Essential Guardrails for AWS Organizations

Cassandra Young

The Diana Initiative - August 2022

If everything's under control, you're going too slow

Mario Andretti

Cassandra Young (aka muteki)

Senior Scientist, Cloud Security
@ Security Risk Advisors

Master's of Computer Science
@ University of Pennsylvania

Director & Meet-a-Mentor Lead
@ Blue Team Village

she/her; lives for international travel,
aurora chasing & scuba diving,
Star Trek & jigsaw puzzles

Twitter: @muteki_rtw
GitHub: github.com/muteki-apps
LinkedIn: linkedin.com/in/cassandray

Agenda

- Cloud Overview & Security Design
- AWS Accounts & Account Security
- AWS Organizations & Organizational Units
- Service Control Policies
- Logging
- Control Tower
- Questions from the audience

Slides & Activities: https://github.com/muteki-apps/Security-Training/


Cloud Overview
& Security Design

The Shared Responsibility Model

Varying levels of access and control requires flexible thinking when designing for security

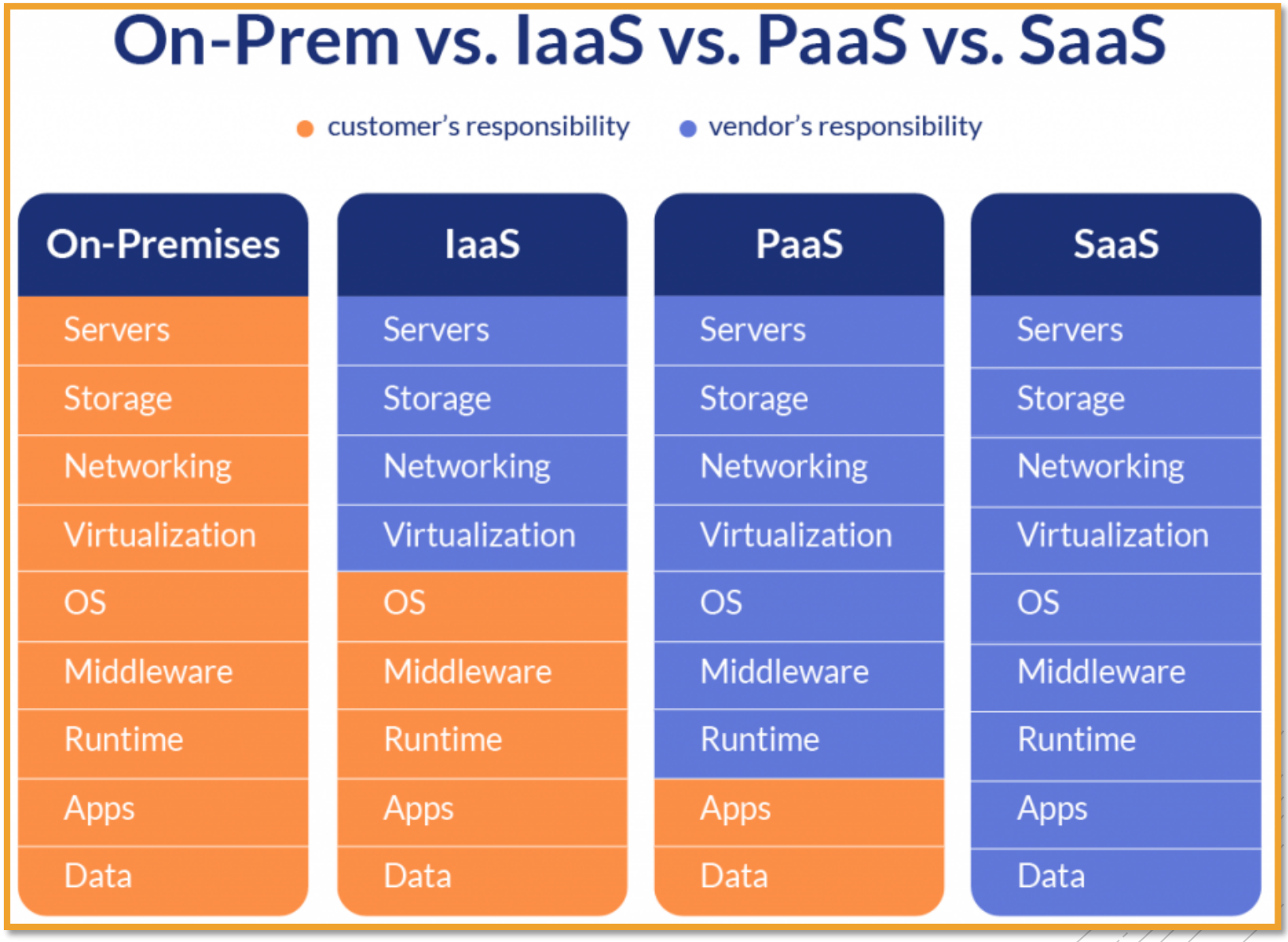


Image Source: https://layots.com/iaas-vs-paas-vs-saas-things-you-need-to-know-its-difference/

Designing Cloud Security

What areas should a robust cloud security plan cover?

Cloud Platform Design

- Architecture (Account, Infrastructure & Application)
- Network (Layout & Traffic Inspection)
- Identity & Access Management
- Secure CI/CD Pipeline & Automation

Service Protection & Hardening

- Logging & Monitoring
- Data Protection
- Service-level Hardening
- Resilience & Disaster Recovery

Policy & Process

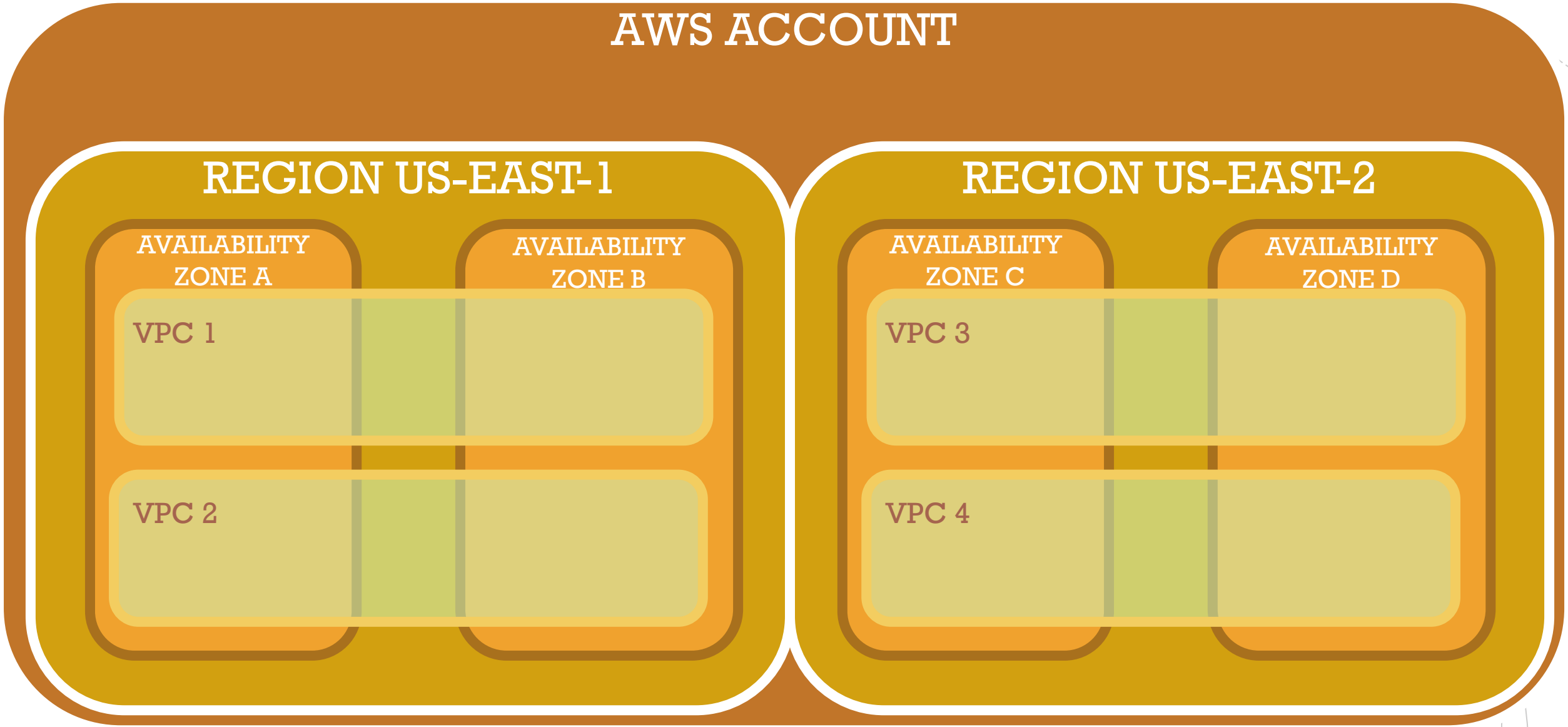
- Governance
- Vulnerability Management
- Incident Response

