Navigating LGPD Compliance on AWS

March 2020

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Contents

Brazilian General Data Protection Overview 1

Changes the LGPD Introduces to Organizations Operating in Brazil 1

AWS Preparation for the LGPD 2

Role of AWS under the LGPD 2

Shared Security Responsibility Model 2

Strong Compliance Framework and Security Standards 3

AWS Compliance Program 3

Data Access Controls 4

AWS Identity and Access Management 5

Multi-Factor Authentication 6

Access to AWS Resources 7

Access to Operational & Configuration Data 8

Geo-Restrictions 9

Control Access to Web Applications and Mobile Apps 9

Monitoring and Logging 10

Manage and Configure Assets with AWS Config 10

Compliance Auditing & Security Analytics with AWS CloudTrail 11

Other Logging Features 13

Centralized Security Management 13

Protecting your Data on AWS 15

Encrypt Data at Rest 15

Encrypt Data in Transit 16

Encryption Tools 17

Data Protection by Design & by Default 22

Contributors 22

Document Revisions 23

Abstract

This document provides information about services and resources that Amazon Web Services (AWS) offers customers to help them align with the requirements of the Brazilian General Data Protection Law (LGPD) that might apply to their activities. These include adherence to IT security standards, data access controls, monitoring and logging tools, encryption, and key management.

Brazilian General Data Protection Overview

The Brazilian General Data Protection Law (Law No. 13,709 of August 14, 2018, as amended by Law No. 13,853 of July 8, 2019) or LGPD is Brazil's first extensive data protection regulation and is largely aligned to the European Union's General Data Protection Regulation (GDPR). The LGPD will take effect in August 2020. The LGPD applies to any processing operation of *personal data* (defined as information related to an identified or identifiable natural person) carried out by individuals or legal entities from the public or private sector, irrespective of the means used for the processing or the country where the controller or the data is located, provided that: 1) the processing operation is carried out in Brazil, 2) the purpose of the processing activity is to offer or provide goods or services to individuals in Brazil, or 3) the personal data was collected in Brazil.

The LGPD established a data protection agency, the National Data Protection Authority (ANPD), which oversees the protection of personal data and issue regulations and procedures related to personal data protection. As of the date of issue of this document, the members of the ANPD have not yet been appointed.

Changes the LGPD Introduces to Organizations Operating in Brazil

The LGPD significantly transformed the data protection system in Brazil by establishing rules for the collection, use, processing, and storage of personal data. Organizations must be able to demonstrate on a continual basis the security of the data they are processing and their compliance with the LGPD by implementing and regularly reviewing robust technical and organizational measures. This requires the establishment and enforcement of compliant policies applicable to the processing of personal data. Those who commit violations under the LGPD may be subject to a range of penalties, including: warnings; suspension or the blocking of processing activities that violate the law; and fines up to 2% of violators gross revenue in Brazil in the previous year, which are limited to R$50 Million.

Under the LGPD, controllers and processors (as defined under the LGPD) are required to adopt security measures, both technical and administrative, to protect personal data from unauthorized accesses, accidental or unlawful situations of destruction, loss, alteration, communication, or any type of improper or unlawful processing. Additionally, the LGPD grants the ANPD authority to establish minimum technical standards to be

implemented by controllers and processors. At the time of this writing, ANPD has not yet issued these minimum technical standards.

AWS Preparation for the LGPD

AWS Compliance, Data Protection, and Security experts have been working with customers across the world to address their questions and help them prepare for running workloads in the cloud after the LGPD comes into effect. These teams are also reviewing the operations and responsibilities of AWS against the requirements of the LGPD to ensure that AWS services can be used in compliance with the LGPD when the law takes effect.

Role of AWS under the LGPD

Under the LGPD, AWS can act both as a *data controller* and a *data processor*. A *data controller* is defined under the LGPD as the natural or legal person, whether public or private, who is responsible for decisions concerning the processing of personal data. A *data processor* is defined in the LGPD as the natural or legal person, whether public or private, who performs the processing of personal data on behalf of the controller.

AWS as a Data Controller

When AWS collects personal data and determines the purposes and means of processing that personal data – for example, when AWS collects and stores information from its direct customers for account registration, administration, services access, service attributes, or contact information for the AWS account to provide assistance through customer support activities – it acts as a data controller.

AWS as a Data Processor

When customers and AWS Solution Providers use AWS services to process personal data in their customer content, AWS acts as a data processor. Customers and AWS Solution Providers can use the controls available in AWS services, including security configuration controls, to process and store personal data. Under these circumstances, the customer or APN Partner may act as a data controller or data processor itself, and AWS acts as a data processor or sub-processor.

Shared Security Responsibility Model

Security and Compliance is a shared responsibility between AWS and the customer. When customers move their computer systems and data to the cloud, privacy and

security responsibilities are shared between the customer and the cloud service provider. When customers move to the AWS Cloud, AWS is responsible for securing the underlying infrastructure that supports the cloud, and customers are responsible for anything they put in the cloud or connect to the cloud. This differentiation of responsibility is commonly referred to as security *of* the cloud versus security *in* the cloud.

This shared model can help reduce customers' operational burden, and provide them with the necessary flexibility and control to deploy their infrastructure in the AWS Cloud. AWS operates, manages, and controls the infrastructure components, from the host operating system and virtualization layer, to the implementation of abstracted services, to the physical security of the facilities in which the service operates. Customers assume responsibility and management of the guest operating system (including updates and security patches), other associated application software, and the configuration of the security group firewall provided by AWS. For more information, see AWS Shared Responsibility Model.

