DEVELOPING ENTERPRISE ARCHITECTURES ON AMAZON WEB SERVICES

The SHI International Corp. framework to providing an operationalized Amazon Web Services platform that delivers traceability across the IT Landscape

Contents

The SHI Enterprise Architecture Assessment Framework for Amazon Web Services

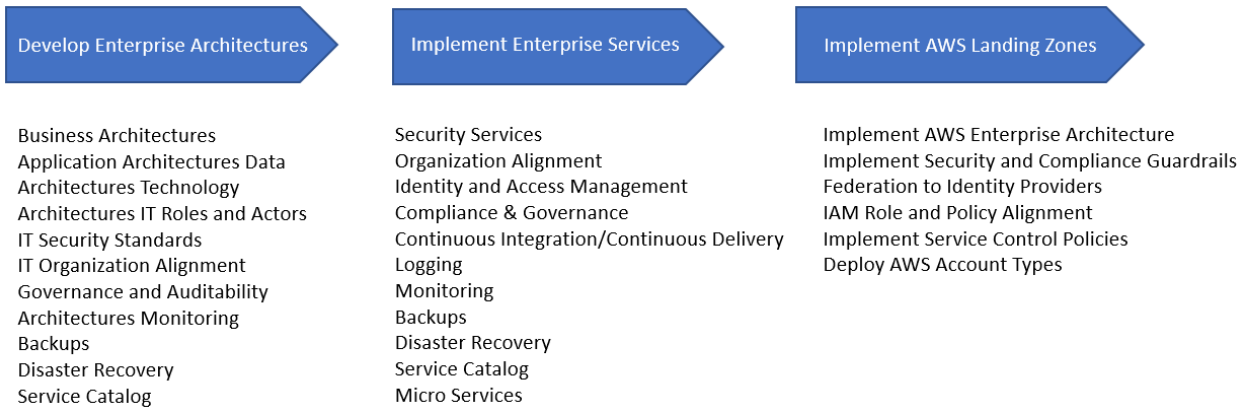
Executive Summary

This whitepaper is an overview of the SHI International Corp. Enterprise Architecture Assessment Framework for Amazon Web Services (AWS) service offering. The SHI Enterprise Architecture Assessment Framework for AWS service offering was developed so IT enterprises could understand, develop and implement a strategic approach and framework for AWS adoption that translated to their unique business vision. This whitepaper provides solutions to many of the challenges IT leadership face and focuses on developing the necessary enterprise architectures and foundational services to provide standardized, secure, and predictable outcomes for the services and enterprise workloads being implemented on the AWS platform.

The SHI Enterprise Architecture Assessment Framework gives IT enterprises the capability to develop and implement the following framework and key performance indicators (KPIs) across their IT organization:

1. Develop Enterprise Architectures for IT Organizational Alignment
2. Centralized Management of AWS Accounts and Resources
3. Operationalization of the AWS Platform by developing key AWS Enterprise Services
4. Auditable Compliance

The SHI Enterprise Architecture Assessment Framework addresses the discovery, planning and implementation of AWS Enterprise Architectures in three distinct phases.



Introduction

As an AWS Principal Consultant at SHI, I am involved in many conversations with IT leadership from a variety of industries across the world. From what I've seen, IT leadership is fully aware that AWS adoption is necessary to transforming their IT enterprise. They understand its role in enabling quicker responses to business requirements, providing agility to development teams, ensuring resilient and always available workloads, and offering a predictable cost associated with providing IT services to their enterprise customers. AWS adoption allows IT enterprises to stay competitive by allowing them to experiment with new technologies, adopt agile frameworks and reduce go-to-market timelines.

Among my many discussions with IT leaders, this statement resonates the most, "If I have a workload on AWS, it should be secure, and I should be able to demonstrate compliance to my leadership or to a third party." At the end of the day, these IT leaders want "***traceability across their IT landscape."***

Traceability is the capability to demonstrate how a workload or service on AWS is communicating with its data source, its interfaces, (other services it will use) and who or what is using the workload. Traceability is simply an audit trail.

The goal of this whitepaper is to illustrate how SHI's Enterprise Architecture Assessment Framework will develop and deliver a solution to every IT leader, how this solution can be used to provide necessary traceability, and to demonstrate the enterprise is safe and compliant. The starting point in delivering this solution is to develop enterprise architectures and foundational enterprise services for an IT enterprise.

The Origins and Benefits of Enterprise Architectures

Enterprise architectures were originally developed by the TOGAF® Standard, Open Group. The TOGAF® is a methodology and framework used to improve business efficiency by aligning IT goals with overall business goals and strategy. TOGAF® provides a framework of rules that ensures the IT organization will adhere to consistent standards, methods and communication. The alignment of an enterprise architecture to your IT organization ensures your IT enterprise, and the internal groups that support that enterprise, are aligned with the same goals, thereby ensuring the provisioning of services will align with your business standards and goals. The TOGAF® approach helps practitioners avoid being locked into proprietary methods, utilize resources more efficiently and effectively, and realize a greater return on investment. The TOGAF® framework directly influences AWS and their approach to developing enterprise architectures that ensure any architecture used on AWS will always be well-architected and future proof.

Translating Enterprise Architectures to an IT Operating Model

An IT organization is based on an enterprise schema that segments the business into departments, divisions, global regions, applications, or specific IT services. This enterprise schema can be a global or a multi-region model that can split and subdivide, or it can be simplistic and flat. Every customer has an enterprise schema that aligns to their current business model. The SHI Enterprise Architecture Assessment Framework translates your current state business model into a target operating model (enterprise architectures) by developing an enterprise organizational schema that aligns your business model to the AWS platform. This framework provides each IT organization with the capability to deliver compliant and centrally managed AWS services to their enterprise by ensuring the four KPIs listed above.

1 - Developing Enterprise Architectures for Organizational Alignment

In this section, we will illustrate the approach and framework needed to develop the necessary enterprise architectures and the associated enterprise services patterns, and how they align to your IT organization and services on the AWS platform.

AWS Technical Content Disclaimer

The following sections of this whitepaper use AWS specific content from, "**Establishing Enterprise Architecture on AWS"** and the "**AWS Landing Zone**." SHI will demonstrate how we use the AWS

Establishing Enterprise Architectures framework for the development of relevant enterprise architectures that align to SHI Enterprise Services, and the capabilities to deliver compliance using AWS Organizations and AWS Landing Zones. Using AWS specific content will demonstrate that SHI is following a proven methodology that AWS uses on their internal platform workloads, this ensures we are providing our customers with a relevant solution that follows AWS best practices. The AWS content we are directly using is shown as text that is bolded and italicized and is directly quoted from an AWS whitepaper. We have also included additional insights from first-hand experience on how we use this AWS framework to develop relevant enterprise architectures for SHI customers.

The AWS Approach Developing AWS Enterprise Architectures

AWS enterprise architectures are developed using the AWS enterprise architecture tenets and enterprise architecture domains. These two AWS frameworks contain rationalization standards that allow the IT organization to review and organize their current enterprise state, and then align AWS to their organization's business models and organizational schema.

AWS Enterprise Architectures

AWS describes enterprise architecture as follows, "***Enterprise architecture aims to define the target IT landscape that realizes the business vision and drives value.***" In other words, enterprise architectures align to a common model for the enterprise's IT organization and allows the organization to provide a solution for the application, service or AWS workload. The challenge we face is to develop an enterprise architecture "***that translates into a model that the customer understands, and that aligns to their business logic.***" To develop a relevant enterprise architecture, AWS asks us "***To define the target IT landscape.***" The first step in defining the IT target landscape is to rationalize the applications and services our enterprise is currently providing to the business, where those applications and services are located, and to identify what compliance guardrails those applications and services must have. This approach allows the IT organization to "***identify the programs and architectures needed to realize the target IT state.***" Once identified, we can align them to the IT organization's enterprise business customers and envision the future state.

