適用於金融服務的雲端安全最佳實踐

HC Lo | Solutions Architect, AWS

Annie Lin | Territory Business Development Manager, AWS

Justine Peng | Lead Development Representative, AWS

2020/06/04

Agenda

- Top ten security things to do
- How to set up and govern a new, secure multi-account AWS environment

Security is Job Zero at AWS

Designed for Security

Constantly Monitored

Highly Automated

Highly Available

Highly Accredited

“Why did we pick AWS?”

“The financial services industry attracts some of the worst cyber criminals. We have worked closely with AWS to develop a security model, which we believe enables us to operate more securely in the public cloud than we can in our own data centers.”

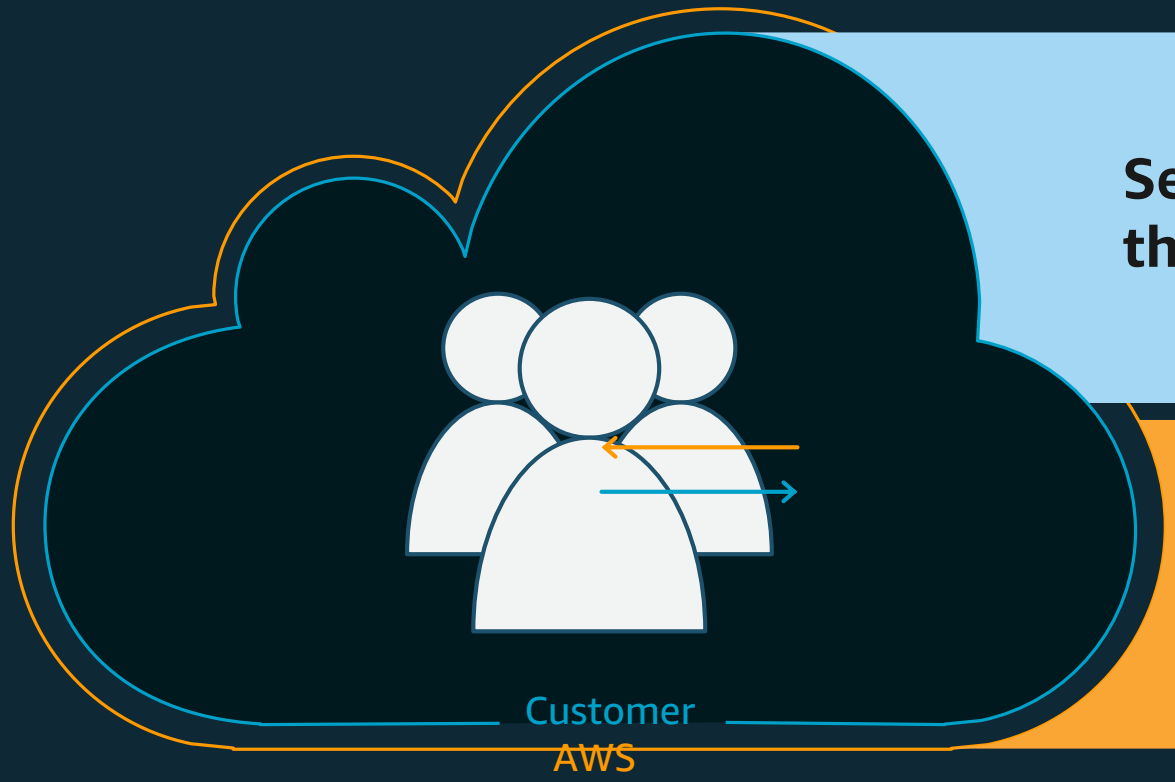
-- Rob Alexander, Capital One's Chief Information Officer

“We think security and identity and access management done correctly can empower our engineers to focus on products within clear and trusted walls, and that’s why we implemented an auditable self-service security foundation with AWS IAM.”

