# Multi-Account Strategy / Organization Unit

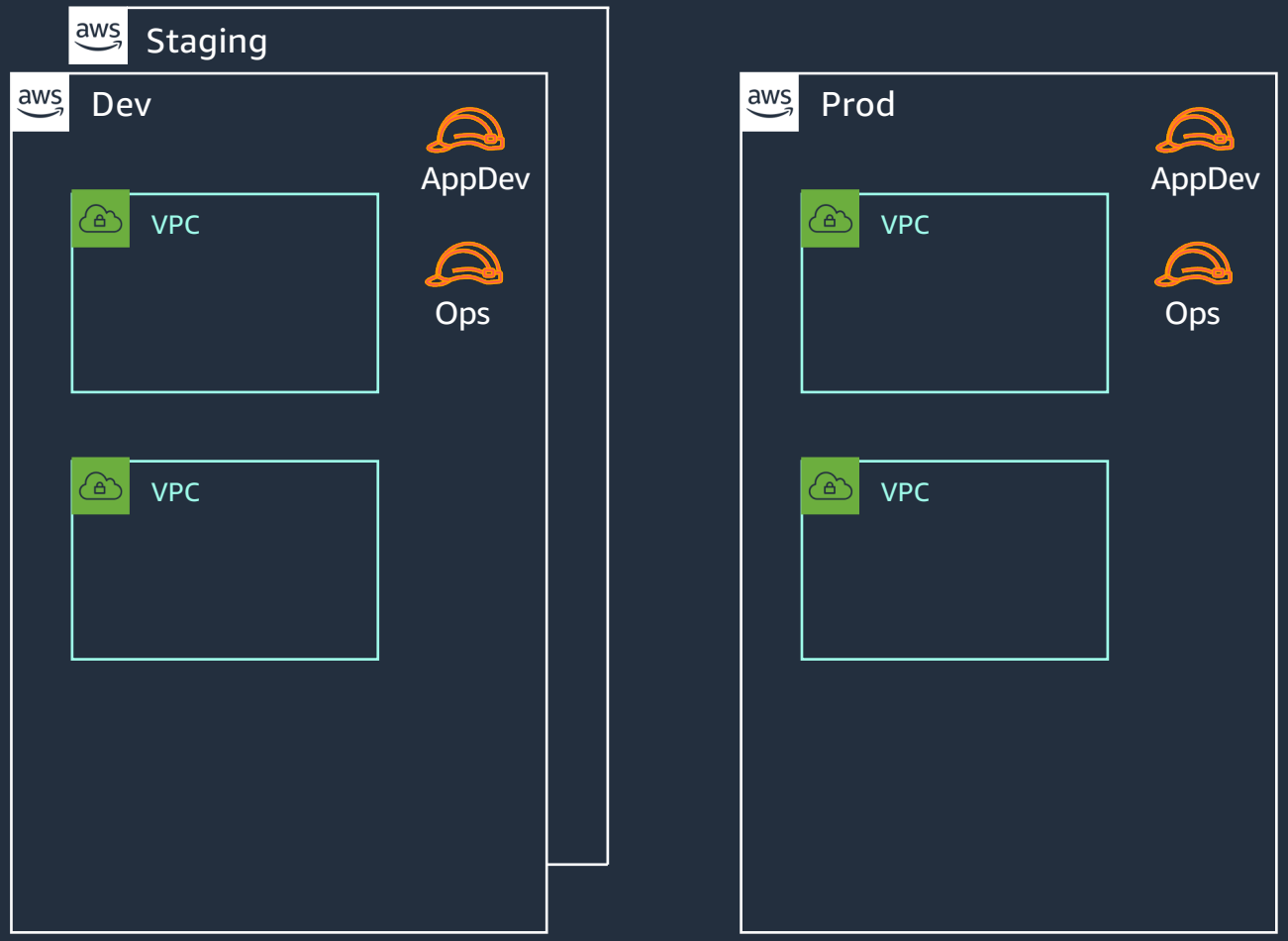
Cloud Architect
Anselmo Shin
AWS Professional Services

# Agenda

- The path to multiple accounts
- Multi-Account 전략
- AWS Organizations
- AWS Organizational Unit(OU) 설계

