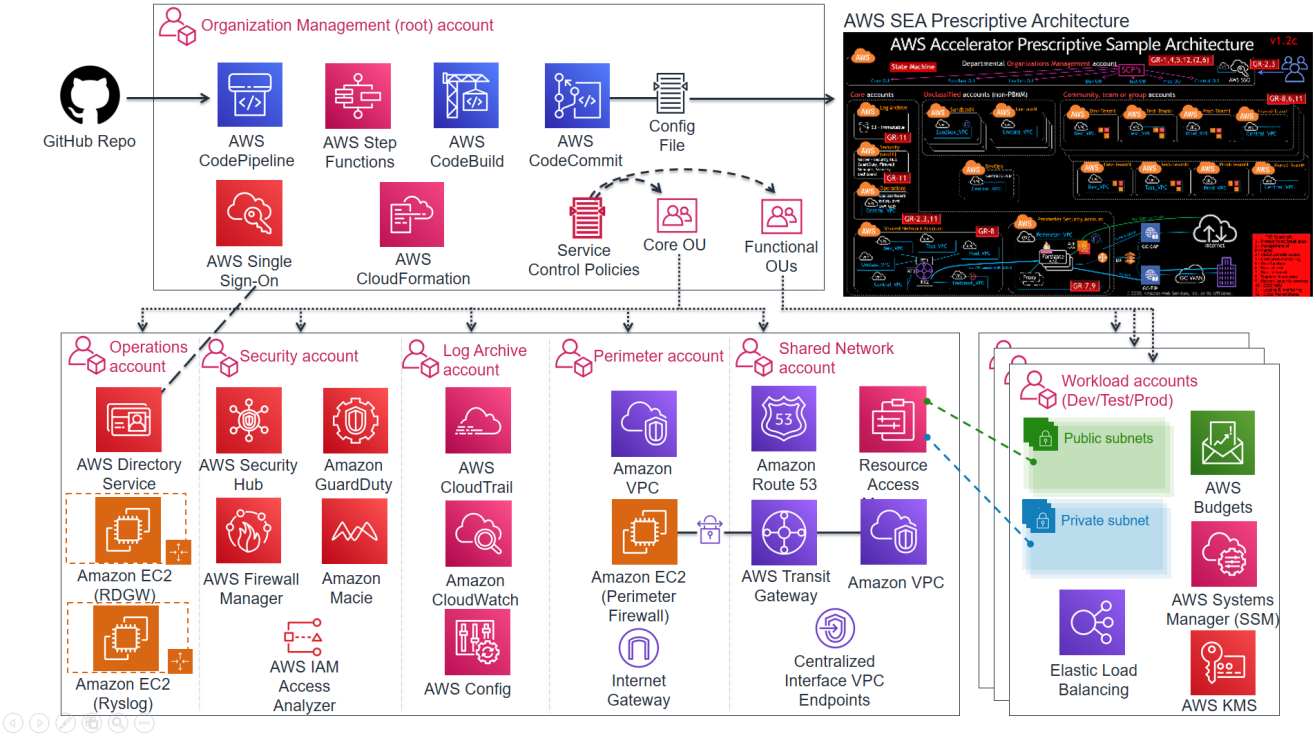
AWS Secure Environment Accelerator

The AWS Accelerator is a tool designed to help deploy and operate secure multi-account, multi-region AWS environments on an ongoing basis. The power of the solution is the configuration file that drives the architecture deployed by the tool. This enables extensive flexibility and for the completely automated deployment of a customized architecture within AWS without changing a single line of code.

While flexible, the AWS Accelerator is delivered with a sample configuration file which deploys an opinionated and prescriptive architecture designed to help meet the security and operational requirements of many governments around the world (initial focus was the Government of Canada). Tuning the parameters within the configuration file allows for the deployment of customized architectures and enables the solution to help meet the multitude of requirements of a broad range of governments and public sector organizations.

The installation of the provided prescriptive architecture is reasonably simple, deploying a customized architecture does require extensive understanding of the AWS platform.