-- Rob Witoff, Coinbase Director

aws

Shared responsibility model



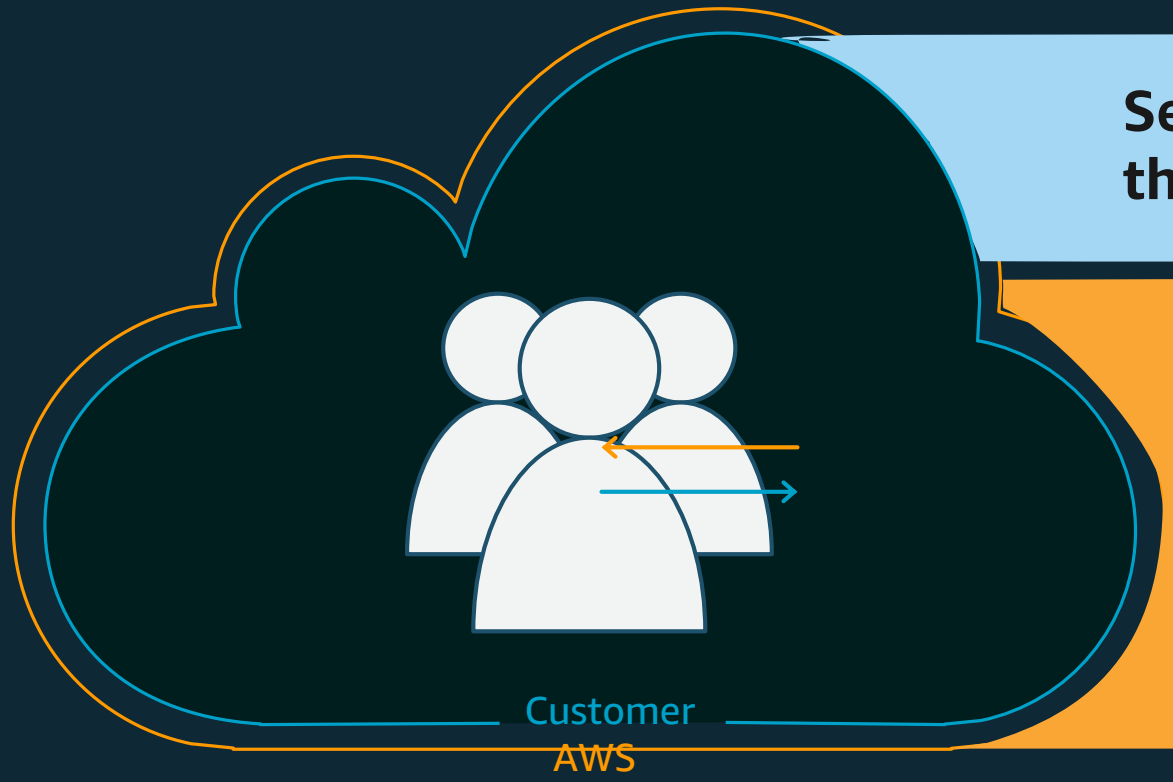
Security IN the Cloud

Customer is responsible for proper configuration and operation of cloud services

Security OF the Cloud

AWS is responsible for the infrastructure that runs all of cloud services and providing rich security capabilities

Dynamic over time



Security IN the Cloud

Customers using higher level services plus security services get more and more “for free”

Security OF the Cloud

AWS is constantly raising the bar in service capabilities and security ease of use

AWS CISO Steve Schmidt: The top ten list

Ten places your security group should spend time

1. Accurate account info
2. Use MFA
3. No hard-coding secrets
4. Limit security groups
5. Intentional data policies
6. Centralize AWS CloudTrail logs
7. Validate IAM roles
8. Take action on GuardDuty findings
9. Rotate your keys
10. **Being involved in dev cycle**

Ten places your security team should spend time

1. Accurate account info
2. Use MFA
3. No hard-coded secrets
4. Limit Security Groups
5. Intentional data policies
6. Centralize AWS CloudTrail logs
7. Validate IAM roles
8. Take action on security findings
9. Regularly rotate credentials (keys)
10. Get involved in the dev cycle

aws

Ten places your security team should spend time

1. Accurate account info
2. Use MFA
3. No hard-coded secrets
4. Limit Security Groups
5. Intentional data policies
6. Centralize AWS CloudTrail logs
7. Validate IAM roles
8. Take action on security findings
9. Regularly rotate credentials (keys)
10. Get involved in the dev cycle

aws

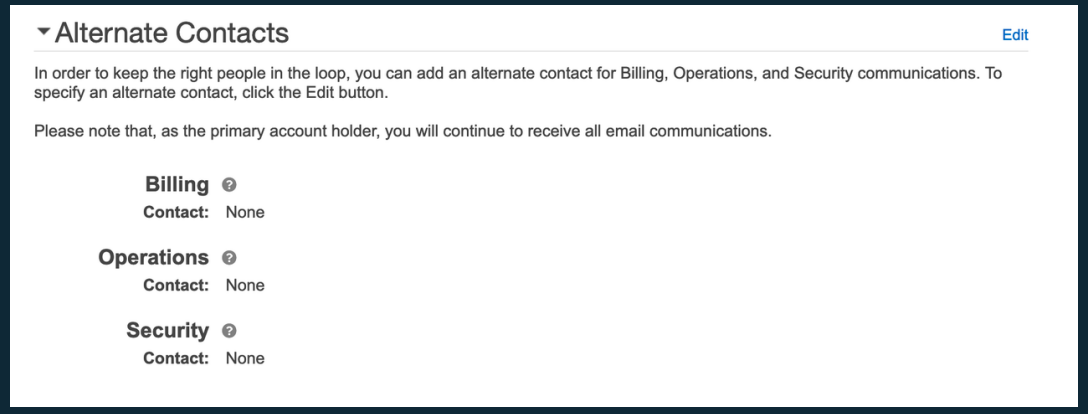
Five of ten: the quick (or at least, start now) list

1. Accurate account info
2. Use MFA
3. No hard-coded secrets
4. Limit Security Groups
5. Intentional data policies
6. Centralize AWS CloudTrail logs
7. Validate IAM roles
8. Take action on security findings
9. Regularly rotate credentials (keys)
10. Get involved in the dev cycle

aws

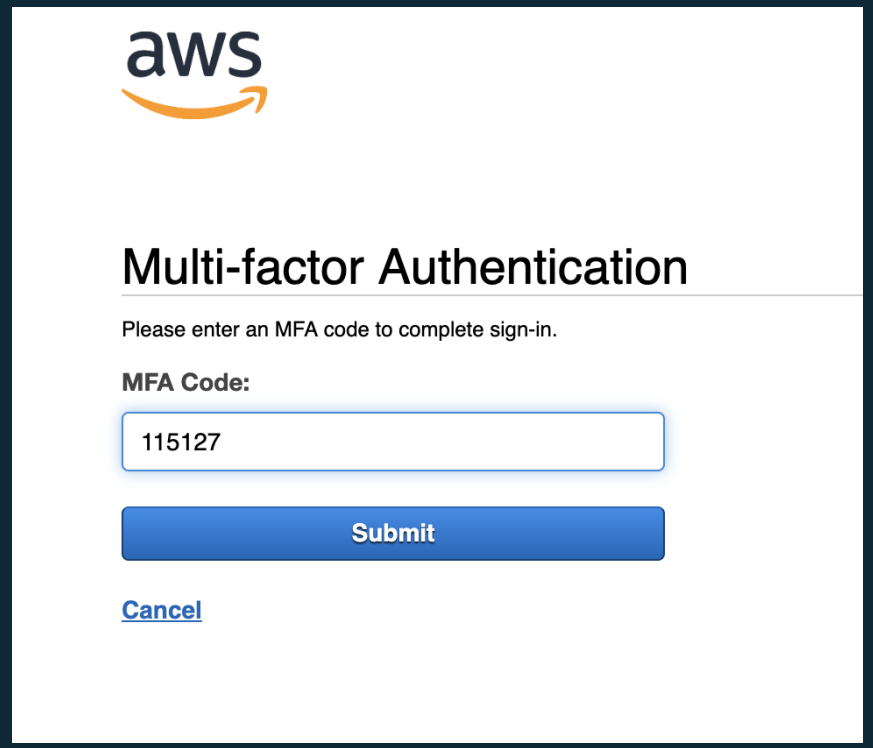
Let's get started!