AWS Accounts & Account Security

AWS Accounts: Overview

an AWS account is the basic container for all AWS resources you create as a customer, ie. virtual machines, databases and storage accounts

- Accounts represent a default (“natural”) security, access and billing boundary and resource isolation
- Most resources within an account can be further subdivided by region, and segmented using Virtual Private Clouds (VPCs), network controls, IAM Roles & Policies, & more…


AWS ACCOUNT
REGION US-EAST-1
AVAILABILITY ZONE A
AVAILABILITY ZONE B
VPC 1
VPC 2
REGION US-EAST-2
AVAILABILITY ZONE C
AVAILABILITY ZONE D
VPC 3
VPC 4

AWS Accounts: Security

ROOT USER ACCOUNTS

- Root user is the primary administrator of an AWS account
- Secure access to email account associated with root user
- Enable MFA, secure & restrict access to MFA method
- Break-glass users must be tightly controlled and monitored

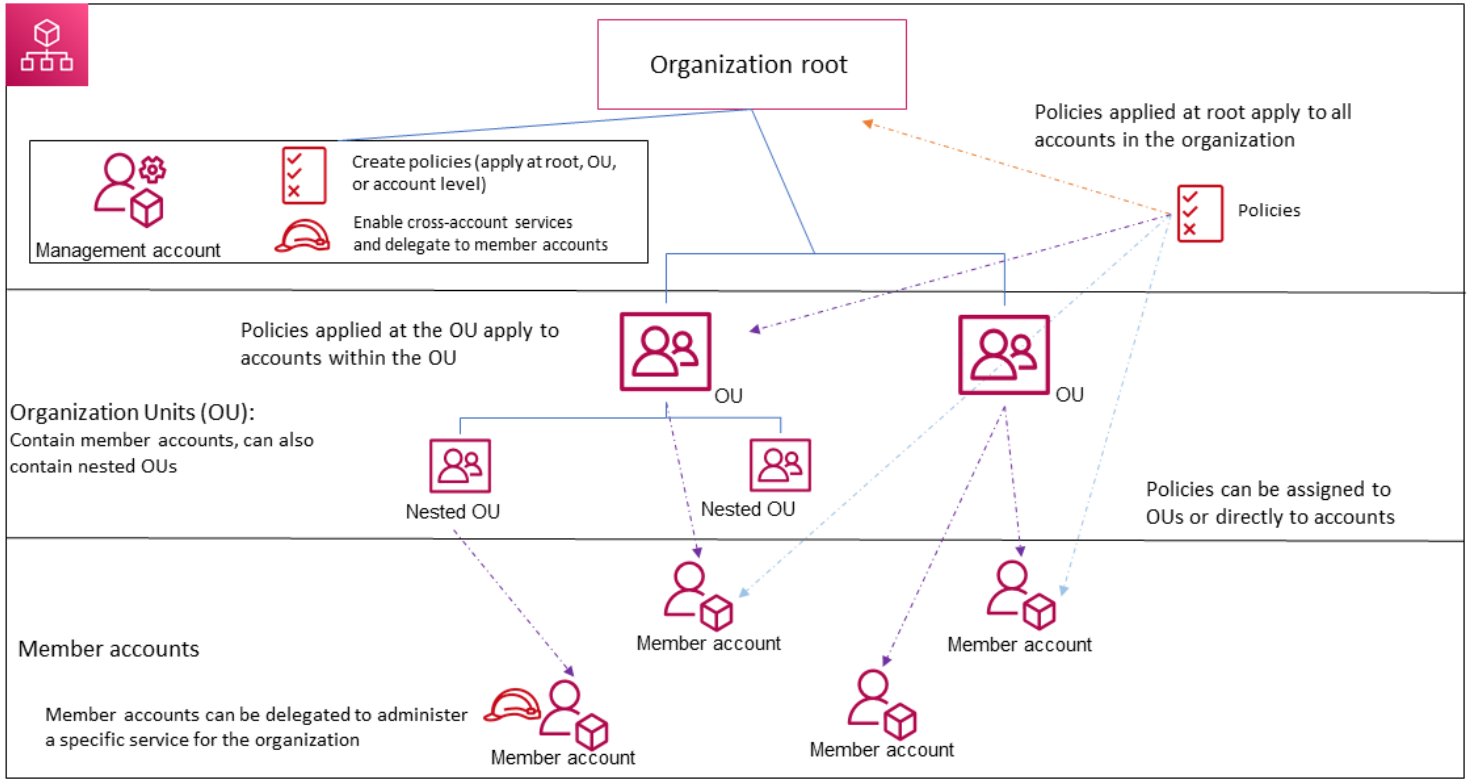
  This account should be enabled but secured and never used!!