What specifically does the Accelerator deploy and manage?

A common misconception is that the AWS Secure Environment Accelerator only deploys security services, not true. The Accelerator is capable of deploying a complete end-to-end hybrid enterprise multi-region cloud environment.

Additionally, while the Accelerator is initially responsible for deploying a prescribed architecture, it more importantly allows for organizations to operate, evolve, and maintain their cloud architecture and security controls over

time and as they grow, with minimal effort, often using native AWS tools. Customers don't have to change the way they operate in AWS.

Specifically the accelerator deploys and manages the following functionality, both at initial accelerator deployment and as new accounts are created, added, or onboarded in a completely automated but customizable manner:

Creates AWS Account

- Core Accounts - as many or as few as your organization requires, using the naming you desire. These accounts are used to centralize core capabilities across the organization and provide `Control Panel` like capabilities across the environment. Common core accounts include:
  - Shared Network
  - Operations
  - Perimeter
  - Log-Archive
  - Security-Audit
- Workload Accounts - automated concurrent mass account creation or use AWS organizations to scale one account at a time. These accounts are used to host a customer's workloads and applications.
- Scalable to 1000's of AWS accounts
- Supports AWS Organizations nested ou's and importing existing AWS accounts
- Performs 'account warming' to establish initial limits, when required
- Automatically submits limit increases, when required (complies with initial limits until increased)

Creates Networking

- Transit Gateways and TGW route tables (incl. inter-region TGW peering)
- Centralized and/or Local (bespoke) VPC's
- Subnets, Route tables, NACLs, Security groups, NATGWs, IGWs, VGWs, CGWs
- VPC Endpoints (Gateway and Interface, Centralized or Local)
- Route 53 Private and Public Zones, Resolver Rules and Endpoints, VPC Endpoint Overloaded Zones
- All completely and individually customizable (per account, VPC, subnet, or OU)
- Layout and customize your VPCs, subnets, CIDRs and connectivity the way you want
- Deletes default VPC's (worldwide)

Cross-Account Object Sharing

- VPC and Subnet sharing, including account level re-tagging (Per account security group 'replication')
- VPC attachments and peering (local and cross-account)
- Zone sharing and VPC associations
- Managed Active Directory sharing, including R53 DNS resolver rule creation/sharing
- Automated TGW inter-region peering
- Populate Parameter Store with all `user` objects to be used by customers' IaC
- Deploy and share SSM documents (4 provided out-of-box, ELB Logging, S3 Encryption, Instance Profile remediation, Role remediation)
  - customer can provide their own SSM documents for automated deployment and sharing

Identity

- Creates Directory services (Managed Active Directory and Active Directory Connectors)
- Creates Windows admin bastion host auto-scaling group
- Set Windows domain password policies
- Set IAM account password policies
- Creates Windows domain users and groups (initial installation only)
- Creates IAM Policies, Roles, Users, and Groups
- Fully integrates with and leverages AWS SSO for centralized and federated login

Cloud Security Services

- Enables and configures the following AWS services, worldwide w/central designated admin account:
  - Guardduty w/S3 protection
  - Security Hub (Enables designated security standards, and disables individual controls)
  - Firewall Manager
  - CloudTrail w/Insights and S3 data plane logging
  - Config Recorders/Aggregator
  - Conformance Packs and Config rules (95 out-of-box NIST 800-53 rules, 2 custom rules, customizable per OU)
  - Macie
  - IAM Access Analyzer
  - CloudWatch access from central designated admin account (and setting Log group retentions)

Other Security Capabilities

- Creates, deploys and applies Service Control Policies
- Creates Customer Managed KMS Keys (SSM, EBS, S3)
- Enables account level default EBS encryption and S3 Block Public Access
- Configures Systems Manager Session Manager w/KMS encryption and centralized logging
- Creates and configures AWS budgets (customizable per ou and per account)
- Imports or requests certificates into AWS Certificate Manager
- Deploys both perimeter and account level ALB's w/Lambda health checks, certificates and TLS policies
- Deploys & configures 3rd party firewall clusters and management instances w/vendor best practices and sample security policies, w/automated TGW ECMP BGP tunnel standup (leverages marketplace)
- Protects Accelerator deployed and managed objects
- Sets Up SNS Alerting topics (High, Medium, Low, Blackhole priorities)
- Deploys CloudWatch Log Metrics and Alarms
- Deploys customer provided custom config rules (1 provided out-of-box, No EC2 Instance Profile)