Strong Compliance Framework and Security Standards

The LGPD grants the ANPD authority to establish minimum technical standards to be implemented by data controllers or data processors. As of the date of this document, the ANPD has not established minimum technical standards, but AWS already offers customers a strong compliance framework and advanced security capabilities that meet the needs of modern security and compliance standards around the globe.

AWS Compliance Program

The AWS Compliance Program helps customers to understand the robust controls in place at AWS to maintain security and data protection in the cloud. As systems are built on top of AWS cloud infrastructure, compliance responsibilities will be shared. By tying together governance-focused, audit-friendly service features with applicable compliance or audit standards, AWS Compliance enablers build on traditional programs; helping customers to establish and operate in an AWS security control environment. The IT infrastructure that AWS provides to its customers is designed and managed in alignment with security best practices and a variety of IT security standards, including:

- SOC 1/SSAE 16/ISAE 3402 (formerly SAS 70)

- SOC 2
- SOC 3
- FISMA, DIACAP, and FedRAMP (USA)
- DoD SRG (USA)
- IRAP (Australia)
- MTCS Tier 3 (Singapore)
- C5 (Germany)
- ENS High (Spain)
- PCI DSS Level 1
- ISO 9001
- ISO/IEC 27001 / ABNT NBR ISO/IEC 27001
- ISO/IEC 27017 / ABNT NBR ISO/IEC 27017
- ISO/IEC 27018 / ABNT NBR ISO/IEC 27018[1]
- FIPS 140-2 (USA and Canada)

In addition, the flexibility and control that the AWS infrastructure provides allows customers to deploy solutions that meet many industry-specific standards.

AWS provides a wide range of information regarding its IT control environment to customers through whitepapers, reports, certifications, accreditations, and other third-party attestations. More information is available in the Risk and Compliance Whitepaper.

Data Access Controls

The LGPD states that controllers and processors must adopt security measures, both technical and administrative, to protect personal data from unauthorized access. The following AWS access control mechanisms can help customers comply with this

[1]As of the publication date of this whitepaper, AWS has applied for, but not yet received official certification under the Brazilian versions of these standards.

requirement by allowing only authorized administrators, users and applications access to AWS resources and customer data:

AWS Identity and Access Management

When you create an AWS account, a root user principal is automatically created for your AWS account. This principal has complete access to all your AWS services and resources in your AWS account. Instead of using this root principal for everyday tasks, you should only use it to initially create additional roles and user accounts, and for a small number of administrative activities that require it. AWS recommends that you apply the principle of least privilege from the start: define different user accounts and roles for different tasks, and specify the minimum set of permissions required to complete each task. AWS Identity and Access Management (IAM) is a web services that you can use to securely control access to your AWS resources.

Users and roles define IAM identities with specific permissions. IAM Users are directly provisioned in AWS and provide a rich set of features and capabilities. With IAM Roles, you can allow users to perform specific tasks to assume the role and leverage temporary credentials for the role session. You can use IAM roles to access your account using existing user identities via SAML or OIDC Federation or the AWS Single-Sign-On (SSO) service. You can also use Roles to securely provide temporary credentials to applications that run in Amazon Elastic Compute Cloud (Amazon EC2), Elastic Container Service (ECS), or AWS Lambda so those applications can access to other AWS resources, such as Amazon Simple Storage Service (Amazon S3 or Amazon S3 buckets), and Amazon RDS or DynamoDB databases.

Temporary Access Tokens Through AWS STS

You can use the AWS Security Token Service (AWS STS) to create and provide trusted users with temporary security credentials that grant access to your AWS resources. Temporary security credentials work almost identically to the long-term credentials that you can provide for your IAM users, with the following differences:

- Temporary security credentials are for short-term use. You can configure the amount of time that they are valid, from a few minutes to several hours. After temporary credentials expire, AWS does not recognize them or allow any kind of access from API requests made with them.

- Temporary security credentials are not stored with the user account. Instead, they are generated dynamically and provided to the user when requested. When (or before) temporary security credentials expire, a user can request new credentials, if that user has permissions to do so.

These differences provide the following advantages when you use temporary credentials:

- You do not have to distribute or embed long-term AWS security credentials with an application.
- Temporary credentials are the basis for Roles and identity federation. You can provide access to your AWS resources to users by defining a temporary AWS identity for them.
- Temporary security credentials have a limited customizable lifespan. Because of this, you do not have to rotate them or explicitly revoke them when they're no longer needed. After temporary security credentials expire, they cannot be reused. You can specify the maximum amount of time the credentials are valid.

Multi-Factor Authentication

For extra security, you can add two-factor authentication to the root principal of your account and to individual IAM User accounts. With multi-factor authentication (MFA) enabled, when you sign into an AWS website, you are prompted for your user name and password (the first factor), as well as an authentication response from your AWS MFA device (the second factor). You can enable MFA for your AWS account and for individual IAM users you have created in your account. You can also use MFA to control access to AWS service APIs.