1. Accurate account info



- Correct email and phone number
- Use a root principal email address that reaches more than one person!
  - At a minimum use a distribution list
  - Better: a shared inbox with forwarding rules
  - Carefully control access to the inbox (email)!
- Activate the security contact email, using the same approach
- Use these all addresses as a detection point, billing alerts are a good thing

2. Use MFA



- MFA adds an extra layer of protection on top of name and password for root, interactive IAM users
  - Virtual MFA devices ✓
  - U2F security key
  - Hardware MFA device
- Identity federation changes the approach, but not the best practice
  - Use MFA at your identity provider
  - Even works with AWS CLI (version 2)

4. Limit Security Groups

A surprisingly common mistake customers make is not using Security Groups effectively

Don’t assign public IPs to EC2 instances for convenience!

- Use bastions hosts or Client VPN to connect
- Or Systems Manager Session Manager (and VPC Endpoints) for highly secure interactive access to EC2 instances
- If you must, lock down access to very limited number of IPs using Security Groups

aws

4. Limit Security Groups (continued)

For public IPs, limit 0/0 traffic to “hardened” endpoints

- No ports 22 (SSH), 3389 (RDP), 3306 (MySql), 9300 (Elasticsearch), etc.
- Scan your own public IPs

Even inside your VPC, lock down inter-instance traffic to what is absolutely required

- Load balancer to web tier
- Web tier to application logic tier
- App tier to data tier

Use Firewall Manager to manage centrally Security Groups across VPCs, accounts

aws

8. Take action on service findings

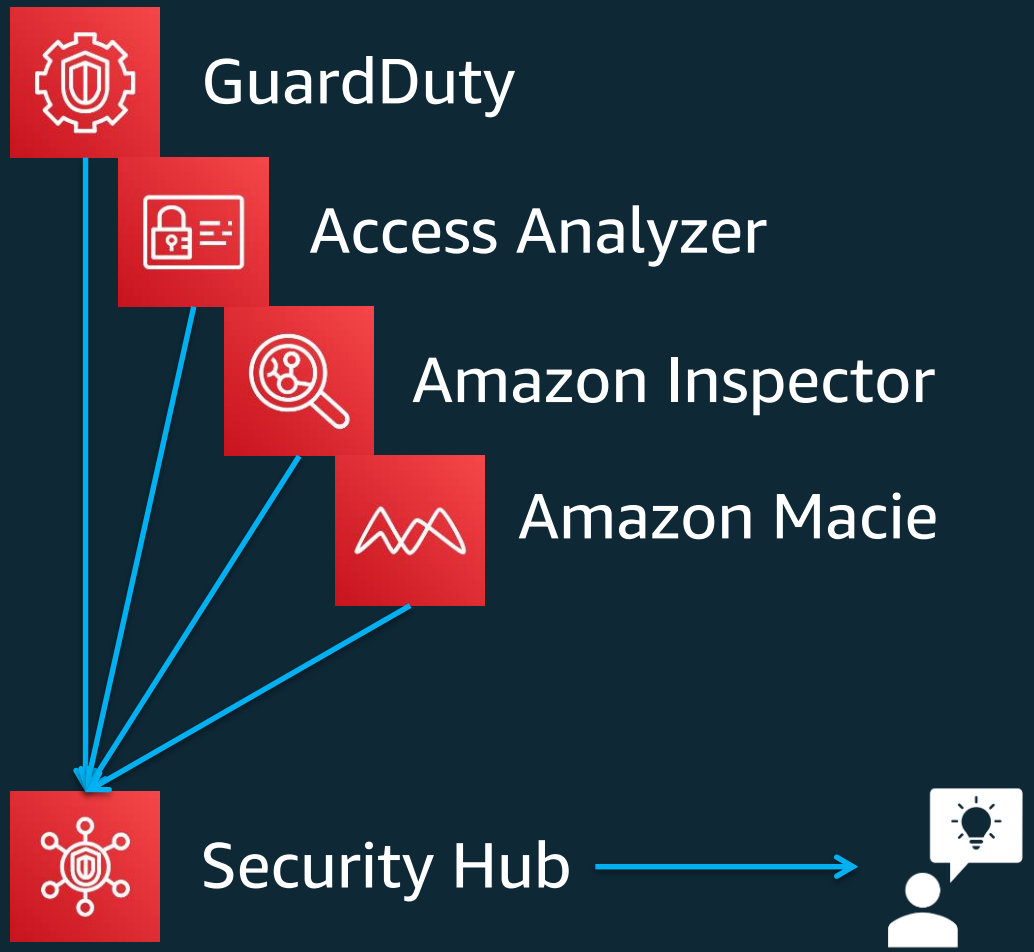
Findings come from more than just Guard Duty!

Security Hub is there – use it!

Each of the findings that come out of these services can tell you different things about your infrastructure

If one of these services tells you something you too often … or you don't want to hear… be very careful about suppressing warnings!

8. Take action on service findings



Findings come from more than just Guard Duty!