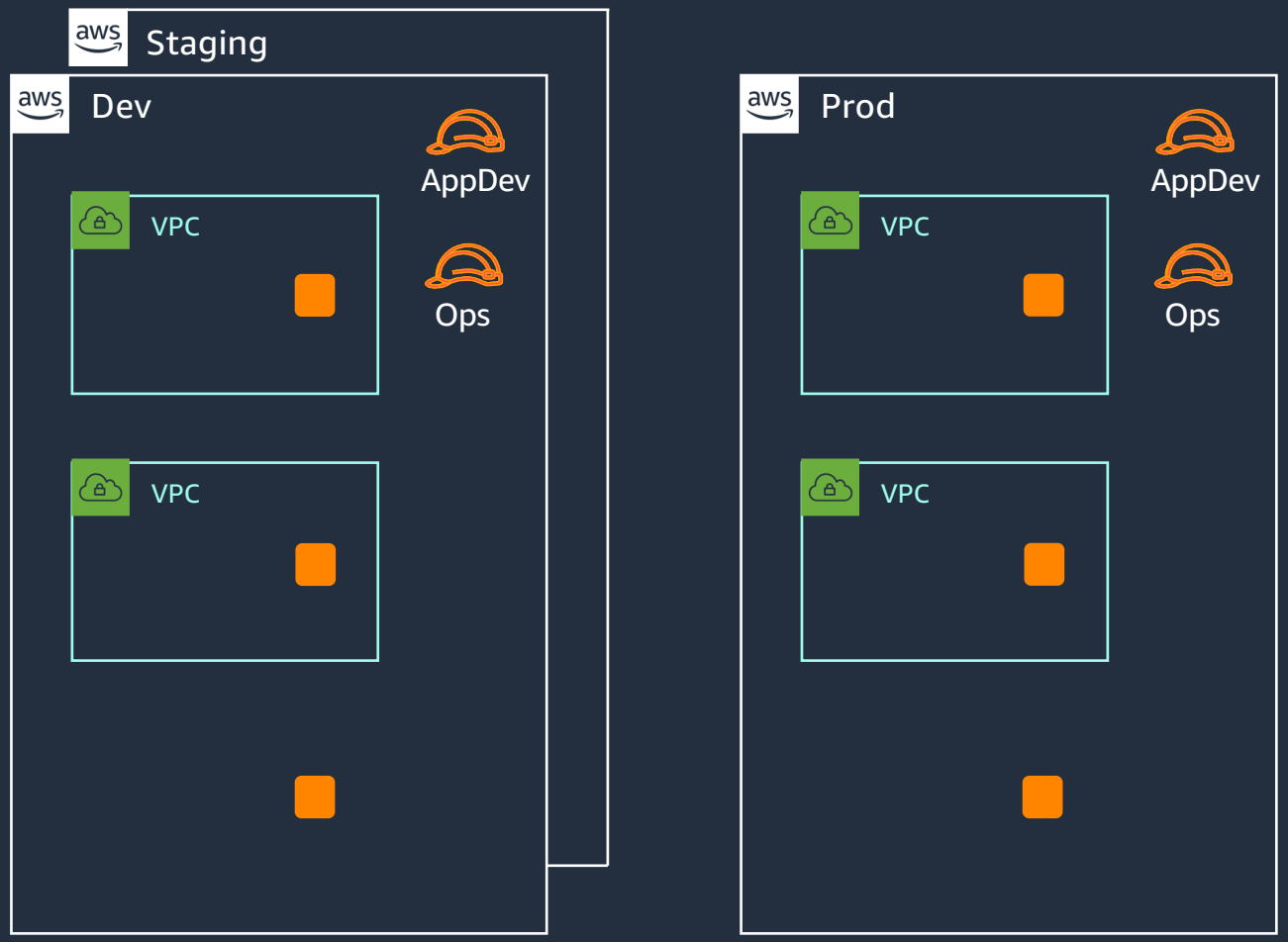
# The path to multiple accounts

# A common starting point



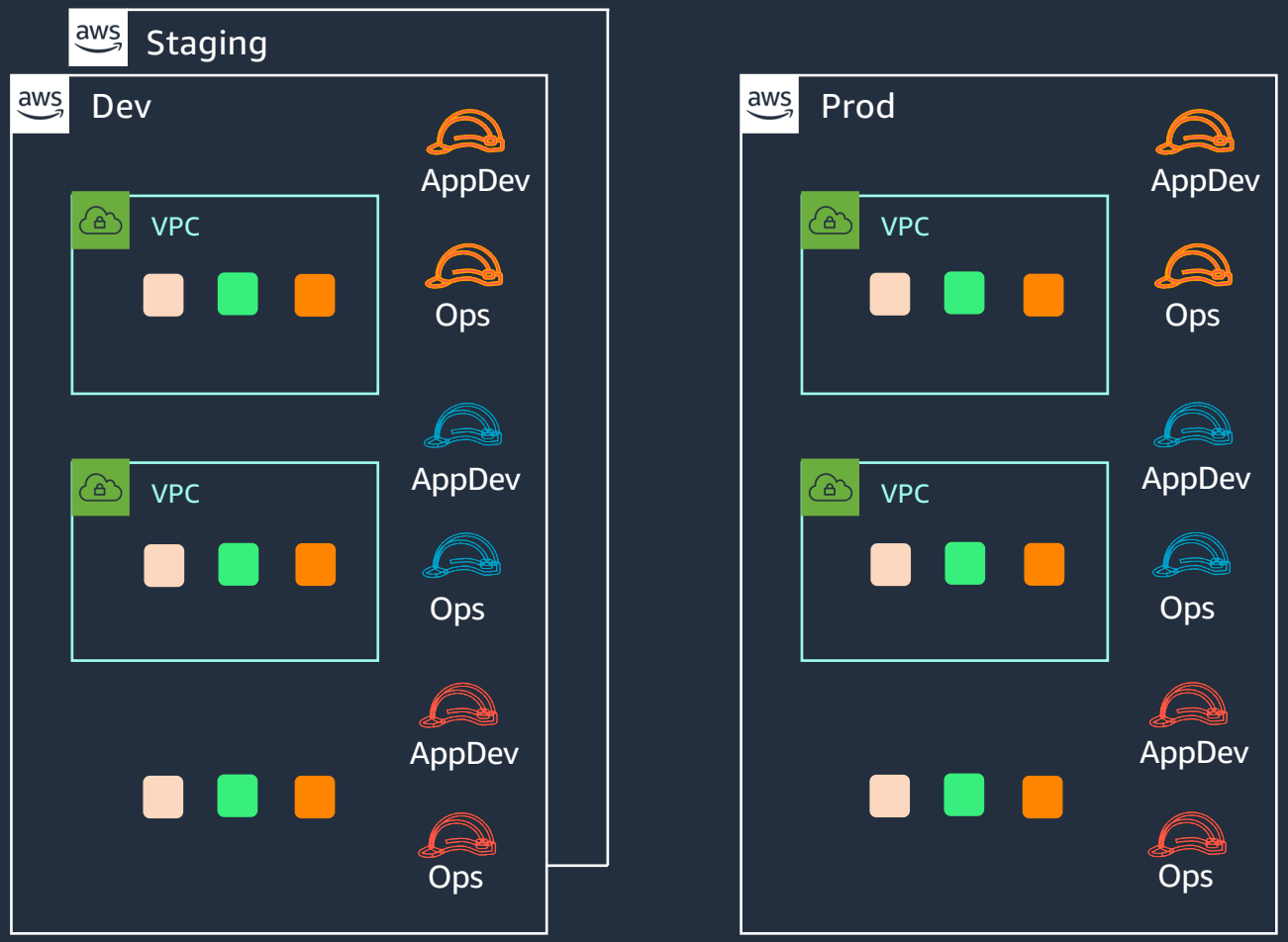

- Limited number of AWS accounts
- VPCs and IAM provide isolation

# A common starting point




- Initially manageable
- Not difficult to manage

# A common starting point

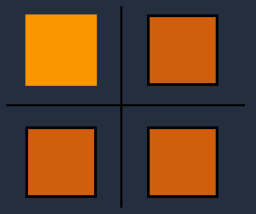



- Complicated and messy over time
- “Grey” boundaries
- Difficult to track resources
- Step on each other’s toes