For example, you can define a policy that allows full access to all AWS API operations in Amazon EC2, but explicitly denies access to specific API operations—such as *StopInstances* and *TerminateInstances*—if the user is not authenticated with MFA.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowAllActionsForEC2",
            "Effect": "Allow",
            "Action": "ec2:*",
            "Resource": "*"
        },
        {
            "Sid": "DenyStopAndTerminateWhenMFAIsNotPresent",
            "Effect": "Deny",
            "Action": [
                "ec2:StopInstances",
                "ec2:TerminateInstances"
            ],
            "Resource": "*",
            "Condition": {
                "BoolIfExists": {"aws:MultiFactorAuthPresent": false}
            }
        }
    ]
}
```

In the case where your users access their AWS resources via federation, you can obtain similar results. In that case multi-factor authentication would be used to authenticate to the identity provider, and then claims from the identity provider can be turned into Principal Tags in IAM and evaluated in the authorization context.

Access to AWS Resources

To implement granular access to your AWS objects, you can grant different levels of permissions to different people for different resources. For example, you can allow only some users to have complete access to Amazon EC2, Amazon S3, Amazon DynamoDB, Amazon Redshift, and other AWS services.

For other users, you can allow read-only access to only some Amazon S3 buckets, permission to administer only some Amazon EC2 instances, or to access only your billing information.

The following policy is an example of one method you can use to allow all actions on a specific Amazon S3 bucket and explicitly deny access to every AWS service that is not Amazon S3.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "s3:*",
            "Resource": [
                "arn:aws:s3:::bucket-name",
                "arn:aws:s3:::bucket-name/*"
            ]
        },
        {
            "Effect": "Deny",
            "NotAction": "s3:*",
            "NotResource": [
                "arn:aws:s3:::bucket-name",
                "arn:aws:s3:::bucket-name/*"
            ]
        }
    ]
}
```

You can attach a policy to a user account or to a role. For other examples of IAM policies, see Example IAM Identity-Based Policies.

Access to Operational & Configuration Data

You can use AWS Systems Manager to see and manage the operations of your AWS infrastructure. You can audit and enforce compliance to defined states. AWS Systems Manager Parameter Store can centrally manage data defining parameters. This enables you to implement granular access to parameter data, whether it is plain-text data (such as database strings) or secrets (such as passwords). You can provide this access control through customized permissions to users and resources (such as instances) for parameter access and to use the integration with IAM. For example, in a development environment, credentials are often hardcoded. Instead of hardcoding your credentials, you can use Parameter Store to save passwords and allow your developers to get access to the credentials with the AWS API get-parameter .

The following API snippet example shows the password retrieval *get-parameter*:

```
Password=$(aws ssm get-parameters --region us-east-1 --names MySecureSQLPassword
```

Another available option for protecting secrets needed to access your applications, services, and IT resources is AWS Secrets Manager. The service enables you to easily rotate, manage, and retrieve database credentials, API keys, and other secrets

throughout their lifecycle. Users and applications retrieve secrets with a call to Secrets Manager APIs, eliminating the need to hardcode sensitive information in plain text.

Secrets Manager offers secret rotation with built-in integration for Amazon RDS, Amazon Redshift, and Amazon DocumentDB.

Geo-Restrictions

You can use geo-restrictions—also known as geoblocking—to prevent users in specific geographic locations from accessing content that you're distributing through an Amazon CloudFront web distribution.

There are two options for using geo-restrictions:

- **CloudFront geo-restriction feature** – Customers may select this option to restrict access to all of the files that are associated with a CloudFront distribution, and to restrict access at the country level.
- **Third-party geolocation service** – Customers may select this option to restrict access to a subset of the files that are associated with a distribution, or to restrict access at a finer level of granularity than the country level.

Beyond these two options, geo-limiting capabilities exist for newly launched Regions. While AWS Regions introduced before March 20, 2019 are enabled by default. Regions introduced after March 20, 2019, such as Asia Pacific (Hong Kong) and Middle East (Bahrain), are disabled by default. You must enable these Regions before you can use them. If an AWS Region is disabled by default, you can use the AWS Management Console to enable and disable the Region. Enabling and disabling AWS Regions allows you to control whether users in your AWS account can access resources in that Region.

Control Access to Web Applications and Mobile Apps

AWS provides service for managing data access control within their applications. If you need to add user login and access control features to your web applications and mobile apps, you can use Amazon Cognito. Amazon Cognito User Pools provide a secure user directory that scales to hundreds of millions of users. To protect the identity of the users, you can add multi-factor authentication (MFA) to your user pools. You can also use adaptive authentication, which uses a risk-based model to predict when you might need another authentication factor.

With Amazon Cognito, you can see who accessed your resources and where the access originated (mobile app or web application). You can use this information to

create security policies that allow or deny access to a resource based on the type of access origin (mobile app or web application).

Monitoring and Logging

Monitoring and logging are key elements of a robust security architecture, and AWS offers a number of monitoring and logging services and features, including the following:

Manage and Configure Assets with AWS Config

AWS Config provides a detailed view of the present configuration as well as the history of the AWS resources in your AWS account. This includes how the resources are related to one another, and how they were previously configured, so that you can see how the configurations and relationships change over time.

