managed keys by default. You can continue to rely on Microsoft-managed keys for the encryption of your data, or you can manage encryption with your own keys.

The storage keys are kept in a key vault, outside the subnet where the storage is itself. But the key vault does more than just store the keys. It also regenerates keys periodically by rotation, providing **shared access signature** (**SAS**) tokens to access the storage account. The concept is shown in the following diagram:



*Figure 16.5: Concept of Azure Key Vault*

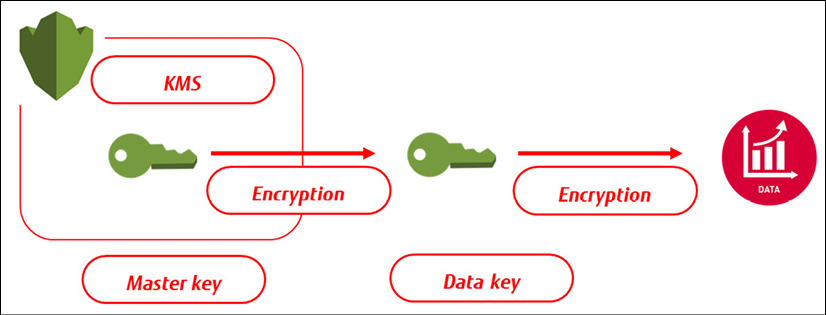
The key vault is highly recommended by Microsoft Azure for managing encryption keys. Encryption is a complex domain in Azure since Microsoft offers a wide variety of encryption services in Azure. Disks in Azure can be encrypted using BitLocker or DM-Crypt for Linux systems. With Azure, **Storage Service Encryption** (**SSE**) data is automatically encrypted before it’s stored in a blob. SSE uses AES-256. For Azure SQL databases, Azure offers encryption for data at rest with **Transparent Data Encryption** (**TDE**), which also uses AES-256 and **Triple Data Encryption Standard** (**3DES or TDES**).

**Using encryption and keys in AWS**

Like Azure, AWS has a key vault solution, called **Key Management Service** (**KMS**). The principles are also very similar, mainly using server-side encryption. Server-side means that the cloud provider is requested to encrypt the data before it’s stored on a solution within that cloud platform. Data is decrypted when a user retrieves the data. The other option is client-side, where the customer takes care of the encryption process before data is stored.

The storage solution in AWS is S3. If a customer uses server-side encryption in S3, AWS provides S3-managed keys (SSE-S3). These are the unique **data encryption keys** (**DEKs**) that are encrypted themselves with a master key, the **key encryption key** (**KEK**). The master key is constantly regenerated. For encryption, AWS uses AES-256.

AWS offers some additional services with **customer master keys** (**CMKs**). These keys are also managed in KMS, providing an audit trail to see who has used the key and when. Lastly, there’s the option to use **customer-provided keys** (**SSE-C**), where the customer manages the key themselves. The concept of KMS using CMKs in AWS is shown in the following diagram:



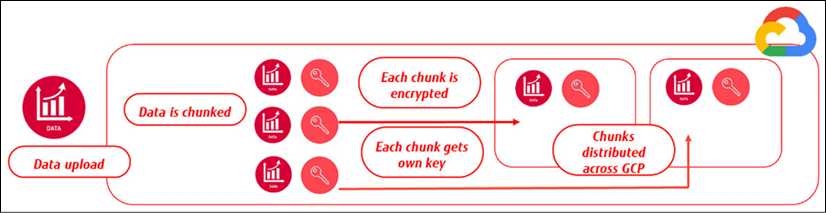
*Figure 16.6: Concept of storing CMKs in AWS KMS*

Both Azure and AWS have automated a lot in terms of encryption. They use different names for the key services, but the main principles are quite similar. That counts for GCP too, which is discussed in the next section.

**Using encryption and keys in GCP**

In GCP, all data that is stored in Cloud Storage is encrypted by default. Just like Azure and AWS, GCP offers options to manage keys. These can be supplied and managed by Google or by the customer. Keys are stored in **Cloud Key Management Service**. If the customer chooses to supply and/or manage keys themselves, these will act as an added layer on top of the standard encryption that GCP provides. That is also valid in the case of client-side encryption – the data is sent to GCP in an encrypted format, but GCP will still execute its own encryption process, as with server-side encryption. GCP Cloud Storage encrypts data with AES-256.

The encryption process itself is similar to AWS and Azure and uses DEKs and KEKs. When a customer uploads data to GCP, the data is divided into chunks. Each of these chunks is encrypted with a DEK. These DEKs are sent to the KMS where a master key is generated. The concept is shown in the following diagram:



*Figure 16.7: Concept of data encryption in GCP*

In this book, we also look at OCI and Alibaba Cloud, hence we will touch on encrypting data in these clouds next.