BILLING ALERTS

- Set up billing alerts immediately!
- Configure email and/or other service notifications to trigger when charges exceed a set threshold

AWS Organizations & Organizational Units

AWS Organizations: Overview



AWS Organizations enables companies to scale and centrally manage secure multi-account environments by grouping accounts within Organizational Units (OUs), and scoping access to services using Service Control Policies (SCPs)

AWS Organizations: Overview

Why use multiple accounts?

- Logical grouping of accounts by purpose
- Can scope permissions of and within each account
- Multi-account environment reduces blast radius
- Centrally manage billing & billing alerts
- Aligns with best practices for compliance

AWS Organizations: Securing the Management Account

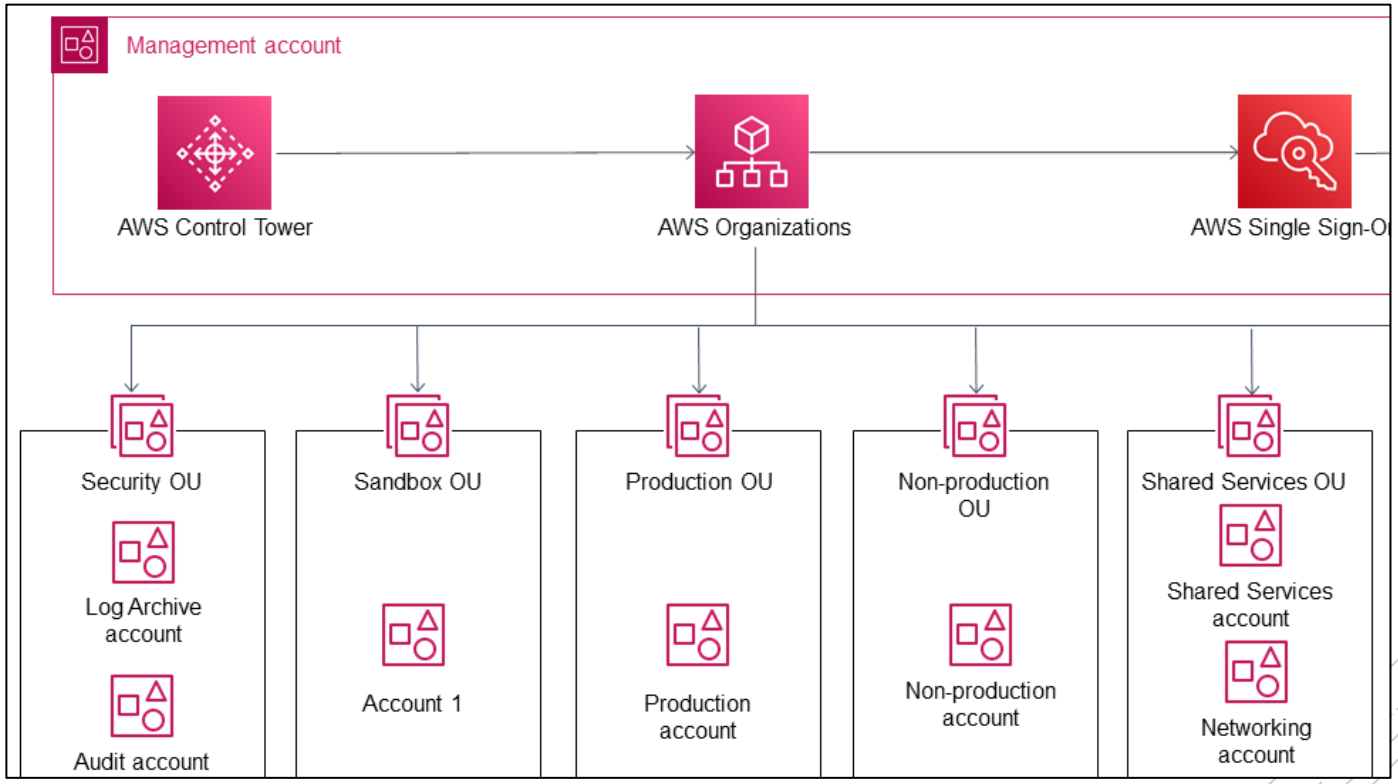
The AWS Organizations Management Account is "Root"!