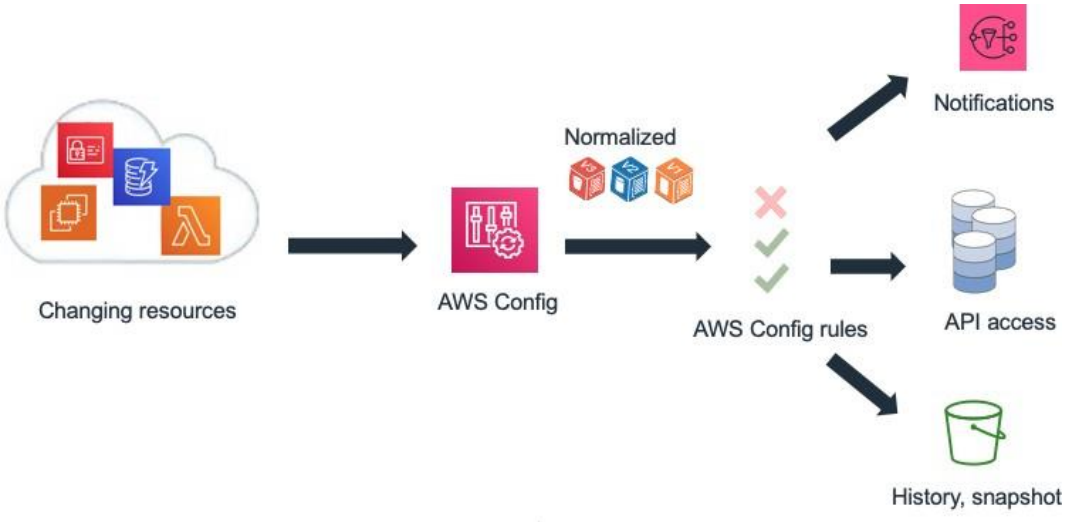

Figure 1 – Monitor configuration changes over time with AWS Config

An AWS resource is an entity that you can work within AWS, such as an EC2 instance, an Amazon Elastic Block Store (Amazon EBS) volume, a security group, or an Amazon Virtual Private Cloud (Amazon VPC). For a complete list of AWS resources supported by AWS Config, see Supported AWS Resource Types.

With AWS Config, you can do the following:

- Evaluate your AWS resource configurations to verify the settings are correct;
- Get a snapshot of the current configurations of the supported resources that are associated with your AWS account;

- Get configurations of one or more resources that exist in your account;
- Get historical configurations of one or more resources;
- Get a notification when a resource is created, modified, or deleted; and
- See relationships between resources. For example, you might want to find all resources that use a particular security group.
- Automatically run code via Config Rules in response to any changes in configuration, either notifying an administrator or even automatically remediating any unwanted changes.

Compliance Auditing & Security Analytics with AWS CloudTrail

With AWS CloudTrail, you can continuously monitor AWS Account activity. A history of the AWS API calls for your account is captured, including API calls made through the AWS Management Console, the AWS SDKs, the command line tools, and higher-level AWS services. You can identify which users and accounts called AWS APIs for services that support CloudTrail, the source IP address the calls were made from, and when the calls occurred. You can integrate CloudTrail into applications using the API, automate trail creation for your organization, check the status of your trails, and control how administrators enable and disable CloudTrail logging. You can organize and store CloudTrail logs in an Amazon S3 bucket for auditing purposes or for troubleshooting activities.

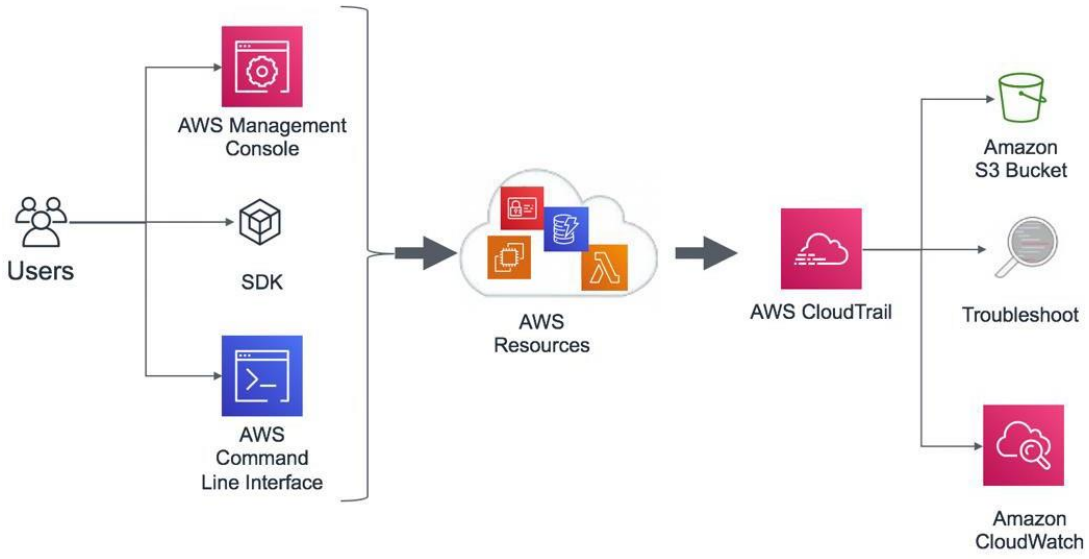

Figure 2 – Example architecture for compliance auditing and security analytics with AWS CloudTrail

AWS CloudTrail logs can also trigger preconfigured Amazon CloudWatch events. You can use these events to notify users or systems that an event has occurred, or for remediation actions. For example, if you want to monitor activities on your Amazon EC2 instances, you can create a CloudWatch Event rule. When a specific activity happens with respect to an Amazon EC2 instance and the event is captured in the logs, the rule triggers an AWS Lambda function, which sends a notification email about the event (when it happened, which user performed the action, Amazon EC2 details, etc.) to the administrator. The following diagram shows the architecture of the event notification.