AWS identifies the following as key goals of enterprise architecture:

1. ***Analyze and evolve the organization's business vision and strategy***
2. ***Describe the business vision and strategy in a common manner (for example, business capabilities, functions, and processes)***
3. ***Define the programs and architectures needed to realize the target IT state***
4. ***Provide tools, frameworks, and specifications to support governance in all the architectural practices***
5. ***Enable traceability across the IT landscape***

AWS Architecture Tenets

AWS enterprise architecture tenets are general rules and guidelines that inform and support the way in which an organization sets about delivering its specific business model.

Enterprise architecture tenets are a lasting standard that aligns to business goals. The framework should be used as a template to develop application workloads that are secure, compliant and cost-effective, and align to a business vision. Enterprise architecture tenets should always be used for enterprise architecture decisions that drive your future enterprise on the AWS cloud. The framework should be

used for architecture planning, framing policies, procedures, and standards, and for supporting the resolution of contradictory situations. Architecture tenets should also be heavily leveraged during the architectural review phases of applications and workloads before they go live to ensure the correct target landscape is being realized. Here are six examples of enterprise architecture tenets identified by AWS:

1. ***Maximizing Cost Benefits***
   An IT organization should always approach provisioning their workloads from the cost perspective. IT organizations "***should encourage architects, application teams, IT stakeholders, and business owners to always consider the cost effectiveness of their workloads. This encourages your IT enterprise to focus on projects that differentiate the business value, not the underlying infrastructure.***" When the IT organization compares the capital and operational expenditure for each workload (a comparison that includes running the workload or service in their data center and on the AWS platform), the results will be the most cost-effective solution for the IT organization. The benefit of studying solution alternatives gives the IT organization an idea of costs associated with implementing the service throughout the service lifecycle and a solution analysis that can be passed on to your IT customers.
2. ***Business Continuity***
   "***The business continuity tenet informs and drives the non-functional requirements for all current and future workloads in your enterprise.***" The AWS cloud has a strategic advantage to deliver business continuity by providing the ability to segregate workloads and services regionally and globally on highly redundant infrastructure. This tenet guides the application teams to develop workloads, applications and services that are natively highly-available by leveraging the reliability and availability of the AWS cloud. This tenet allows for the development of enterprise architectures that use the AWS cloud to allow services on AWS to be segmented by regions by providing high availability to workloads through redundant AWS infrastructure.
3. ***Agility and Flexibility***
   The agility and flexibility tenet enforces the need that all enterprise applications should be well-architected to "***respond rapidly to business requirements as customer behaviors evolve.***" This approach enables teams to validate and test new technologies, implement continuous integration and continuous delivery by allowing development teams to rapidly set up environments to compare architectures, AWS services and best practices. This flexibility and agility should always be considered when developing enterprise architecture as it provides your IT organization the ability to experiment and develop applications rapidly and efficiently.
4. ***A Cloud-First Strategy***
   "***The cloud-first strategy tenet is key to an organization that wishes to migrate applications and workloads to the cloud.***" The cloud-first strategy prescribes that new applications should be implemented on AWS, prohibiting the use of legacy hardware, architectures and systems for their solutions. A cloud-first strategy always takes into consideration cost, agility, business continuity and security. Being cloud-first only allows applications, services and workloads to be implemented on approved architecture standards and infrastructure.
5. ***Organizational Units***
   An enterprise should use AWS organizational units to ensure its target landscape reflects the enterprise's organizational schema. AWS organizational units segment cloud activities and

provide governance for your IT organization's business vision, while allowing the necessary autonomy individual business units may need to deliver applications and service workloads to the IT organization.

6. ***Security***
   AWS states, "***The security tenet describes the security values of the IT organization.***" The security tenet allows your architecture team to determine what level of trust they have in the cloud, provides solutions that use AWS services to align with your IT organization's regulation, compliance and security requirements, and guides you in deciding where your enterprise is on that scale.

AWS Architecture Domains

To deliver the enterprise architecture goals, we will use the AWS enterprise architecture domains framework. "***The enterprise architecture domain framework will guide your organization's business, information, process, and technology decisions and enable it to execute its business strategy and meet customer needs.***" The enterprise architecture domains identify the existing current state and how that will influence the development of enterprise architectures. AWS has identified four architecture domains that should be used to define an IT organization.

1. ***Business Architecture Domain***
   "***The business architecture domain describes how the enterprise is organizationally structured and what functional capabilities are necessary to deliver the business vision.***" This enterprise architecture defines the use cases, strategy and business goals for adopting AWS. Business architecture identifies the goals and strategy and should address the questions WHAT and WHO:
   - ***WHAT is the organization's business vision, strategy, and objectives that guide creation of business services or capabilities?***
   - ***WHO is executing defined business services or capabilities?***

   An example of "what" is to reduce the number of the IT organization's data centers by consuming compute and storage capacity on AWS. Reducing the IT organization's data centers provides a business vison for less capital expenditures and is a strategic approach to providing IT services based on a pay-as-you-go model, and developing applications based on specific business use cases or business units.

   The Who - Roles and Actors

   AWS defines an actor as "***A person, organization, or system that has a role that initiates or interacts with activities. Actors belong to an enterprise, and in combination with the role, perform the business function.***" The "who" is a list of all users in your IT organization that work with and support IT enterprise systems. Per AWS, "***Understanding actor-to-role relationships is necessary to enable organizational change management and organizational transformation.***" An example of "who" is to develop an enterprise architecture that defines the "actors" that are specific to an application development team. The application development team will use an enterprise directory (Microsoft AD) that contains existing access roles, allowing the IT organization to map the actor to that role. This mapping will allow the actors (security, infrastructure, application support teams) access to the environments they support (DEV, Pre-PROD, PROD) on AWS. This approach will define specific business processes by aligning them to your IT organization. This example enterprise architecture should be used to define the

necessary Identity and Access Management requirements to provide the security, compliance and support for enterprise services running on AWS to the IT organization.

2. ***Application Architecture Domain***
   The application architecture domain describes the enterprise line-of-business applications and how they interface with other workloads and services, and their relationships to the core business processes of the organization. Application architectures address the current state of the application and ask the question, "***HOW are previously defined business services or capabilities implemented?***" IT organizations commonly align their line-of-business applications with a business unit, a global region or a company division. This approach allows the IT organization to segment the application or service, and align it to specific support teams, global regions or industry compliance rules that the business must provide to provision the application to the IT organization's customers.
3. ***Data Architecture Domain***
   "***The data architecture domain describes the structure of an organization's logical and physical data assets.***" This domain identifies where the data currently resides, the necessary compliance rules for the governance, and management of those data resources.
4. ***Technology Architecture Domain***
   "***The technology architecture domain describes the software and hardware needed to implement the business, data and application services.***" This domain identifies what technologies will be used to develop the architecture, and deliver the application, workload or service on AWS.

The architecture domains and their retaliation to each other are to be reviewed to develop the IT organization's strategy for future state architectures as well as providing the following answers to these questions:

- Where are assets located, and how will that location influence compliance?
- Why is the service, workload or application being re-factored, and is there an existing AWS service that can provide the solution?