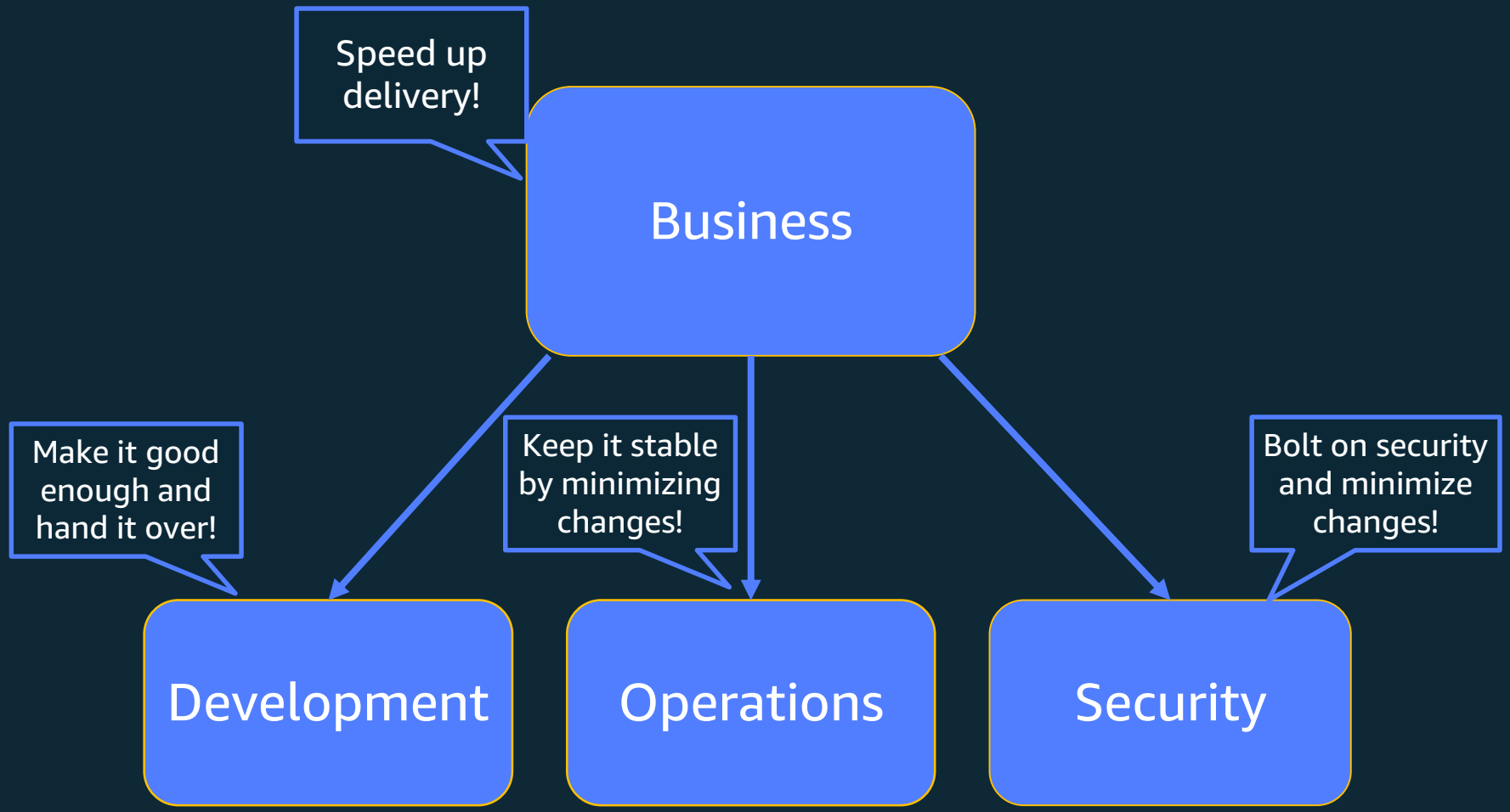
Security Hub is there – use it!

Each of the findings that come out of these services can tell you different things about your infrastructure

If one of these services tells you something you too often … or you don't want to hear… be very careful about suppressing warnings!