**Implementing encryption in OCI and Alibaba Cloud**

Like in the named major public clouds, OCI offers services to encrypt data at rest using an encryption key that is stored in OCI’s Key Management Service. Companies can use their own keys too, but these must be compliant with the **Key Management Interoperability Protocol** (**KMIP**). The process for encryption is like the other clouds. First, a master encryption key is created that is used to encrypt and decrypt all data encryption keys. Remember, we talked about these KEKs and DEKs before. Keys can be created from the OCI console, the command-line interface, or the SDK.

Data can be encrypted and stored in block storage volumes, object storage, or in databases. Obviously, OCI also recommends the rotation of keys as best practice.

Keys that are generated by customers themselves are kept in the OCI Vault and stored across a resilient cluster.

Storing keys in vaults is done in a **Hardware Security Module** (**HSM**), a physical device that holds and manages secrets such as encryption keys. Most HSMs are compliant with **FIPS** (**Federal Information Processing Standard**) Publication 140-2 Level 3 – a standard that is issued by the American government to safeguard high standards for encrypting data. The standard includes tamper-evident mechanisms, hardening, and automatic deletion of plaintext keys.

The steps are the same for Alibaba Cloud. From the console or CLI, keys are created and then encrypted data is stored in one of the storage services that Alibaba Cloud provides: OSS, Apsara File Storage, or databases.

So far, we have been looking at data itself, the storage of data, the encryption of that data, and securing access to data. One of the tasks of an architect is to translate this into principles. This is a task that needs to be performed together with the CIO or the **chief information security officer** (**CISO**). The minimal principles to be set are as follows:

* Encrypt all data in transit (end-to-end)
* Encrypt all business-critical or sensitive data at rest
* Apply DLP and have a matrix that shows clearly what critical and sensitive data is and to what extent it needs to be protected

In the *Further reading* section, some good articles are listed on encryption in the cloud and best practices for securing data.

Finally, develop use cases and test the data protection scenarios. After creating the data model, defining the DLP matrix, and applying the data protection controls, an organization has to test whether a user can create and upload data and what other authorized users can do with that data—read, modify, or delete it. That does not only apply to users but also to data usage in other systems and applications. Data integration tests are therefore also a must.

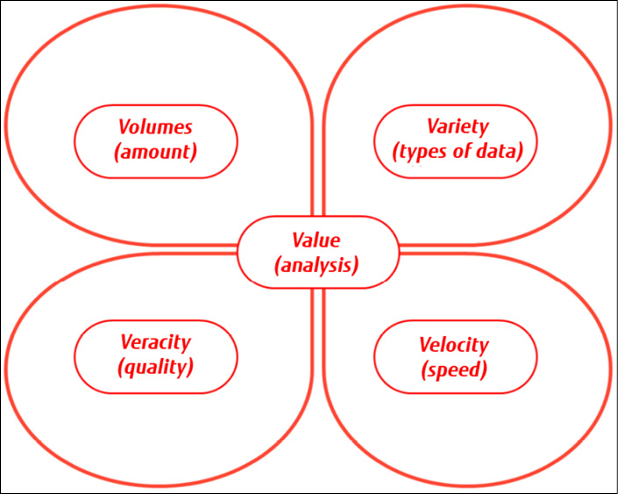
Data policies and encryption are important, but one thing should not be neglected: encryption does not protect companies from misplaced **identity and access** **management** (**IAM**) credentials. Thinking that data is fully protected because it’s encrypted and stored safely gives a false sense of security. Security really starts with authentication and authorization.

**Securing raw data for big data modeling**

One of the big advantages of the public cloud is the huge capacity that these platforms offer. Together with the increasing popularity of public clouds, the industry saw another major development in the possibilities to gather and analyze vast amounts of data, without the need to build infrastructure in on-premises data centers to host the data. With public clouds, companies can have enormous data lakes at their disposal. Data analysts program their analytical models to these data lakes. This is what is referred to as **big data**. Big data modeling is about four Vs:

* **Volume**: The quantity of data
* **Variety**: The different types of data
* **Veracity**: The quality of data
* **Velocity**: The speed of processing data

Data analysts often add a fifth V to these four, and that’s **value**. The original model does not have it, but many analysts feel that value is likely the most important driver. Big data gets value when data is analyzed and processed in such a way that it actually means something. The four-Vs model is shown in the following diagram:



*Figure 16.8: The four Vs of big data*

Processing and enriching the data is something that is done in the data modeling stage. Cloud providers offer a variety of solutions for data mining and data analytics: Azure Synapse Analytics, Redshift in AWS, BigQuery in GCP, Oracle’s BigData, and AnalyticDB in Alibaba Cloud. These solutions require a different view of data security.

As with all data, the encryption of data at rest is required and, in almost every case, is enabled by default on any big data platform or data warehouse solution that is scalable up to petabytes. Examples are Azure Data Lake, AWS Redshift, and Google’s BigQuery. These solutions are designed to hold any kind of unstructured data in one single repository.

To use Azure Data Lake as an example, as soon as the user sets up an account in Data Lake, Azure encryption is turned on and the keys are managed by Azure, although there’s an option for companies to manage keys themselves. In Data Lake, the user will have three different keys: the master encryption key, the data encryption key, and the block encryption key. The latter is necessary since data in Data Lake is divided into blocks.