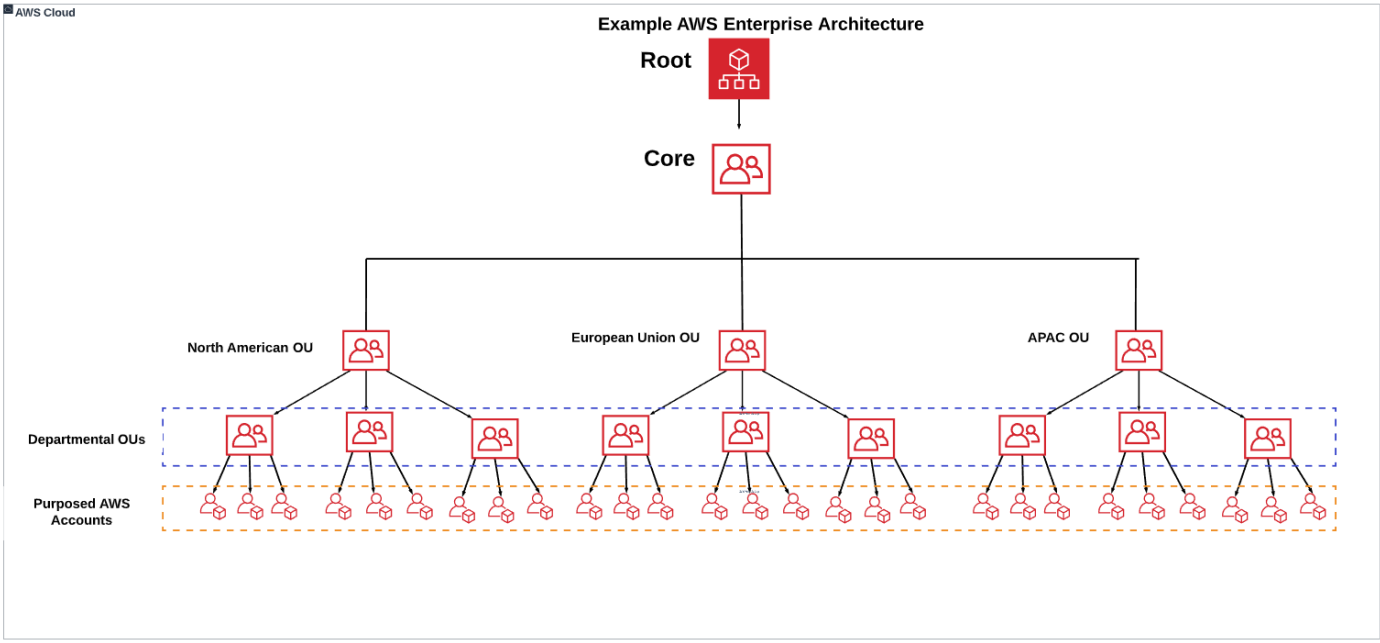
To deliver a relevant enterprise architecture to your AWS enterprise, we must align all the information we have discovered by reviewing our enterprise architecture tenets and aligning that information to the enterprise architecture domains. Remember that the AWS enterprise tenets are the "rules," and the enterprise architecture domains are where you apply those rules. With this information in hand, we will now apply this to our AWS enterprise by developing an operating model that aligns the AWS platform to our business.

2 - Centralized Management of AWS Accounts and Resources

Developing Enterprise Architectures that Provide Centralized Management

As we discussed in the Introduction section of this whitepaper, an IT organization is based on an enterprise schema that segments the business into departments, divisions, global regions, applications, or specific IT services. The reference architecture below illustrates a sample enterprise architecture that segments our business by using organizational units (OUs) into three global regions, that is then subdivided by departmental OUs, with each department OU having its own specific AWS accounts assigned to it. The top level OU is a core OU that we will explain in detail in later sections of this

whitepaper. Using the SHI Enterprise Architecture Assessment Framework methodology, we will have identified the necessary requirements to design an enterprise architecture and align it to your IT organization to provide a centralized management operating model that includes all the necessary enterprise services. We will deliver this IT organizational alignment by using AWS Organizations.



Delivering Enterprise Architectures with AWS Organizations

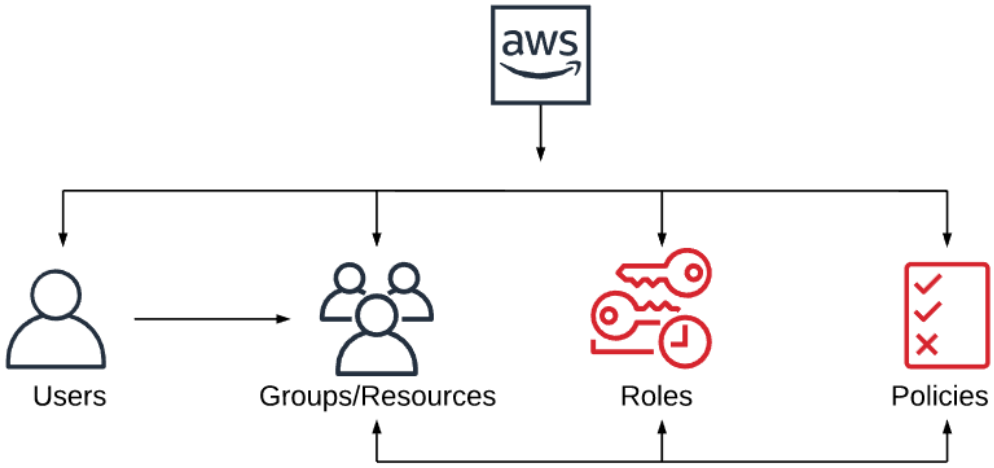
AWS Organizations

AWS Organizations is a scalable permissions and account management service that enables IT organizations to consolidate multiple AWS accounts into an enterprise organization. AWS Organizations provides well-architected, highly scalable AWS accounts that are aligned to your current IT organization's enterprise business model, allowing for a predictable and compliant consumption of AWS resources. AWS Organizations includes account management and consolidated billing capabilities that enable IT organizations to better meet their budgetary, security and compliance needs. AWS Organizations provides SHI customers with a predictable and compliant consumption of AWS resources, allowing your IT enterprise to automate account creations, create groups of accounts based on business needs, and apply guardrails based on security and compliance requirements. AWS Organizations also provides additional value by native integration of AWS services to deliver an end-to-end solution for compliance and traceability across your IT landscape. AWS Organizations will ensure our AWS enterprise architectures are always well-architected, aligned to business goals, and are always secure, compliant, cost-effective and future proof. Once our enterprise architecture is developed (in our example of enterprise architecture, we segmented our operating model into regions and departments), we focus on identifying specific business visions for our AWS accounts, and how they will be used and accessed.

What is an AWS Account?

An AWS account is a perimeter around AWS resources that includes AWS Users, AWS Groups and Resources, AWS Identity and Access Management Roles (IAM) and the IAM Policies that are aligned to those roles that in turn, provide access to AWS resources. AWS IAM Policies allow the IT organization to

manage access in AWS by attaching the IAM Policy to IAM identities (users, groups of users, or roles) or to AWS resources. IAM is tightly integrated with AWS Organizations, providing centrally managed, scalable Role Based Access Control (RBAC) access across all AWS accounts in the IT organization.



An AWS account is a perimeter around AWS resources that include the following attributes:

- An account baseline and administrative boundary for accessing AWS resources
- A Service Control Policy (SCP) that provides access to AWS-specific services for that AWS account
- Identify and Access Management (IAM) policies to provide Role Based Access Control (RBAC) for AWS Users, Groups and AWS services available in the account

AWS Service Control and IAM Policies, how they integrate into AWS Organizations, and how they provide security and compliance guardrails for your AWS accounts is covered in later sections in this whitepaper.