This is the way it should look

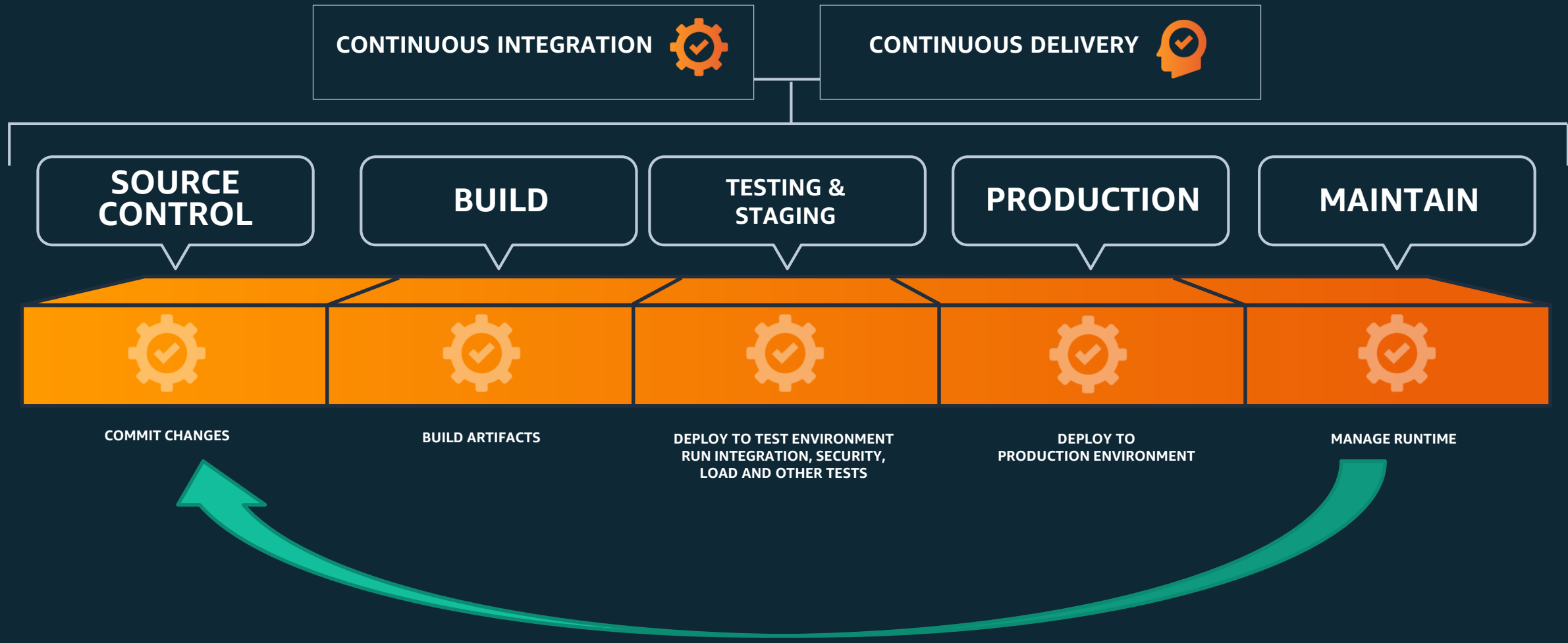
10. Get involved in dev cycle



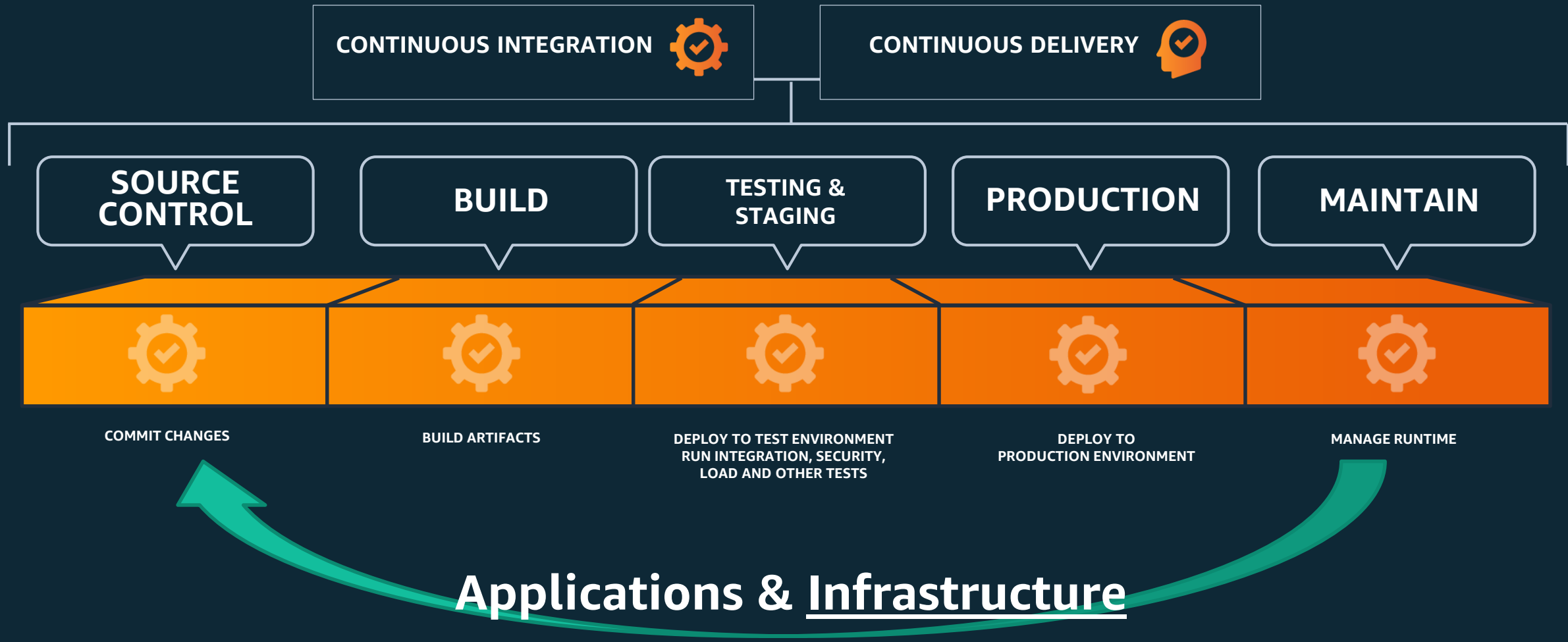
DevSecOps

The business

The modern automated software factory



Security must be built in to the CI/CD Pipeline



Govern multi-account AWS environments at scale

Account models

One account

1,000s of accounts

Balancing the needs of builders and central cloud IT

Builders: Stay agile

Innovate with the speed and agility of AWS

Cloud IT: Establish governance

Govern at scale with central controls

More innovation, greater agility, with control

Agility

Experiment

Be productive
Empower distributed teams
Self-service access

Respond quickly to change

Don't choose between Agility or Control

You need and want both

Governance

Enable

Provision

Operate

Secure & Compliant

Operations & Spend Management

aws

AWS management and governance services

Security and IAM

Enable

- AWS Control Tower
- AWS Organizations
- AWS Budgets
- AWS License Manager
- AWS Well-Architected Tool

Provision

- AWS CloudFormation
- AWS Service Catalog
- AWS OpsWorks
- AWS Marketplace

Operate

- Amazon CloudWatch
- AWS CloudTrail
- AWS Config
- AWS Systems Manager
- AWS Cost and Usage Report
- AWS Cost Explorer

BUSINESS AGILITY + GOVERNANCE CONTROL

Automation

AWS Control Tower: Easiest way to set up and govern AWS at scale

What is a “landing zone”