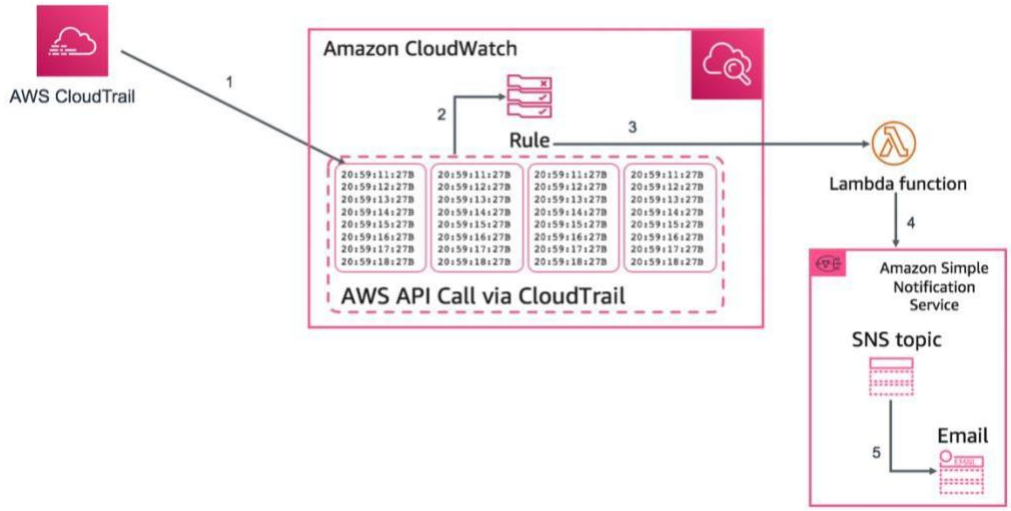

Figure 3 – Example of AWS CloudTrail event notification

Other Logging Features

In addition to CloudTrail API logging, there are a number of other important log sources and types that enable you to keep your AWS environment secure. For example, when you enable S3 logging, you can get detailed access logs for the requests that are made to your Amazon S3 bucket. An access log record contains details about the request, such as the request type, the resources specified in the request, and the time and date the request was processed. For more information about the contents of a log message, see Amazon S3 Server Access Log Format in the *Amazon Simple Storage Service Developer Guide*.

Other kinds of important log sources beyond CloudTrail and S3 include:

- Detailed information about all the network flows in your virtual network through VPC-Flow Logs.
- Request logs from your Elastic Load Balancers.
- Filtering and monitoring of HTTP access to applications with WAF functions in CloudFront.
- Operating system logs centrally gathered and analyzable using CloudWatch Logs and the EC2 logging agent.

Logs are also a useful source of information for threat detection. The Amazon GuardDuty thread detection service analyzes logs from AWS CloudTrail, VPC Flow Logs, and AWS DNS, which enables you to continuously monitor your AWS accounts and workloads. This service uses machine learning, threat intelligence, and anomaly detection to deliver detailed and actionable alerts any time a malicious activity or an unauthorized behavior is recorded.

Centralized Security Management

Many organizations have challenges related to visibility and centralized management of their environments. As your operational footprint grows, this challenge can be compounded unless you carefully consider your security designs. Lack of knowledge, with decentralized and uneven management of governance and security processes, can make your environment vulnerable.

AWS provides tools that help you to address some of the most challenging requirements for IT management and governance, and tools for supporting a data protection by design approach.

AWS Organizations helps you centrally manage and govern very complex environments. It enables you to control access, compliance, and security in a multi-account environment. AWS Organizations supports the Service Control Policy (SCP), which defines the AWS service actions available to use with different accounts in an organization.

AWS Control Tower provides an easy method to set up and govern a new, secure, multi-account AWS environment. It automates the setup of a landing zone which is a multi-account environment that is based on best-practices blueprints, and enables governance using guardrails that you can choose from a pre-packaged list. Guardrails implement governance rules for security, compliance, and operations. AWS Control Tower provides identity management using AWS Single Sign-On (SSO) default directory and enables cross-account audit using AWS SSO and AWS IAM. It also centralizes logs coming from Amazon CloudTrail and AWS Config logs, which are stored in Amazon S3.

AWS Security Hub is another service that supports centralization and can improve visibility into an organization. Security Hub centralizes and prioritizes security and compliance findings from across AWS accounts and services, and can be integrated with security software from third-party partners to help you analyze security trends and identify the highest priority security issues.

Amazon CloudWatch Events enables you to set up your AWS account to send events to other AWS accounts, or become a receiver for events from other accounts or organizations. This mechanism can be very useful for implementing cross-account incident response scenarios, by taking timely corrective actions – for example, by calling a Lambda function, or running a command on EC2 instance – as necessary any time a security incident event occurs.