The Benefits of Developing a Multi-AWS Account Strategy for your IT Organization

An AWS multi-account strategy offers the capability to provision AWS accounts in a consistent hierarchy based on the IT organization's enterprise architectures. A multi-AWS account strategy will also provide solutions to regional-specific compliance requirements that may influence how a service or application is hosted in global region. For example, there are different laws and regulations for the storage of data in the European Union, APAC, and the United States. By aligning these regional requirements to the IT organization's enterprise architecture, implementing an AWS account strategy for each region will provide a framework and approach to address the necessary compliance requirements for each region, and also provide the billing in the necessary currency for the accounts and their associated resources in each region. Finally, developing multi-AWS accounts for specific environments provides a simplified charge and show back model. Specific business units, application teams and developers that have their own accounts can be charged for the services they are consuming, such as specific costs for development, hosting applications and skill building.

The first step in developing a multi-AWS account strategy is to identify how your AWS accounts will be used, (the purpose and business vision for the account), and how the accounts will align to your IT organization's enterprise architecture. Using this multi-AWS account approach gives you the capability for standardized consumption of AWS across accounts, and the ability to align your AWS accounts to provide and consume shared services (shared infrastructure and transit), security, and role-

based access to account resources (IAM and SCP), audit services (logging), compliance for regional and industry requirements, consolidated billing, and AWS accounts for specific environments (workload, DEV, Test, PROD). We will discuss providing shared services and the role of the Core OU in section 4 of this document.

When starting to make AWS account decisions, consider the following:

1. Identify Account Requirements - What is the purpose and commonalities for each AWS account?
2. AWS accounts should align to your enterprise architecture. For example, your accounts should align to Business Units, Environments (DEV, TEST), Regional (USA, Europe), workloads or individual accounts. These account classifications allow you to create a model for standardizing the AWS account types and the guardrails (compliance rules) that are needed for each account type.
3. AWS account lifecycles should be identified. For example, the type of AWS account that is being used and how long will the account be used for (long term, workload or by project).
4. Compliance and security requirements for the workloads and the AWS services running in the account.
5. How the accounts will be accessed by AWS Principals (Users and Services).

Once the account types and account lifecycles are identified, it is time to align your account types to your enterprise architecture and organize them into organizational units within AWS Organizations.

Organizing your AWS Accounts using AWS Organizations OUs

One of the key goals in developing enterprise architectures is the alignment of AWS resources to the IT organization's business model. Aligning AWS resources to an operating model allows the IT organization to have control on the locations of the AWS resources (regions), what AWS services are available to the AWS account (what global region that you can use AWS services), as well as the access of the AWS accounts (departments). To deliver this solution, we will use AWS Organization OUs as a solution to organize our AWS accounts and the resources that reside in each AWS account.

AWS describes the use case for organizational units as follows:

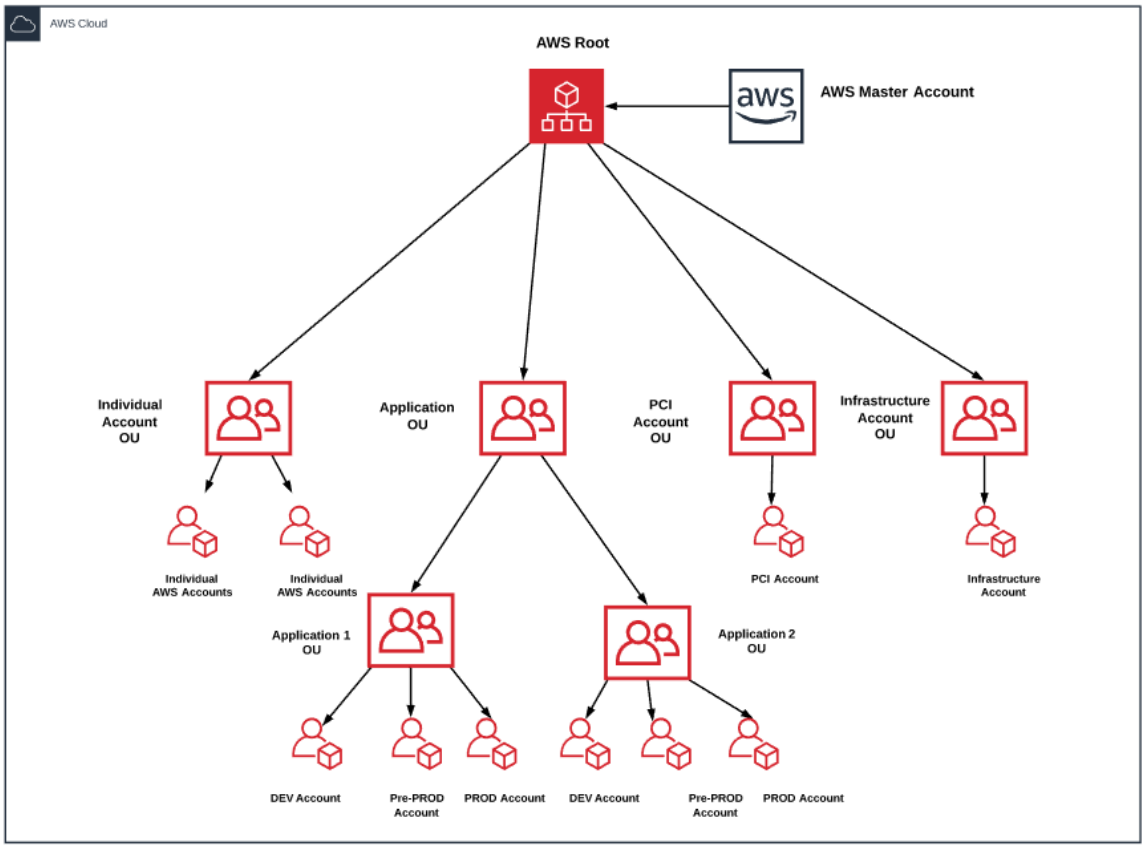
AWS Organizations lets you arrange your AWS accounts into groups called organizational units (OUs) that reflect your enterprise architecture model. Within and across OUs, you can define centrally managed policies and apply them in a uniform manner. AWS Organizations can also define how accounts are created and removed from within the IT organization. AWS Organizations provides the following features and capabilities:

- ***Replicate your organizational enterprise architecture in your AWS enterprise environment.***
- ***Provide your business units autonomy while maintaining global and IT Organization governance rules.***
- ***Manage your target IT landscape (the creation and deletion of accounts), compliance and global expenditures.***

In AWS Organizations, the root contains the AWS Master Account, which is the parent container for all the AWS accounts for your organization and the AWS account that is used to create your AWS organization. The master account is used to create "member" accounts in your organization, to invite

and manage invitations for other accounts to join your organization and remove accounts from your organization. The AWS Master Account can attach policies to entities, such as administrative roots, organizational units (OUs), or specific accounts within your organization, to provide compliance and security guardrails for each specific account type, purpose or use case.