Centralized Logging and Alerting

- Deploys an rsyslog auto-scaling cluster behind a NLB, all syslogs forwarded to CloudWatch Logs
- Centralized access to "Cloud Security Service" Consoles from designated AWS account
- Centralizes logging to a single centralized S3 bucket (enables, configures and centralizes)
  - VPC Flow logs w/Enhanced metadata fields (also sent to CWL)
  - Organizational Cost and Usage Reports
  - CloudTrail Logs including S3 Data Plane Logs (also sent to CWL)
  - All CloudWatch Logs (includes rsyslog logs)
  - Config History and Snapshots
  - Route 53 Public Zone Logs (also sent to CWL)
  - GuardDuty Findings
  - Macie Discovery results
  - ALB Logs
  - SSM Session Logs (also sent to CWL)
  - Resolver Query Logs (also sent to CWL)

Relationship with AWS Landing Zone Solution (ALZ)

The ALZ is an AWS Solution designed to deploy a multi-account AWS architecture for customers based on best practices and lessons learned from some of AWS' largest customers. The AWS Accelerator draws on design patterns from the Landing Zone, and re-uses several concepts and nomenclature, but it is not directly derived from it, nor does it leverage any code from the ALZ. The initial versions of the AWS Accelerator presupposed the existence of an AWS Landing Zone Solution in the AWS Organization; this requirement has since been removed as of release `v1.1.0`.

The Accelerator is now a completely standalone solution.

Relationship with AWS Control Tower

AWS Control Tower is the successor to the ALZ, but offered as an AWS managed service.

When appropriate, it is envisioned that the AWS Accelerator will add the capability to be deployed on top of AWS Control Tower, as we initially allowed with the ALZ.

Accelerator Deployment Process (Summary)

This summarizes the installation process, the full installation document can be found in the documentation section below.

- Create a config.json (or config.yaml) file to represent your organizations requirements (several samples provided)
- Create a Secrets Manager Secret which contains a GitHub token that provides access to the Accelerator code repo
- Create a unique S3 input bucket and place your config.json and any additional custom config files in the bucket
- Download and execute the latest installer CloudFormation template in your root accounts preferred 'primary' / 'home' region
- Wait for:
  - CloudFormation to deploy and start the Code Pipeline (~5 mins)
  - Code Pipeline to download the Accelerator codebase and install the Accelerator State Machine (~20 mins)
  - The Accelerator State Machine to finish execution (~1.5 hrs)
- Perform required manual follow-up activities (configure AWS SSO, set firewall passwords, etc.)
- When required:
  - Use AWS Organizations to create new AWS accounts, which will automatically be guardrailed by the Accelerator
  - Update the config file in CodeCommit and run the Accelerator State Machine (~25 min) to:
    * deploy, configure and guardrail multiple accounts at the same time
    * change Accelerator configuration settings

Documentation

- Accelerator Installation and Upgrade Guide

- Link to Accelerator releases and change history
- Sample configuration files and customization details
- Chart containing details as to WHAT we do and WHERE we support it (regions, accounts, etc.)
- Accelerator central logging bucket structures
- Unofficial Accelerator Roadmap (GitHub projects) - *Please upvote desired features*

- Accelerator Operations/Troubleshooting Guide

- Accelerator Basic Operation and Frequently Asked Questions (FAQ)

- Accelerator Developer Guide

- Accelerator release process (AWS Internal)

- Contributing & Governance Guide

- Prescriptive PBMM Architecture Design Document (Early Draft)

- AWS PBMM architecture sample diagrams

Note: A ZIP file containing a PDF version of most documentation can be found here.

Go to Accelerator Table of Contents