# The Challenge

Many teams

Billing



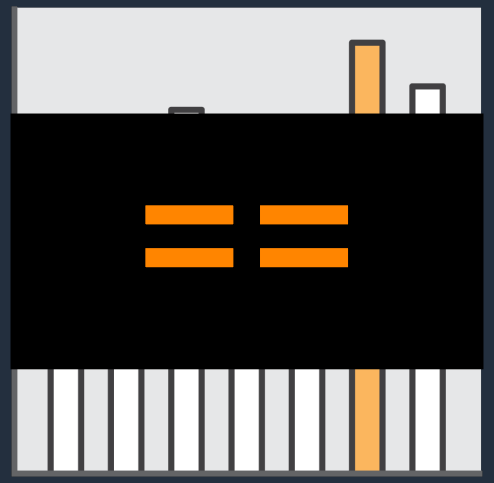
Isolation

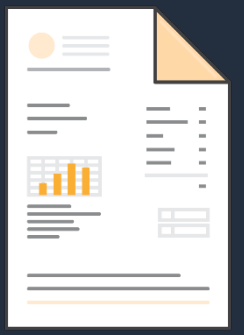
Security / compliance controls

Business process

# Customers Need

# Resource Container



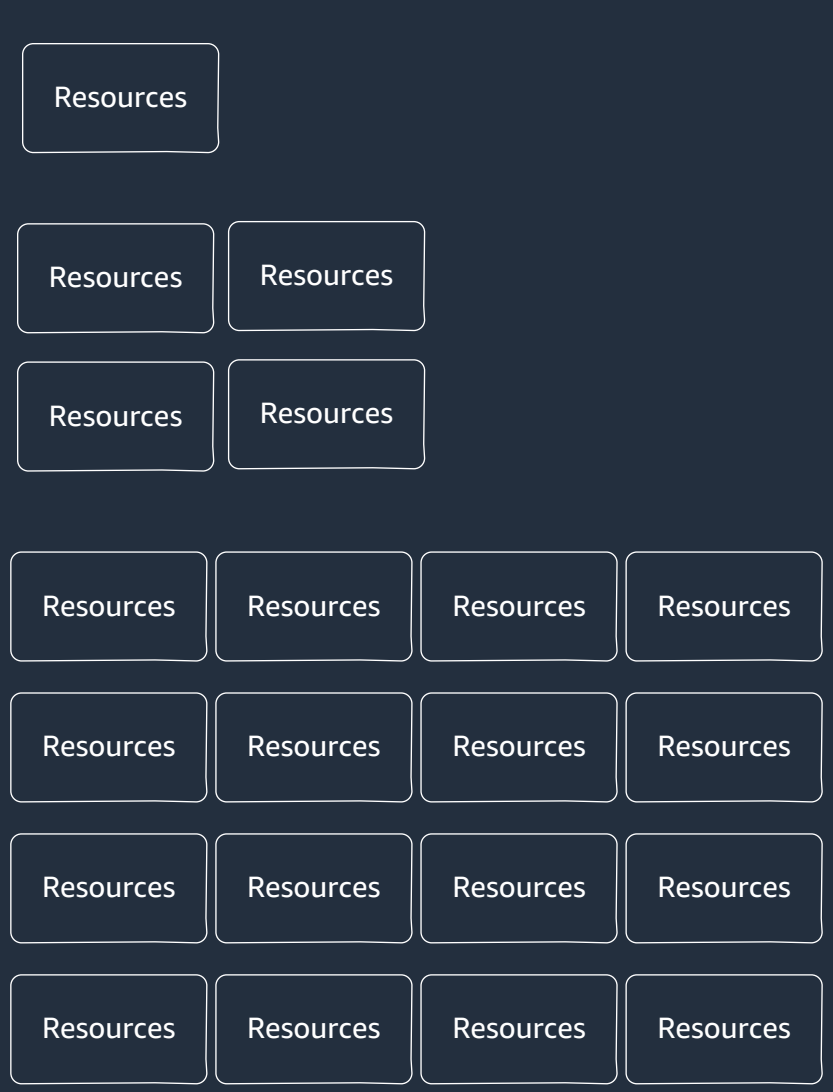


- Security/Resource Boundary
- API Limits/Throttling
- Billing Separation

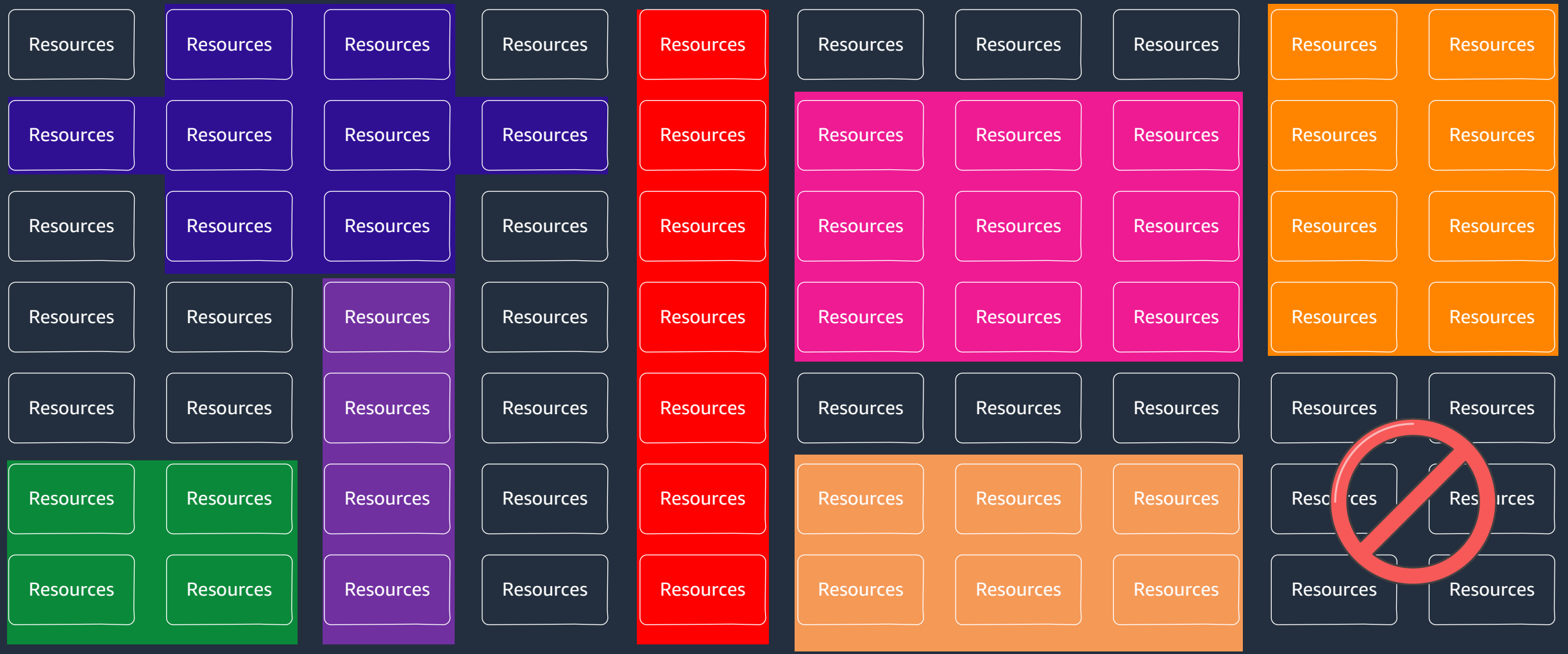
# AWS Account

# Where does a landing zone fit in?

# Resource Containers over time



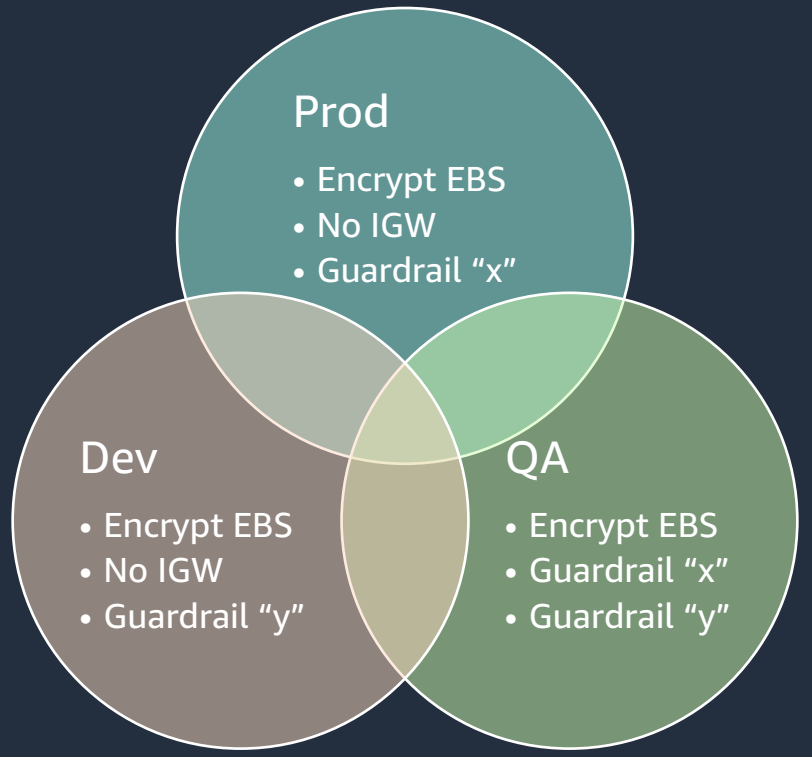