AWS Organizations uses OUs that are nested under the root to provide the capability to group AWS accounts together to align them to your IT organization's enterprise architecture. The reference architecture below illustrates another example of a typical AWS Organization aligned to a sample enterprise architecture. This sample AWS Organization has four OUs under the AWS root to provide the following business vision for your IT organization:



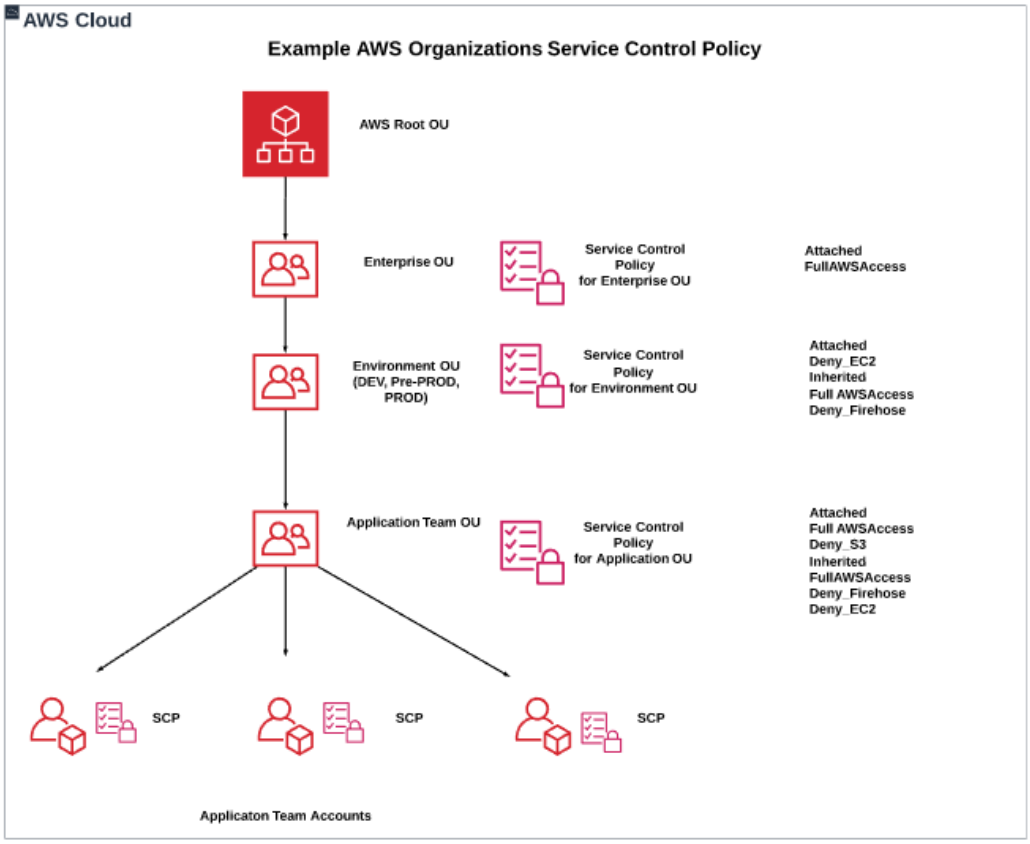
- **Individual Account OU –** Groups individual user AWS accounts that are being used for learning AWS. By placing these accounts in an OU, it allows the IT organization to place specific controls on these accounts, limit access to specific AWS resources, and provide monitoring and alerting for account spending limits.
- **Application OU –** This is a parent OU that will contain all of the IT organization's application accounts. Each line-of-business application will have a unique, individual OU under this parent OU to provide specific account guardrails and compliance baselines.
- **Application OUs –** Each line-of-business application will have an AWS account assigned to the application for each application environment (DEV, Pre-PROD, PROD). These accounts will contain specific security and compliance baselines that are required for each line-of-business application in each environment.

- **PCI Account OU –** In our IT organization, we have an application that must meet PCI compliance. We have an OU just for PCI compliant workloads, and we have placed an AWS account in an OU that will only allow PCI compliant AWS services and have specific compliance guardrails assigned to the account via SCP policies.
- **Infrastructure Account OU –** This OU will contain an AWS account that will provide IT Infrastructure services to the IT organization.

AWS Service Control Policies (SCP) and Organizational Units (OUs)

AWS Organizations uses AWS Service Control Policies (SCPs) to place boundaries around the permissions that AWS Identity and Access Management (IAM) policies can grant to entities in an account, such as IAM users and roles. The AWS account inherits the SCPs defined in, or inherited by, the OU above it in the hierarchy. SCPs are used for access to AWS services, resources, and individual API actions the users and roles in each member account can access and define conditions for when to restrict access to AWS services, resources, and API actions. As an administrator of the master account of an organization, you can use SCPs to specify the maximum permissions for member accounts in the organization.

In this example reference architecture, we have three levels in an OU hierarchy: Enterprise OU, Environment OU and Application Team OU. At each OU level, we have attached an SCP that is unique to that OU. Our reference architecture illustrates how the SCP policies flow down the OU hierarchy, providing specific compliance guardrails at each level. This AWS Organizations feature allows IT organizations to create specific purposed accounts for each business vision or use case and provide compliance guardrails by only allowing specific access to AWS services to the AWS principals (users, groups or roles) to accounts that reside in that OU.



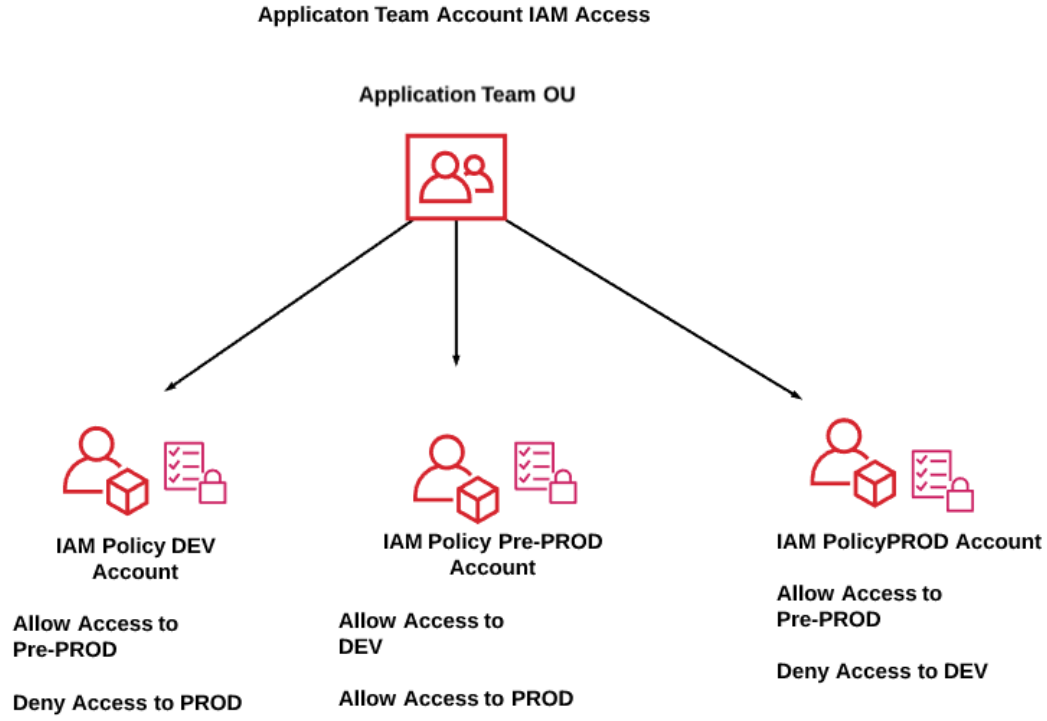
This enterprise architecture illustrates that the top-level enterprise OU has an attached policy that provides FullAWSAccess to the enterprise OU. As we move down the OU hierarchy, we see two member OUs, the Environment and Application Team OUs, and at each level of the OU hierarchy, we have a different SCP defined. This allows our IT organization to limit specific AWS services to the accounts in each OU. This capability provides the IT organization to allow or restrict specific AWS services within and across OUs in the hierarchy. SCPs are combined with IAM roles and policies to provide the IT organization the ability to allow only approved AWS services to specific IAM users and roles.