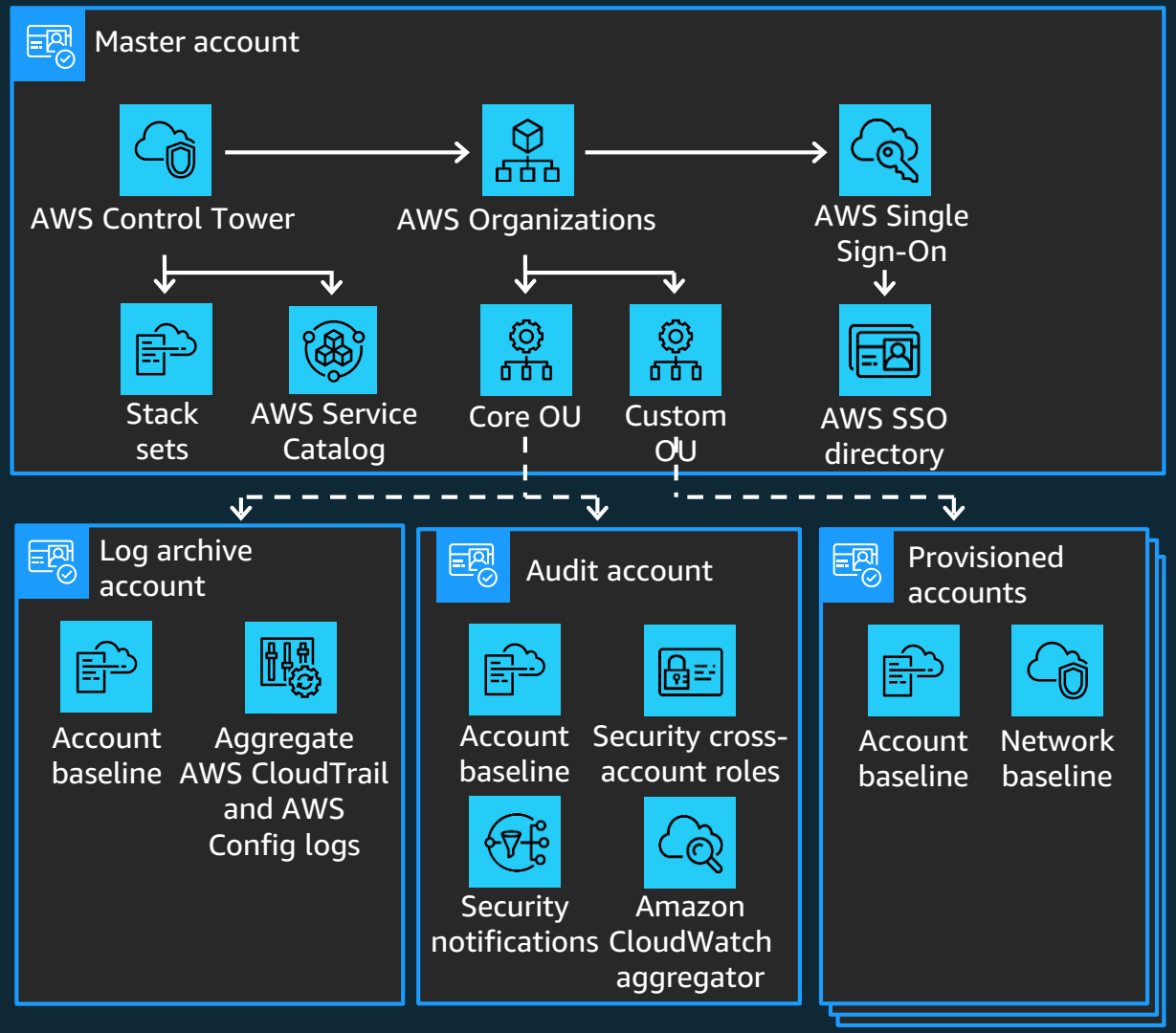
- A configured, secure, scalable, multi-account (multiple resource containers) AWS environment based on AWS best practices
- A starting point for net new development and experimentation
- A starting point for migrating applications
- An environment that allows for iteration and extension over time