- Use management account only for tasks that require it
- Use a group/shared email account for the management account's root user & secure access to that account
- Use a complex password for the account's root user
- Enable MFA for root user credentials & store securely
- Review and keep track of who has access, and when they use it

Organizational Units: Overview

OUs represent logical groupings of accounts

- Group accounts by function or common controls
- Centrally deploy and run supported services from within one functionally aligned account
- Delegate service administration to member accounts



https://aws.amazon.com/organizations/getting-started/best-practices/

Organizational Units: Security

SECURITY TOOLING

- Use to centrally operate security services, monitor AWS accounts, and automate security alerting and response
- Centralize administration of Security Hub, GuardDuty, Config and other services that support Orgs integration

LOG ARCHIVE

- Use to store and archive centralized CloudTrail, VPC, and other logs within S3 buckets
- For log export (ie, SIEM ingestion), can scope required external access within only one account

https://docs.aws.amazon.com/organizations/latest/userguide/orgs_integrate_services_list.html

Organizational Units: Infrastructure

NETWORK

- Isolates the networking services, configuration, and operation from the individual application workloads, security, and other infrastructure
- Use to run inspection VPC, firewall, network inspection systems (IDS/IPS)

SHARED SERVICES

- Runs the services that multiple applications and teams use to deliver their outcomes
- Use to centralize services such as Systems Manage, AWS Managed Microsoft AD, and for delegated admin of IAM Identity Center

Organizational Units: Workloads

PRODUCTION OU

- Application accounts should be used to isolate individual app deployments
- Accounts within production OU should use properly scoped IAM permissions and external access

NON-PRODUCTION OU(S)

- Application accounts should be used to isolate individual app deployments
- Logical isolated non-production apps allow for different levels of IAM permissions and

Service Control Policies

Service Control Policies (SCPs)

Service Control Policies (SCPs) are used to enforce permissions guardrails at the OU and account level

- Used to define the *maximum available* permissions
- Prevent or remediate high-risk configurations at the organizational level
- Restrict access to unapproved AWS services
- Apply to individual accounts, OUs, and nested OUs

SCP Example: Region Restriction

WHAT IS IT?

- Restricts all region-specific resource creation to specified regions, blocking attempts to create outside of that region

WHY DO IT?