AWS Organizations and Identity and Access Management

In section two, we discussed the AWS Business Architecture domain framework and how it identifies the "***WHO will be executing defined business services or capabilities?***" AWS defines "WHO" as an actor or a role, the entity that will be interacting with AWS services. The WHO may be a specific IAM user or it can be a defined IAM role that is granted to a group of users or a service, (for example, AWS EC2). An account administrator can control access to AWS resources by attaching permissions policies to IAM identities (users, groups, and roles), providing granular control over users and roles in individual accounts. AWS Organizations expands that control to the account level by giving you control over what users and roles in an account or a group of accounts can do. All AWS account resources, including the account roots, OUs, AWS accounts, and the policies in an organization, are owned by an AWS account. The permissions to create or access a resource are governed by permissions policies, and in the case of an AWS Organization, the AWS Organization Master Account is the owner of all the resources and provides a central place to manage those resources. An AWS account within an OU defines the users for that account and the corresponding roles that users can adopt. With IAM, you can securely control

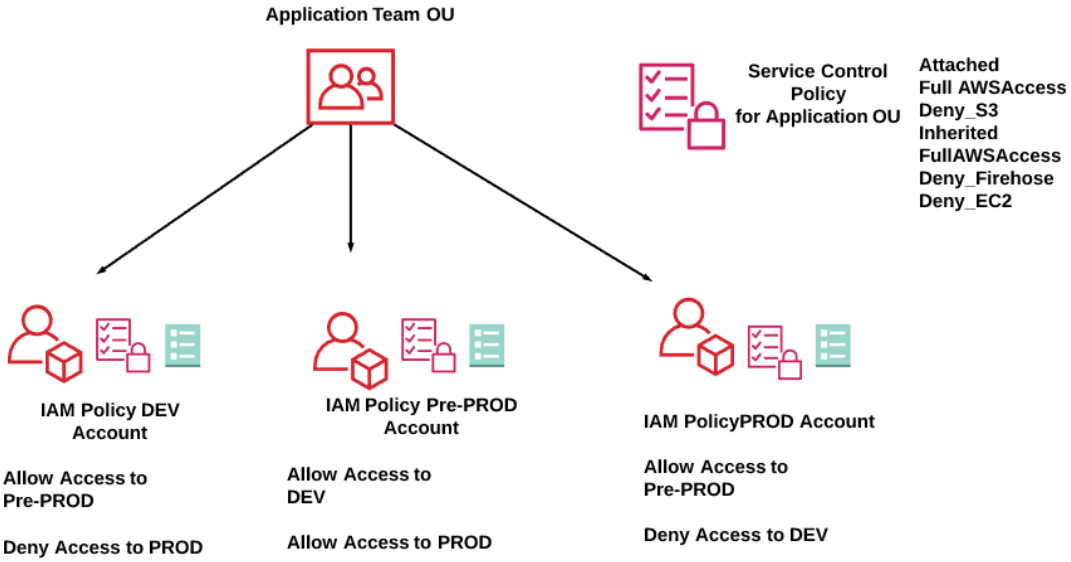
access to AWS services and resources for your IAM users and roles. You can also create and manage AWS users and groups and use permissions to allow and deny their access to AWS resources.

The resulting permissions are the logical intersection of what is allowed by AWS Organizations at the account level, and what permissions are explicitly granted by IAM at the user or role level within that account. In other words, the user can access only what is allowed by both the AWS Organizations SCPs and IAM policies. If either blocks an operation, the user can't access that operation. The value of AWS Organizations allows our IT organization to centrally set and manage RBAC permissions across all the AWS accounts. In the reference architecture below, we have the Application Team OU, and there are three accounts that reside in their team OU. The IAM "actors" are the members of the Application Team and have access to the three accounts in the OU, (DEV, Pre-PROD and PROD). We want the IAM actors to have access to all three accounts, but we want to restrict the AWS cross-account access between the accounts. We can deliver this solution using AWS Organizations and restrict access between the accounts with IAM policies.



Identity and Access Management and Service Control Policies

SCPs and IAM provide an end-to-end compliance solution for AWS account and AWS service access to all the member accounts. In the reference architecture below, we have an OU that contains AWS accounts for an application team (DEV, Pre-PROD, PROD), and an SCP that is applied to the Application Team OU that restricts access to specific AWS services in the application accounts for that team of developers. In this example, we have an AWS account for each environment and an IAM policy that only allows cross-account access for each account. This combination of only allowing specific AWS services and cross-account access rules provides a predictable and auditable baseline for compliance and access control.



3 - Operationalizing the AWS Platform by Developing Key Enterprise Services

A successful AWS adoption starts with operationalizing the AWS platform by implementing a solid foundational approach to reference architectures, security frameworks, and organizational effectiveness, and aligning these requirements to "AWS enterprise services." AWS enterprise services are patterns that will provide the foundational components necessary to deliver compliance and traceability across the IT landscape. Enterprise services are foundational, and each AWS Enterprise Service should be available to any workload or service that will be implemented on the AWS platform.

Providing enterprise services to all AWS resources will provide the capability to produce auditable logs, and fully redundant AWS services and workloads to the enterprise. The SHI Enterprise Architecture Assessment Framework ensures all AWS resources will have the necessary AWS enterprise service alignment in its IT organization's associated enterprise architecture. AWS enterprise services will use the following AWS services and leverage their native integration in the AWS platform to deliver an end-to-end solution that provides compliance and traceability across your enterprise IT landscape.

AWS Enterprise Services

AWS enterprise services are necessary to fully operationalize the AWS platform. Providing these AWS enterprise services to all AWS resources in your IT organization will ensure that all AWS workloads are resilient, scalable and provide compliant and consistent outcomes for the IT organization's business vision. In the table below, we list example AWS enterprise services, the details of the service, and a list of AWS services that can be used together to provide a solution. For example, the AWS enterprise service, "Security," has several key requirements that we must make available to the AWS workloads in our account:

- Cloud Security Controls
- Encryption
- Role Based Access Control
- Virtual Networking Security
- Log Analysis

The SHI Enterprise Architecture Assessment will identify all enterprise architecture tenets necessary to deliver the required security to our AWS workload or service. We then align the AWS security services necessary to deliver that vision to our enterprise architecture. SHI has identified 11 AWS enterprise services that should be included in each enterprise architecture, and an example of an AWS service that can be used to deliver the solution.