AWS Redshift works in clusters to store data. These clusters can be encrypted so that all data created by users in tables is encrypted. This data can be extracted to SQL database clients. In that case, data in transit is encrypted using **Secure Sockets Layer** (**SSL**). Finally, the Redshift cluster will sit in a **virtual private cloud** (**VPC**) in AWS; access to the environment is controlled at the VPC level.

Google’s BigQuery is a fully managed service, yet users have a ton of choices for how to treat data in BigQuery. The service comprises over 100 predefined detectors to scan and identify data. Next, GCP offers a variety of tools to execute DLP policies, such as data masking and the pseudonymization of data. Scanning data in BigQuery is easy through the GCP cloud console. This is also the place where the user can enable the DLP API. As with all data in GCP, it will be encrypted upon entry by default. BigQuery doesn’t check whether data is already encrypted; it runs the encryption process at all times.

Data entered in OCI is encrypted by default. This is true for all data services in OCI: block, object, or file services storage, or in one of Oracle’s platform solutions such as Oracle Analytics Cloud service. In all cases, data encryption at rest is turned on by default. This includes encryption of the database backups. OCI automatically manages the keys, although customers can choose to use their own keys.

For encryption at rest, OCI provides the option to encrypt data stored in Object Storage using the encryption feature of this service. This feature allows for encrypting data using a customer-managed encryption key or a key managed by OCI. For encryption in transit, OCI provides the option to encrypt data transmitted over the network using the Oracle Cloud Infrastructure Networking Transport Encryption feature. This feature uses the TLS protocol. Note that encryption options for big data services may be different from the options available for other OCI services.

To store large amounts of unstructured, raw data in Alibaba Cloud, the use of OSS is recommended. Data analytics can be performed using DataWorks, which allows for data cleansing and then loading the data into data warehouses for further analysis. Again, data entered in OSS is not encrypted by default. For encryption at rest, the **Server-Side Encryption (SSE)**feature in OSS must be used. SSE encrypts data with AES-256. As in other clouds, customers can use their own keys with SSE-C. Alibaba Cloud utilizes the TLS protocol for data in transit.

**Summary**

This chapter was about securing and protecting data in cloud environments. We have learned that when moving data from on-premises systems to the cloud, companies have to set specific controls to protect their data. The owner of the data remains responsible for protecting the data; that doesn’t shift to the cloud provider.

We have learned that companies need to think first about data protection policies. What data needs to be protected? Which laws and international frameworks are applicable to be compliant? A best practice is to start thinking about the data model and then draw a matrix, showing what the policy should be for critical and sensitive data. We’ve also studied the principles of DLP using data classification and labeling.

This chapter also explored the different options a company has to store data in cloud environments and how we can protect data from a technological point of view. After finishing this chapter, you should have a good understanding of how encryption works and how Azure, AWS, GCP, OCI, and Alibaba Cloud treat data and the encryption of data. Lastly, we’ve looked at the big data solutions on the cloud platforms and how raw data is protected.

The next chapter is the final one about security operations in multi-cloud. Cloud providers offer native security monitoring solutions, but how can enterprises monitor security in multi-cloud? The next chapter will discuss **integrated security monitoring** in the cloud.

**Questions**

1. To define the risk of data loss, businesses are advised to conduct an assessment. Please name this assessment methodology.
2. In this chapter, we’ve studied encryption. Please name two encryption technologies that are commonly used in cloud environments.
3. What’s the service in AWS to manage encryption keys?
4. In Azure, companies keep keys in Azure Key Vault. True or false: a key vault is hosted in the same subnet as the storage itself.

**Further reading**

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

* Information about the management of storage keys in Azure: <https://learn.microsoft.com/en-gb/azure/security/fundamentals/data-encryption-best-practices>
* Encryption overview in Azure: <https://docs.microsoft.com/en-us/azure/security/fundamentals/encryption-overview>
* Data protection in AWS: <https://docs.aws.amazon.com/AmazonS3/latest/dev/UsingEncryption.html>
* Encryption options in GCP: <https://cloud.google.com/storage/docs/encryption>
* Blog by Kenneth Hui on Cloud Architect Musings about encryption in Azure, AWS, and GCP: <https://cloudarchitectmusings.com/2018/03/09/data-encryption-in-the-cloud-part-4-aws-azure-and-google-cloud/>
* Information about encryption in OCI: <https://blogs.oracle.com/cloudsecurity/post/how-oci-helps-you-protect-data-with-default-encryption#:~:text=Data%20at%2Drest%20encryption,is%20turned%20on%20by%20default>
* Information about encryption in Alibaba Cloud: <https://www.alibabacloud.com/blog/data-encryption-at-storage-on-alibaba-cloud_594581#:~:text=Your%20data%20and%20the%20associated,FIPS)%20140%2D2%20standard>