- Prevents accidental use of unapproved regions
- In case of compromise, prevents resource creation in unused regions

SCP Example: Region Restriction

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "NotAction": [
                "a4b:*",
                "acm:*",
                "...",
                "waf:*",
                "wafv2:*",
                "wellarchitected:*"
            ],
            "Resource": "*",
            "Effect": "Deny",
            "Condition": {
                "StringNotEquals": {
                    "aws:RequestedRegion": [
                        "us-east-1"
                    ]
                }
            }
        }
    ]
}
```

- Define actions effected
- Specify services/resources
- Action to take (allow/deny)
- Condition
  - Use to scope further
  - Combine with "Deny" to exclude any condition not explicitly stated

SCP Example:
Lock Down Network Configuration

WHAT IS IT?

- SCPs that block IAM users from changing network configurations, adding internet access to VPCs that don't have it, etc.

WHY DO IT?

- Prevent accidental expose of resources residing within VPC designed for restricted access
- Reduces methods of data exfiltration or remote access for malware, C2s from within compromised account

SCP Example:
Lock Down VPC Internet Access

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ec2:AttachInternetGateway",
                "ec2:CreateInternetGateway",
                "ec2:AttachEgressOnlyInternetGateway",
                "ec2:CreateVpcPeeringConnection",
                "ec2:AcceptVpcPeeringConnection"
            ],
            "Resource": "*",
            "Effect": "Deny"
        },
        {
            "Action": [
                "globalaccelerator:Create*",
                "globalaccelerator:Update*"
            ],
            "Resource": "*",
            "Effect": "Deny"
        }
    ]
}
```

Want to extend this?

- Add condition to allow admins to change settings
- Include/exclude services to change resources affected, scope of policy
- Longer version of this can lock down all network settings to only admins

SCPs: Examples

- Prevent Users from Disabling AWS CloudTrail
- Deny access to AWS based on the requested AWS Region
- Require a tag on specified created resources
- Prevent users from disabling/changing AWS Config
- Prevent Creation of New IAM Users/ Access Keys
- Prevent non-public VPCs from getting public internet access

https://docs.aws.amazon.com/organizations/latest/userguide/orgs_manage_policies_scps_examples_general.html#example-scp-deny-region
https://asecure.cloud/a/scp_deny_iam_user_creation_w_exception/

Logging

Logging: CloudTrail & CloudWatch

CLOUDTRAIL FOR ORGANIZATIONS

- Centralize logging across all accounts with Organizational CloudTrail
- Export to Log Archive account for SIEM integration
- Monitor and enforce CloudTrail compliance using SCPs and AWS Config
- Use to monitor Organizations activity ie. account creation, deletion, updates to OUs, etc.

CLOUDWATCH EVENTS

- Share CloudWatch Events across AWS accounts
- Raise alerts on specified Organizations actions

Logging & Monitoring

Centralize within Log Archive account:

- CloudTrail
- Cloudwatch
- VPC Flow Logs
  - Track network interface traffic
  - Enable in all VPCs and export
- Service-level logging:
  - Most AWS services can be configured to log additional service-specific behavior
  - Any logs that need to be exported externally should be saved to S3 in the Log Archive account

Extending Organizations with Control Tower

Extending Organizations: Control Tower

Control Tower extends AWS Organizations, allowing you to orchestrate capabilities of multiple underlying services to create a Landing Zone foundational AWS environment for seamless account creation and governance

- Why Control Tower?
  - Allows for more granular control over AWS environments
  - Use to manage multiple AWS Organizations
  - Offers the ability to templatize new account creation

Extending Organizations: Control Tower

Landing zone

- A well-architected, multi-account environment based on security and compliance best practices. Can scale to fit the needs of an enterprise of any size.