AWS Enterprise Service	Service Details	AWS Service Solution
Security	Cloud security controls for services and applications, encryption, Virtual Networking, Log Analysis, Secrets Management	Amazon Guard Duty, Amazon Inspector, AWS Trusted Advisor, Amazon Macie, VPC Flow Logs, AWS Athena, AWS KMS, AWS Secrets Manager
Organization Alignment	Multiple AWS accounts, multiple regions, global regions, application teams, workloads	AWS Organizations, AWS Landing Zones
Identity and Access Management (IAM)	AWS Accounts, AWS Roles and Actors, RBAC, Federation	AWS IAM, AWS Service Control Policies, AWS SSO
Compliance & Governance	Regulations, Compliance Standards, Standard Operating Procedures, Governance	Amazon Guard Duty, AWS Config, AWS Config Rules, AWS Artifact, S3 Events, CloudFormation, AWS WAF
Continuous Integration Continuous Delivery (CI/CD)	Infrastructure as Code, Agile Development	AWS CodeCommit, CodeDeploy, CodePipeline
Logging	Logging requirements, Logging Architectures, Logging Lifecycles, Role and Actors	Amazon CloudTrail
Monitoring	Service Metrics, Metric Alarms and Auto-remediation	Amazon CloudWatch
Backups	Data Resiliency and Data Classification	AWS Snapshots, AWS Cross-Region Replication
Disaster Recovery	Application and Service Continuity	AWS VPC
Service Catalog	Self Service for AWS Accounts, Application Portfolios, and Documentation for Business Units and Enterprise Developers	AWS Service Catalog
Micro Services	Serverless, API Gateway, Containers, Relational Database Services	AWS Lambda, AWS API Gateway, AWS RDS, AWS ECS

**These are examples of foundational enterprise services; additional services may be necessary for your IT organization.*

4 - Auditable Compliance

Delivering Compliance Starts with Delivering the Key Goals of Enterprise Architectures

In section two, we identified the key goals of Enterprise Architectures, and we discussed how to align those goals to the IT organization's business vision and strategy by using the AWS enterprise architecture tenets and AWS enterprise architecture domains. We used that information to make data-driven decisions and to develop an enterprise architecture that aligns to AWS Organizations.

In section three, "Operationalizing the AWS Platform by Developing Key AWS Enterprise Services," we identified the 11 AWS enterprise services we want all our AWS resources to leverage and always have access to. These AWS enterprise services will be integrated into our AWS enterprise architectures to provide the foundational services that will deliver traceability across the IT landscape.

Up until this point, we have used the SHI Enterprise Architecture Assessment Framework to deliver the following key goals of AWS enterprise architectures:

1. ***Analyze and evolve the organization's business vision and strategy.***
2. ***Describe the business vision and strategy in a common manner (e.g. business capabilities, functions, and processes).***
3. ***Define the programs and architectures needed to realize the target IT state.***
4. ***Provide tools, frameworks and specifications to support governance in all the architectural practices.***

In this section, we will discuss and illustrate on how to deliver this remaining key goal of an AWS enterprise architecture:

5. ***Enable traceability across the IT landscape.***

Enabling traceability across the IT landscape means having the ability to provide the following to your IT organization:

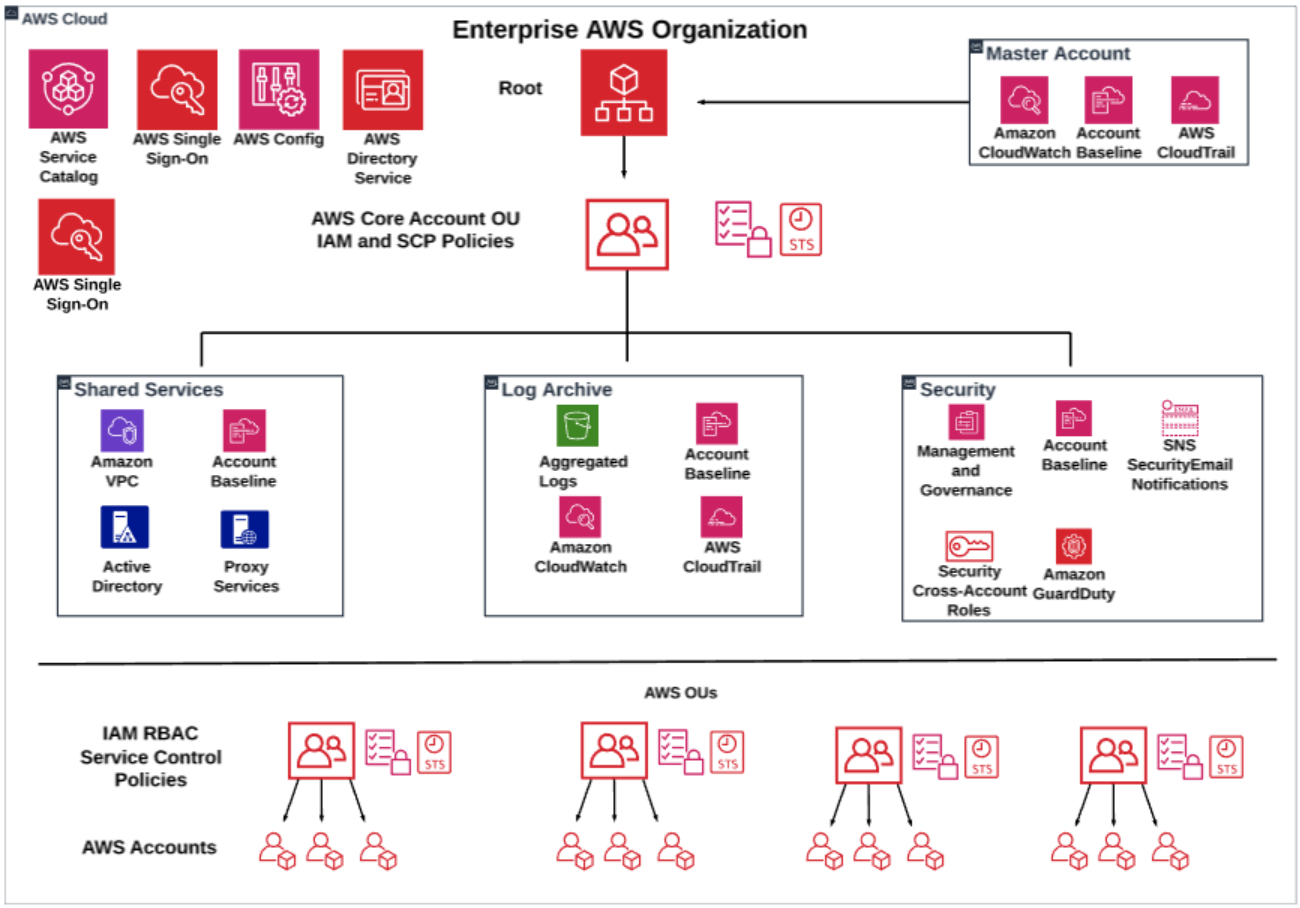
- The ability to track and audit every change made in every environment in our AWS Enterprise.
- The capability to provide a multi-AWS account solution that has specific business vision and compliance controls tailored to that account business vision.
- Centralized account billing management capability for charge-back.
- Self-Service for approved AWS Enterprise applications and AWS Accounts.

The following sections will illustrate how to use the AWS Organizations and AWS Landing Zones services to ensure that specific standards for governance are available in every AWS account and environment that is in our enterprise (all the architectural practices). The AWS Landing Zones service implements enterprise AWS architectures for shared services, monitoring, logging, billing, application portfolio management and security to all AWS accounts in our IT organization. AWS Landing Zones provides the IT enterprise with the ability to standardize specific AWS accounts that, when implemented, have the necessary compliance guardrails around the AWS resources available to that account.

AWS Landing Zones

AWS Landing Zones is a series of AWS services that is deeply integrated into the AWS Organizations framework, providing additional AWS capabilities across your implementation of AWS accounts and enterprise architectures. AWS Landing Zones starts with a well-architected AWS Organizations implementation and adds centralized logging, a cross-account permissions structure, and an automated "Account Vending Machine" that uses the AWS Service Catalog service, providing you the ability to deploy purposed AWS account types in an automated compliant manner. The AWS Landing Zone service deploys an AWS account architecture that utilizes "core" AWS accounts that provide specific shared services to all AWS accounts in your organization. The AWS core accounts utilize specific AWS services that allow your enterprise to provide the tools and frameworks that support governance and traceability across the IT landscape. The AWS Landing Zones service uses AWS Organizations to provide visibility into user activity and the ability to validate compliance across multiple AWS accounts in your IT organization.