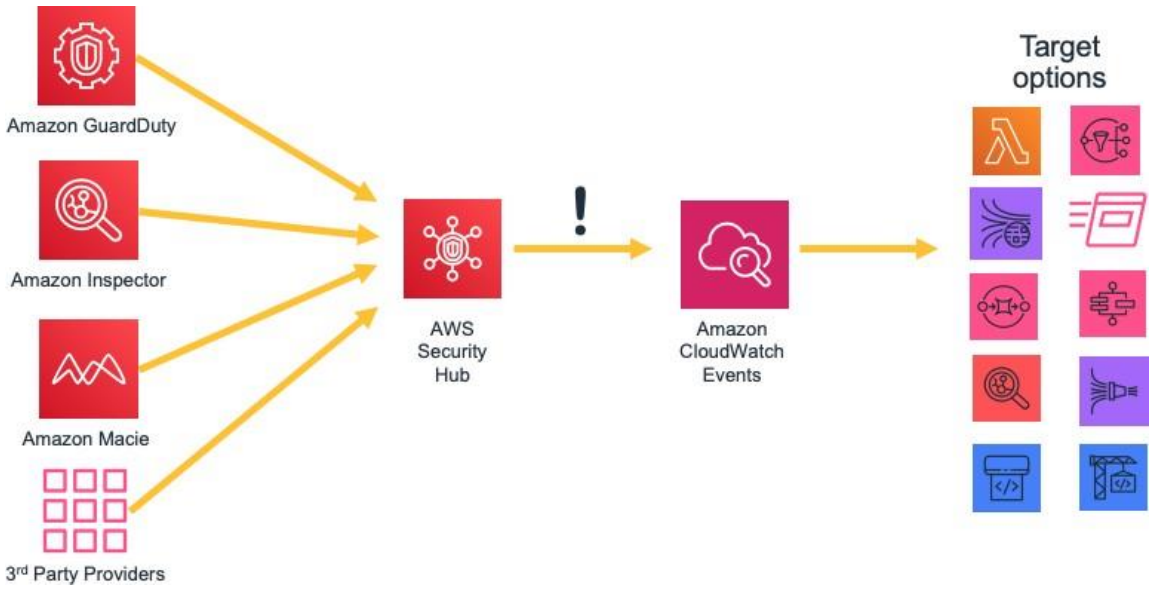

Figure 4 – Taking action with AWS Security Hub and Amazon CloudWatch Events

Protecting your Data on AWS

The LGPD requires that businesses adopt technical and administrative security measures capable of protecting personal data from unauthorized access and accidental or unlawful situations of destruction, loss, change, communication, dissemination, or any form of improper or unlawful treatment. To that end, AWS offers various highly scalable and secure data encryption mechanisms to help protect customer data stored and processed on AWS.

Encryption reduces the risks associated with the storage of personal data because data is unreadable without the correct key. A thorough encryption strategy can help mitigate the impact of various security events, including certain kinds of security breaches.

Encrypt Data at Rest

Encrypting data at rest is vital for regulatory compliance and data protection. It helps to ensure that sensitive data saved to persistent storage is not readable by any user or application without proper authorization including a valid key. AWS provides multiple options for encryption at rest and encryption key management. For example, you can use the AWS Encryption SDK with a customer master key (CMK) created and managed in AWS Key Management Service (AWS KMS) to encrypt arbitrary data. Many AWS services also provide automatic encryption at rest while still allowing the customer to control, rotate, or revoke the master encryption keys in AWS KMS.

Encrypted data can be securely stored at rest and can be decrypted only by a party with authorized access to the CMK. As a result, you get confidential envelope-encrypted data, policy mechanisms for authorization and authenticated encryption, and audit logging through AWS CloudTrail. As noted, most AWS storage and database services have built-in encryption at rest features, providing the option to encrypt data before it is written to non-volatile storage. For example, you can configure your account to automatically encrypt all Amazon Elastic Block Store (Amazon EBS) volumes using data keys protected by a master key in KMS. You can also configure Amazon S3 buckets for server-side encryption (SSE) using AES-256 encryption. Amazon Relational Database Service (Amazon RDS) also supports Transparent Data Encryption (TDE).

Another method for encrypting data on EC2 storage volumes is using built-in Linux or Windows libraries. These methods encrypts files transparently, which protects confidential data. As a result, applications that process the data are unaware of the disk-level encryption.

You can use two methods to encrypt files on instance stores. The first method is *disk encryption*, in which the entire disk, or block within the disk, is encrypted using one or more encryption keys. Disk encryption operates below the file-system level, is operating-system agnostic, and hides directory and file information, such as name and size. Encrypting File System, for example, is a Microsoft extension to the Windows operating system's New Technology File System (NTFS) that provides disk encryption.

The second method is *file-system-level encryption*. With this method, files and directories are encrypted, but not the entire disk or partition. File-system-level encryption operates on top of the file system and is portable across operating systems.

For non-volatile memory express (NVMe) SSD instance store volumes in EC2, hardware-accelerated encryption is always enabled. Data in an NVMe instance storage is encrypted using an XTS-AES-256 block cipher implemented in a hardware module called a Nitro controller on the instance. The encryption keys are generated using the hardware module and are unique to each NVMe instance storage device. All encryption keys are destroyed when the instance is stopped or terminated and cannot be recovered. Unlike with EBS volumes, you cannot in this case use your own encryption keys.

Encrypt Data in Transit

AWS strongly recommends encrypting data in transit from one system to another, including resources within and outside of AWS.