Guardrails

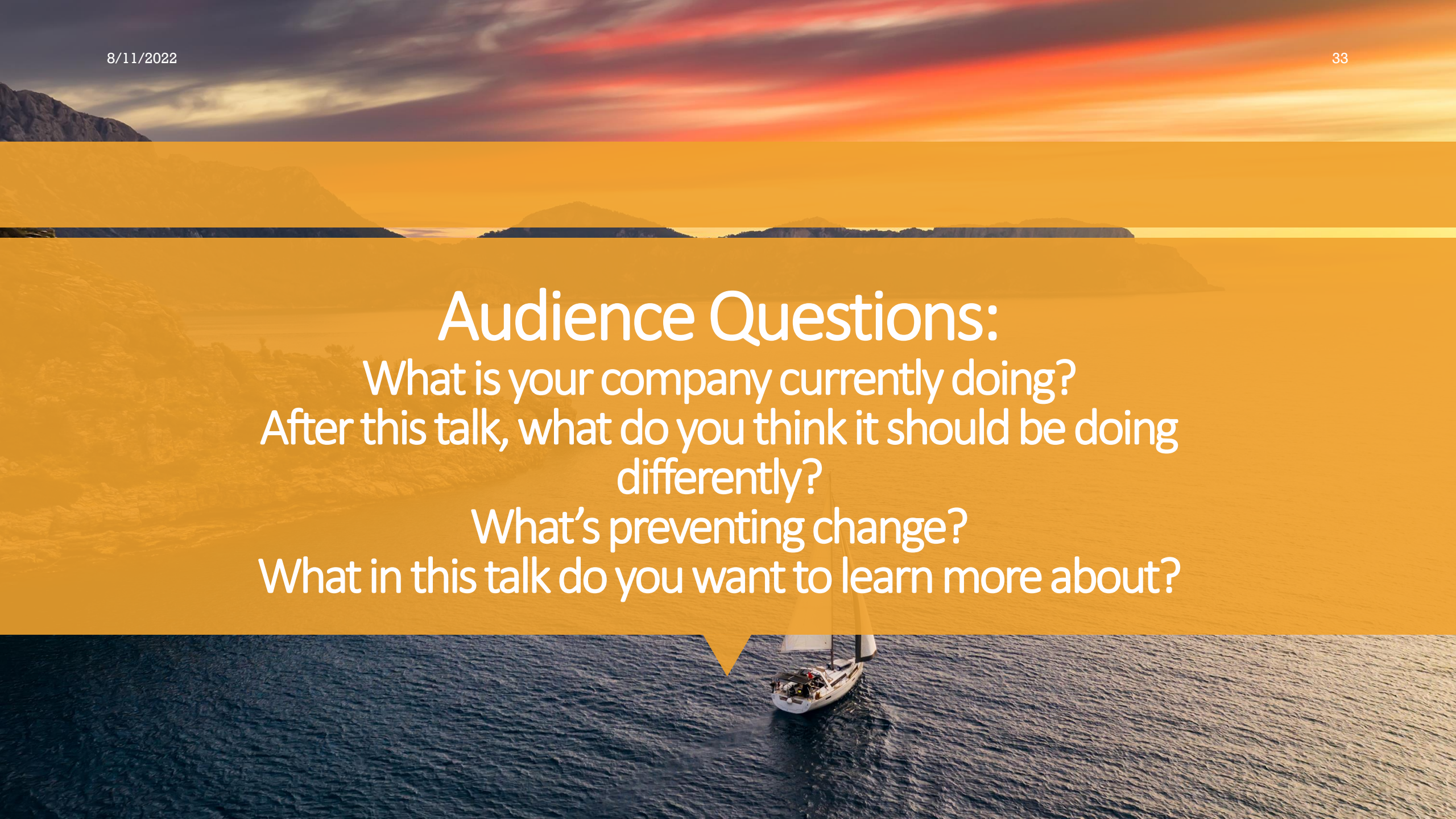
- A high-level rule that provides ongoing governance for your AWS environment. Guardrails can be preventive or detective.

Account Factory

- A configurable account template to standardize the provisioning of new accounts with pre-approved account configurations.

Dashboard

- The dashboard offers continuous oversight of your landing zone to your team of central cloud administrators.



Audience Questions:

What is your company currently doing?
After this talk, what do you think it should be doing differently?
What's preventing change?
What in this talk do you want to learn more about?

So Long and Thanks for All the Fish!

Cassandra Young

Twitter: @muteki_rtw

GitHub: github.com/muteki-apps

LinkedIn: linkedin.com/in/cassandray