The implementation of AWS Landing Zones creates a Master Account and three core accounts, Shared Services, Log Archive and Security. This AWS account architecture will provide the tools and frameworks to support governance and traceability across the organization. We will review the four AWS accounts and how this AWS enterprise architecture will benefit your IT organization, allowing you to deliver compliance across all accounts and environment architectures. The reference architecture below illustrates an Enterprise AWS Organization deployed using the AWS Landing Zone service.



Master Account

In section two of this whitepaper, we reviewed what the role of the Master Account is in AWS Organizations. The Master Account acts as the central hub for management for your AWS member accounts deployed via AWS Landing Zones. The Master Account also rolls up account activity from multiple AWS Accounts into a single invoice and provides a centralized way of tracking costs via consolidated billing.

AWS Landing Zones Core Accounts

Shared Services

The AWS shared services account is used for creating infrastructure shared services, such as enterprise directory services (Microsoft AD), proxy services, infrastructure monitoring, CI/CD pipelines, configuration management and a networking transit VPC. This account hosts an Amazon Virtual Private Cloud (Amazon VPC) that can be automatically peered with new AWS accounts created with the Account Vending Machine (AVM) to provide the necessary shared services (tools and frameworks) to the IT organization's AWS accounts.

Security

The security account creates auditor (read-only) and administrator (full-access) cross-account roles from a security account to all AWS Landing Zone managed accounts. These roles are to be used by security and compliance teams to audit, host custom AWS Config Rule Lambda functions, and perform automated security operations, and break-glass remediation actions. The security account is also used to manage the AWS GuardDuty service findings and to receive Simple Notification Service (SNS) security notifications for the AWS member accounts in your organization. This account should be restricted to authorized security and compliance personnel, and related security or audit tools.

Logging

The log archive account creates a central Amazon Simple Storage Service (S3) bucket for storing a copy of all AWS CloudTrail and AWS Config log files to this centralized log account. This logging enterprise architecture ensures compliance by sending a copy of all logs to a central location that is a single source of truth, and that is only accessible by security and compliance teams. AWS CloudTrail and Amazon CloudWatch logs for the workload are also created locally in the account so developers and operation teams can have access to the workload logs.

How AWS Landing Zones Delivers Compliance

When AWS Landing Zones deploys a new member account, it applies a security baseline that includes the following default configurations to the new AWS account. AWS accounts that are provisioned by AWS Landing Zones can have several different baselines that provide specific compliance and security baselines.

Amazon CloudTrail and CloudWatch

When a new AWS account is provisioned, one CloudTrail trail is created for that account and configured to send CloudTrail logs to a centrally managed Amazon S3 bucket in the log archive account. This ensures all API calls that are made from that AWS account are sent to a separate logging account, in turn providing a secure and centrally managed architecture for these CloudTrail logs. This ensures that all API calls are tracked across all accounts and managed in a central location (traceability), and access to these logs will only be made available to security teams and audit resources. Additionally, CloudWatch logs are

also configured in the local account to provide operations teams with the ability to determine operability metrics, alerts and remediation triggers to AWS microservices (AWS Lambda) derived from these CloudWatch events.

AWS Config

When AWS Config is enabled, account configuration log files are stored in a centrally managed Amazon S3 bucket in the log archive account, and each new account will have the AWS Config enabled to provide account compliance baselines that align to the business vision, compliance and environment rules for that account. The AWS Config service can send notification alerts when configuration changes to AWS Config Rules are detected in your AWS account.

AWS Config provides the following compliance features:

- Centralized auditing and governance for all the accounts in your AWS organization.
- AWS Config enables you to record software configuration changes within your Amazon EC2 instances and servers running on-premises, allowing your IT organization to gain visibility into operating system (OS) configurations, system-level updates, installed applications, network configuration and AWS account access.

AWS Config Rules

AWS Config rules are enabled for monitoring storage encryption (Amazon Elastic Block Store, Amazon S3, and Amazon Relational Database Service), AWS Identity and Access Management (IAM) password policy, root account multi-factor authentication (MFA), Amazon S3 public read and write, and insecure security group rules. AWS Config works with pre-built rules or your own custom rules for evaluating provisioning and configuring of your AWS resources as well as software within managed instances, including Amazon EC2 instances and servers running on-premises.

AWS Identity and Access Management

AWS Identity and Access Management (IAM)

Configures a default IAM password policy with the following settings:

- Allow users to change their password: true
- Prevent users from changing expired passwords: false
- Password complexity: Require uppercase, lowercase, symbols, and numbers
- Minimum password length: 12
- Prevent reusing passwords: 6
- Maximum password age: 90 days

AWS Cross-Account Access

AWS Landing Zones is tightly integrated with AWS IAM, providing a centrally managed cross-account access solution between AWS member accounts. AWS Landing Zones also configures audit and emergency security administrative access to AWS Landing Zone accounts from the security account to provide auditors with read-only access and security teams the ability to access accounts when necessary.

Amazon Virtual Private Cloud (VPC)

All member accounts will have a VPC initial network for the AWS account; this includes deleting the default VPC in all regions, deploying the Amazon Account Vending Machine requested network type, and network peering with the Shared Services VPC.

AWS Landing Zone Notifications

AWS Landing Zone notifications will configure Amazon CloudWatch alarms and events to send a notification on security events such as:

- Root Account Login
- Console Sign-in Failures
- API Authentication Failures

AWS Landing Zone notifications will also notify you to changes within an account to:

- Security Groups
- Network ACLs
- Amazon VPC gateways and peering connections
- Transit Gateway
- Amazon Elastic Compute Cloud (Amazon EC2) instance state
- AWS CloudTrail
- IAM policies
- AWS Config rule compliance status

This will allow your IT organization operation teams to be notified to potential compliance issues. This service also allows your IT organization the ability to take automated remediation actions from these alerts.

Amazon GuardDuty Member

Amazon GuardDuty is a group of AWS services (Route53, CloudTrail, VPC Flow logs) that analyzes traffic and can identify and alert anomalies in network communication. Amazon GuardDuty can send notifications to Amazon CloudWatch events, alerting your IT organization of the security finding. GuardDuty sends its findings to the Amazon GuardDuty master account (security account), again providing a single source of truth for monitoring information that is gathered. Amazon GuardDuty provides traceability to AWS services by monitoring communication.

Conclusion

This whitepaper has illustrated the SHI Enterprise Architecture Assessment Framework and how it will provide your IT organization with the necessary data-driven decisions to align your IT organization's enterprise schema to the AWS platform. The SHI Cloud & Innovative Solutions team has developed this framework and a comprehensive project plan from our extensive experience in IT transformation. The data gathered from an enterprise architecture assessment project will provide your IT organization with the following success criteria:

- Rationalize the current IT organization Business, Application, Data and Technology Architectures.

- Identify Current IT organization Roles and Actors and align them to Identity and Access Management Roles.
- Define and Implement AWS Enterprise Services for AWS accounts.
- Identify Security and Compliance Standards.
- Develop Governance and Auditability Architectures.
- Align AWS accounts to AWS Organizations and AWS Landing Zones to provide compliance guardrails to the AWS account resources.