Hyperscaler Governance Solution

Contents

[1. Requirement: 2](#_Toc177463940)

[2. Solution: 3](#_Toc177463941)

# Requirement:

**Scoped Deliverables:**

1. **Security Guardrails Design and Implementation:**
   * **Design security policies** for network, data, and access management based on AWS best practices using IaC tools such as AWS CloudFormation or Terraform.
   * **Implement AWS Identity and Access Management (IAM) policies** to ensure secure access to AWS resources through IaC.
   * **Deploy network security measures** such as Amazon Virtual Private Cloud (VPC), subnets, security groups, and AWS WAF (Web Application Firewall) using IaC.
   * **Configure encryption mechanisms** for data at rest (e.g., AWS KMS) and in transit using IaC scripts.
   * **Set up automated security compliance checks** using AWS Config Rules and AWS Security Hub, integrated with IaC.
2. **Compliance Framework Development:**
   * **Develop and implement compliance controls** tailored to the identified regulations using AWS Config and AWS CloudFormation.
   * **Create compliance documentation and reporting templates** for audit purposes, automated through IaC.
   * **Integrate compliance controls** into the continuous integration/continuous deployment (CI/CD) pipeline using GitHub and IaC.
3. **Technical Governance Platform Establishment:**
   * **Design and implement a governance framework** that defines roles, responsibilities, and processes for AWS management using IaC.
   * **Develop and deploy a policy management system** to enforce security and compliance policies using AWS Organizations and AWS Service Control Policies (SCPs) via IaC.
   * **Create a central governance dashboard** to monitor compliance status, policy violations, and remediation actions using AWS Control Tower and AWS Management Console, automated through IaC.
   * **Conduct training sessions** for stakeholders on governance processes and tools implemented via IaC.
4. **Enhanced Visibility and Monitoring:**
   * **Deploy AWS-native monitoring tools** (e.g., AWS CloudTrail, Amazon CloudWatch, AWS Config) to track security and compliance events, automated through IaC.
   * **Implement logging and alerting mechanisms** for suspicious activities and policy violations using Amazon CloudWatch Logs and AWS CloudWatch Alarms through IaC.
5. **Knowledge Transfer and Documentation:**
   * **Prepare comprehensive documentation** for all implemented solutions, including architecture diagrams, configuration settings, and user guides automated through IaC.
   * **Conduct knowledge transfer sessions** with the internal IT team to ensure smooth handover and ongoing management of AWS-based solutions.
   * **Develop a training plan and materials** to upskill the internal team on maintaining and evolving the security and compliance guardrails through IaC.

# Solution:

**Proposed Solution & Architecture**

The worked with AWS cloud platform, security, SRE, logging, and networking teams to design the AWS organizational unit structure and then to use Control Tower to handle the Core OU and to provide AWS accounts to the Business Units.

The built out a secure Landing Zone with AWS’s Control Tower product that allowed the Enterprise IT team to launch an AWS account and have the matrix of security guardrails and identities automatically deployed. This enabled them to start onboarding Business Units to use AWS.

Control Tower has built-in security Guard Rails and automated account creation and configuration. Then provided guidance on industry best practices for security and operations to get the Customer to the point where a Business Unit could onboard to AWS to immediately start developing their applications.

A diagram of a software

Description automatically generated

Control Tower will be implemented in a Master Account, where all of the AWS Services used to create and manage AWS Organizations and Accounts were located. Part of the default implementation of Control Tower is a Core Organization with Audit and Logging Accounts. Each of these receive the default Control Tower configurations including IAM Roles and VPC Resources.

Any AWS Organization or AWS Account under Control Tower can have Service Control Policies (SCPs) and AWS Config Rules assigned for security guardrails. Control Tower was used to deploy 22 base, 13 strongly recommended, and 18 custom SCPs and Config Rules to AWS Accounts.

AWS SSO need to be integrated with MFA Authentication Tool. AWS SSO will be managed in the Master Account, and User Roles will be mapped directly to their defined user access patterns.

AWS Service catalog will be used in the actual creation of AWS Accounts under the Control Tower Domain. Here we need discussion with account creation team, how they are creating account using **Jonus** and how we can introduce AWS Service catalog and control tower.

GitHub will be used for automation orchestration. GitHub Action jobs will create and manage of resources, calling the AWS API to interact with Control Tower or individual accounts based on the need.

**Examples:**

* A GitHub Action jobs that would quarantine an EC2 instance in a given account by removing all access and controls, stopping the instance and copying a backup of the EBS Volumes to a quarantined VPC in the Audit Account.
* A GitHub Action jobs to trigger Service Catalog to create a new account in a given organization.
* A GitHub Action jobs that would create a privileged user for a given account in the event of a Break Glass scenario when someone needs immediate privileged access to an account.

In the end, the Rackspace will have a way to create an AWS Organization / Account with proper Security guardrails in an automated method that could be controlled by a central enterprise team, while allowing the individual Business Units to manage their own environments.

**AWS Services**

* **AWS** CloudFormation Template
* **AWS** Lambda
* **AWS** S3
* **AWS Networking Services** – VPC
* **AWS Management and Governance Services** – CloudWatch, Config, CloudTrail, Service Catalog, Organizations, Control Tower
* **AWS Security, Identity, Compliance Services** – IAM, Key Management Service, GuardDuty, AWS Single Sign-On

**Third-party applications or solutions**

* GitHub
* Terraform (Optional)
* MFA App like RSA