Enable governance

Enable

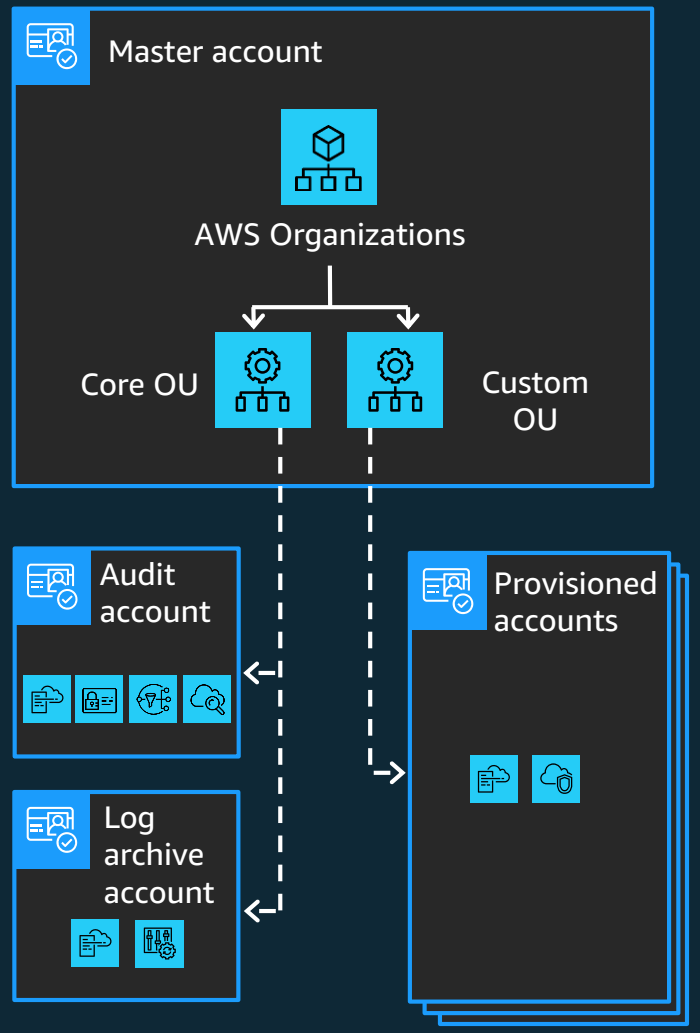
aws

Set up an AWS landing zone



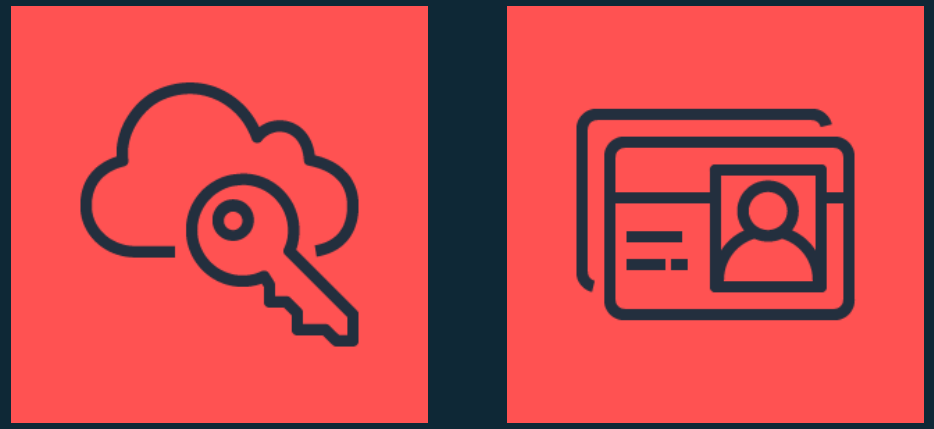
- Landing zone - a preconfigured, secure, scalable, multi-account AWS environment based on best practice blueprints
- Multi-account management using AWS Organizations
- Identity and federated access management using AWS SSO
- Centralized log archive using AWS CloudTrail and AWS Config
- Cross-account audit access using AWS SSO and AWS IAM
- End user account provisioning through AWS Service Catalog
- Centralized monitoring and notifications using Amazon CloudWatch and Amazon SNS

Multi-account architecture



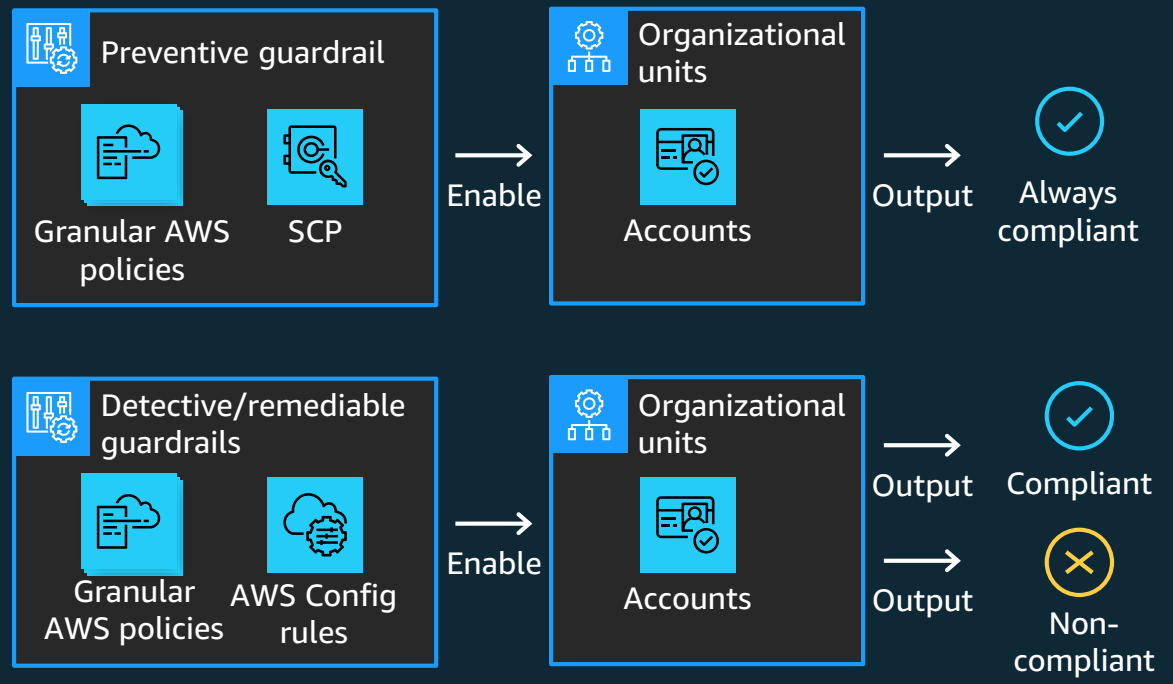
- Master account: designation of your existing account to create a new organization. Also your master payer account
- Organization consists of 2 OUs with pre-configured accounts -
  - Core OU: AWS Control Tower-created accounts, i.e., Audit account and Log archive account
  - Custom OU: Your provisioned accounts

Centralize identity and access



- AWS SSO provides default directory for identity
- AWS SSO also enables federated access management across all accounts in your organization
- Preconfigured groups (e.g., AWS Control Tower administrators, auditors, AWS Service Catalog end users)
- Preconfigured permission sets (e.g., admin, read-only, write)