When you create an AWS account and utilize an EC2 virtual machine service resource, a logically isolated section of the AWS Cloud is provisioned to it, the Amazon Virtual Private Cloud (Amazon VPC). There you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment, including selecting your own IP address range, creation of subnets, and configuration of route tables and network gateways. You can also create a IPSec encrypted Virtual Private Network (VPN) connection between your corporate datacenter and your Amazon VPC, so you can use the AWS Cloud as an extension of your corporate datacenter.

For protecting communication between your Amazon VPC and your corporate datacenter, you can select from several VPN connectivity options, and choose one that best matches your needs. You can use the AWS Client VPN to enable secure access to your AWS resources using client-based VPN services. You can also use a third-party software VPN appliance, which you can install on an Amazon EC2 instance in your Amazon VPC. Or, you can create an IPsec VPN connection to protect the communication between your VPC and your remote network. To create a dedicated private connection from a remote network to your Amazon VPC, you can use AWS Direct Connect. You can combine this connection with an AWS Site-to-Site VPN to create an IPsec-encrypted connection.

AWS provides HTTPS endpoints using the TLS (Transport Layer Security) protocol for communication, which provides encryption in transit when you use AWS APIs. You can use the AWS Certificate Manager (ACM) service to generate, manage, and deploy the private and public certificates you use to establish encrypted transport between systems for your workloads. A number of AWS services that support TLS, such as Amazon Elastic Load Balancing, are integrated with ACM and to provide public or private X.509 certificates and, in some cases, to automatically rotate certificates on your behalf. If your content is distributed through Amazon CloudFront, it also supports encrypted endpoints and ACM-managed certificates.

Encryption Tools

AWS offers various highly scalable data encryption services, tools, and mechanisms to help protect your data stored and processed on AWS. For information about AWS Service functionality and privacy, see AWS Service Capabilities for Privacy Considerations.

Cryptographic services from AWS use a wide range of encryption and storage technologies that are designed to maintain integrity of your data at rest or in transit.

AWS offers four primary tools for cryptographic operations.

- **AWS Key Management Service (AWS KMS)** is an AWS managed service that generates and manages both master keys and data keys. AWS KMS is integrated with many AWS services to provide server-side encryption of data using KMS keys from customer accounts. KMS hardware security modules (HSMs) are FIPS 140-2 Level 2 validated.
- **AWS CloudHSM** provides HSMs that are FIPS 140-2 Level 3 validated. They securely store a variety of your self-managed cryptographic keys, including master keys and data keys.
- **AWS Cryptographic Services and Tools**
    - **AWS Encryption SDK** provides a client-side encryption library for implementing encryption and decryption operations on *all* types of data.
    - **Amazon DynamoDB Encryption Client** provides a client-side encryption library for encrypting data tables before sending them to a database service, such as Amazon DynamoDB.

AWS Key Management Service

AWS Key Management Service (AWS KMS) is a managed service that makes it easy for you to create and control the encryption keys used to encrypt your data, and uses FIPS-validated Hardware Security Modules (HSMs) to protect the security of your keys. AWS KMS is integrated with dozens of other AWS services to help you protect the data you store with these services. AWS KMS is also integrated with AWS CloudTrail to provide you with logs of all your key usage for your regulatory and compliance needs.

You can easily create, import, and rotate keys, as well as define usage policies and audit usage from the AWS Management Console or by using the AWS SDK or AWS Command Line Interface (AWS CLI).

The master keys in AWS KMS, whether imported by you or created on your behalf by AWS KMS and known as customer master keys (CMKs), are stored in highly durable storage in an encrypted format to help ensure that they can be used when needed. You can choose to have AWS KMS automatically rotate CMKs created in AWS KMS once per year without having to re-encrypt data that has already been encrypted with your master key. You don’t need to keep track of older versions of your CMKs because AWS KMS keeps them available to automatically decrypt previously encrypted data.

For any CMK in KMS, you can control who has access to those keys and which services they can be used with through a number of access controls, including grants, and key policy conditions within key policies or IAM policies. You can also import keys from your own key management infrastructure and use them in KMS.

For example, the following policy uses the *kms:ViaService* condition to allow a customer managed CMK to be used for the specified actions only when the request comes from Amazon EC2 or Amazon RDS in a specific Region (*us-west-2*) on behalf of a specific user (*ExampleUser*).

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:user/ExampleUser"
  },
  "Action": [
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*",
    "kms:CreateGrant",
    "kms:ListGrants",
    "kms:DescribeKey"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:ViaService": [
        "ec2.us-west-2.amazonaws.com",
        "rds.us-west-2.amazonaws.com"
      ]
    }
  }
}
```

AWS Service Integration

AWS KMS has integrated with a number of AWS services (nearly sixty at the time of this writing). These integrations allow you to easily use AWS KMS CMKs to encrypt the data you store with these services. In addition to using a customer managed CMK, a number of the integrated services allow you to use an AWS managed CMK that is created and managed for you automatically, but is only usable within the specific service that created it.

Audit Capabilities

If AWS CloudTrail is enabled for your AWS account, each use of a key that you store in KMS is recorded in a log file that is delivered to the Amazon S3 bucket that you

specified when you enabled AWS CloudTrail. The information recorded includes details of the user, time, date, and the key used.

Security

AWS KMS is designed to make sure that no one has access to your master keys. The service is built on systems that are designed to protect your master keys with extensive hardening techniques, such as never storing plaintext master keys on disk, not persisting them in memory, and limiting which systems can access hosts that use keys. All access to update software on the service is controlled by a multi-party access control that is audited and reviewed by an independent group within Amazon.

For more information about AWS KMS, see the AWS Key Management Service whitepaper.

AWS CloudHSM

The AWS CloudHSM service helps you meet corporate, contractual, and regulatory compliance requirements for data security by using dedicated Hardware Security Module (HSM) appliances in the AWS Cloud. With CloudHSM, you control the encryption keys and cryptographic operations performed by HSM.

AWS and AWS Marketplace partners offer a variety of solutions for protecting sensitive data within the AWS infrastructure, but for applications and data subject to rigorous contractual or regulatory requirements for managing cryptographic keys, additional protection is sometimes necessary. Previously, the only option to store sensitive data (or the encryption keys protecting the sensitive data) may have been in on-premises datacenters. This might have prevented you from migrating these applications to the cloud or significantly slowed their performance. With AWS CloudHSM, you can protect your encryption keys within HSMs designed and validated to government standards for secure key management. You can securely generate, store, and manage the cryptographic keys used for data encryption to make sure that only you can get access to them. AWS CloudHSM helps you comply with strict key management requirements without sacrificing application performance.

The AWS CloudHSM service works with Amazon Virtual Private Cloud (Amazon VPC). CloudHSM instances are provisioned inside your Amazon VPC with an IP address that you specify, which provides simple and private network connectivity to your EC2 instances. When you locate your CloudHSM instances near your Amazon EC2 instances, you decrease network latency, which can improve application performance. AWS provides dedicated and exclusive (single tenant) access to CloudHSM instances, which are isolated from other AWS customers.

Available in multiple Regions and Availability Zones, CloudHSM enables you to add secure and durable key storage to your applications.

Integration with AWS Services and Third-Party Applications

You can use CloudHSM with Amazon Redshift, Amazon Relational Database Service (Amazon RDS) for Oracle, or third-party applications (such as SafeNet Virtual KeySecure) as your Root of Trust, Apache (SSL termination), or Microsoft SQL Server (transparent data encryption). You can also use CloudHSM when you write your own applications and continue to use the standard cryptographic libraries you're familiar with, including PKCS#11, Java JCA/JCE, and Microsoft CAPI and CNG.

Audit Activities

If you need to track resource changes, or audit activities for security and compliance purposes, you can review all of the CloudHSM API calls made from your account through AWS CloudTrail. Additionally, you can audit operations on the HSM appliance using syslog or send syslog log messages to your own log collector.

AWS Cryptographic Services and Tools

AWS offers mechanisms that comply with a wide range of cryptographic security standards that you can use to implement best-practice encryption. The AWS Encryption SDK is a client-side encryption library, available in Java, Python, C, JavaScript, and a command line interface that supports Linux, macOS, and Windows. The AWS Encryption SDK offers advanced data protection features including secure, authenticated, symmetric key algorithm suites, such as 256-bit AES-GCM with key derivation and signing. Because it was specifically designed for applications that use Amazon DynamoDB, the DynamoDB Encryption Client enables users to protect their table data before it is sent to the database. It also verifies and decrypts data when it is retrieved. The client is available in Java and Python.

Linux DM-Crypt Infrastructure

Dm-crypt is a Linux kernel-level encryption mechanism that allows users to mount an encrypted file system. Mounting a file system is the process in which a file system is attached to a directory (mount point), which makes it available to the operating system. After mounting, all files in the file system are available to applications without any additional interaction. These files are, however, encrypted when stored on disk.

Device mapper is an infrastructure in the Linux 2.6 and 3.x kernel that provides a generic method to create virtual layers of block devices. The device mapper crypt target provides transparent encryption of block devices using the kernel crypto API. The

solution in this post uses dm-crypt in conjunction with a disk-backed file system mapped to a logical volume by the Logical Volume Manager (LVM). LVM provides logical volume management for the Linux kernel.

Data Protection by Design & by Default

Any time a user or an application tries to use the AWS Management Console, the AWS API, or the AWS CLI, a request is sent to AWS. The AWS service receives the request and executes a set of several steps to determine whether to allow or deny the request, according to a specific policy evaluation logic. All requests on AWS are denied by default (the default *deny* policy is applied). This means that everything that is not explicitly allowed by the policy is denied. In the definition of policies and as a best practice, AWS suggests that you apply the least privilege principle, which means that every component (such as users, modules, or services) must be able to access only the resources required to complete its tasks.

AWS also provides tools to implement *infrastructure as code,* which is a powerful mechanism for including security from the beginning of the design of an architecture. AWS CloudFormation provides a common language to describe and provision all infrastructure resources, including security policies and processes. With these tools and practices, security becomes part of your code and can be versioned, monitored, and modified (with a versioning system) according to the requirements of your organization.

This enables the *data protection by design* approach, because you can include security processes and policies in the definition of your architecture, and these processes and policies can also be continuously monitored by security measures in your organization.

Contributors

Contributors to this document include:

- Stacy Shelhorse, Security Assurance TIS, Amazon Web Services
- Fernando Gebara Filho, Security Assurance TIS, Amazon Web Services
- Cristiane Moncau, Corporate Counsel, Amazon Web Services
- Diane Young, Corporate Counsel, Amazon Web Services
- Debra Farber, Industry Specialist, Amazon Web Services

Document Revisions

Date	Description
March 2020	First publication