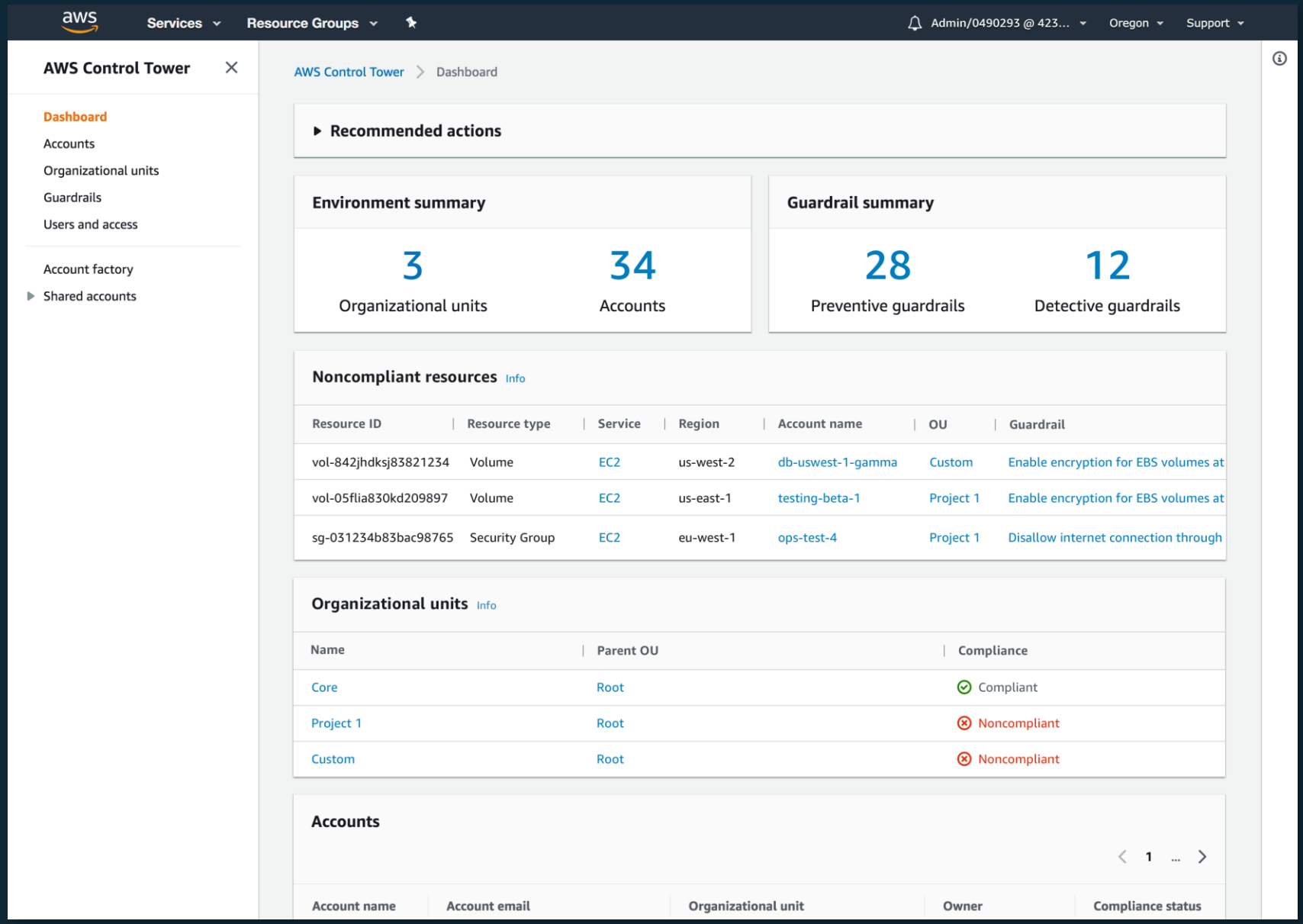
Establish guardrails



- Guardrails are preconfigured governance rules for security, compliance, and operations
- Expressed in plain English to provide abstraction over granular AWS policies
- Preventive guardrails: prevent policy violations through enforcement; implemented using AWS CloudFormation and SCPs
- Detective guardrails: detect policy violations and alert in the dashboard; implemented using AWS Config rules
- Mandatory and strongly recommended guardrails for prescriptive guidance
- Easy selection and enablement on organizational units

aws

Dashboard for oversight



AWS Control Tower capabilities

Account Management	• Framework for creating and baselining a multi-account environment using AWS Organizations • Initial multi-account structure including security, audit, & shared service requirements • An account vending machine that enables automated deployment of additional accounts with a set of managed and monitored security baselines • A management console that shows compliance status of accounts • The ability to apply AWS best practice guardrails and Blueprints to accounts at account creation • The ability to detect and report on any drift/changes that have occurred that deviate from initial configuration options
Identity & Access Management	• User account access managed through AWS SSO federation • Integration options with other 3rd party SSO providers (PING/OKTA, Azure AD – native support) • Cross-account roles enable centralized management
Security & Governance	• Multiple accounts enable separation of duties • Initial account security and AWS Config rules baseline • Network baseline

Case Study: Deutsche Börse Group

“We started using AWS Control Tower to speed up our AWS account creation with its 'Account Factory.’”

“It gives us an easy way to create accounts across our organization and establish guardrails to enforce or check for policy compliance. Now our teams can quickly create accounts with pre-configured permissions to enable us to perform audit or administrative actions.”

-- Christian Tueffers, Deutsche Börse Group Cloud Architect

Summary

Ten places your security team should spend time

1. Accurate account info
2. Use MFA
3. No hard-coded secrets
4. Limit Security Groups
5. Intentional data policies
6. Centralize AWS CloudTrail logs
7. Validate IAM roles
8. Take action on security findings
9. Regularly rotate credentials (keys)
10. Get involved in the dev cycle

aws

Five of ten: the quick (or at least, start now) list

1. Accurate account info
2. Use MFA
3. No hard-coded secrets
4. Limit Security Groups
5. Intentional data policies
6. Centralize AWS CloudTrail logs
7. Validate IAM roles
8. Take action on security findings
9. Regularly rotate credentials (keys)
10. Get involved in the dev cycle

aws

AWS Control Tower key features

Automated landing zone with best practice blueprints

Built-in identity and access management

Guardrails for policy management

Preconfigured log archive and audit access to accounts

Account factory for account provisioning

Built-in monitoring and notifications

Dashboard for visibility and actions

Automatic updates

Thank you!