# Resource Containers grouping



# You Need… Orchestration Framework

account management

**Account Metadata:** Owner, function, policies, BU, SDLC, cost center, etc …




policy deployment

policy enforcement

Notification

Remediation

# With capabilities…

Billing Management

Identity and Access Management

Immutable Security Logs

Shared Infrastructure

Resource Isolation

Support Dev Lifecycle

Central Network Connectivity

Security Tooling

# You need 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

# landing zone, AWS Landing Zone, AWS Control Tower

**landing zone:**

- 사전 구성된 안전한 AWS 환경 제공
- 확장 가능하고 유연함
- 민첩성과 혁신 제공

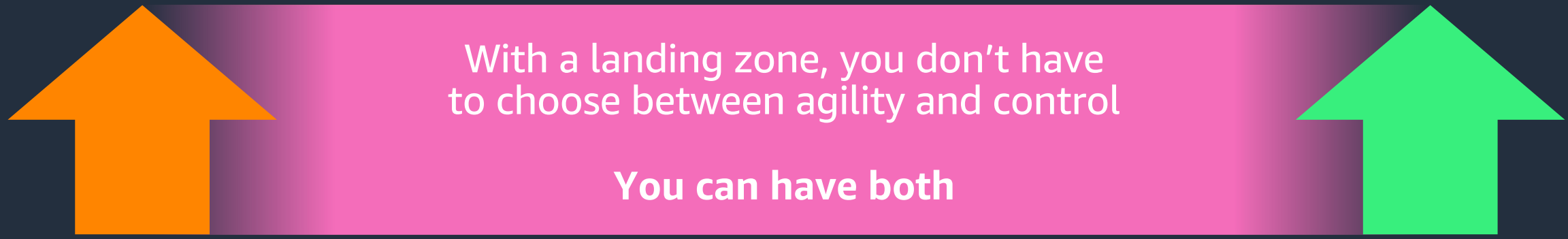
**AWS Landing Zone Solution:**

- 다중 계정 전략 지침을 기반으로 landing zone 구현
- 고객이 코드를 사용하여 직접 관리 및 유지보수 가능
- ALZ 솔루션은 2020년 이후로 업데이트 중지

**AWS Control Tower:**

- AWS Landing Zone의 AWS Managed Service 버전

# Business agility *and* governance control




## Agility

Self-service access

Experiment fast

Respond quickly
to change

## Governance

Security

Compliance

Operations

Spend Management

# Multi-Account 전략

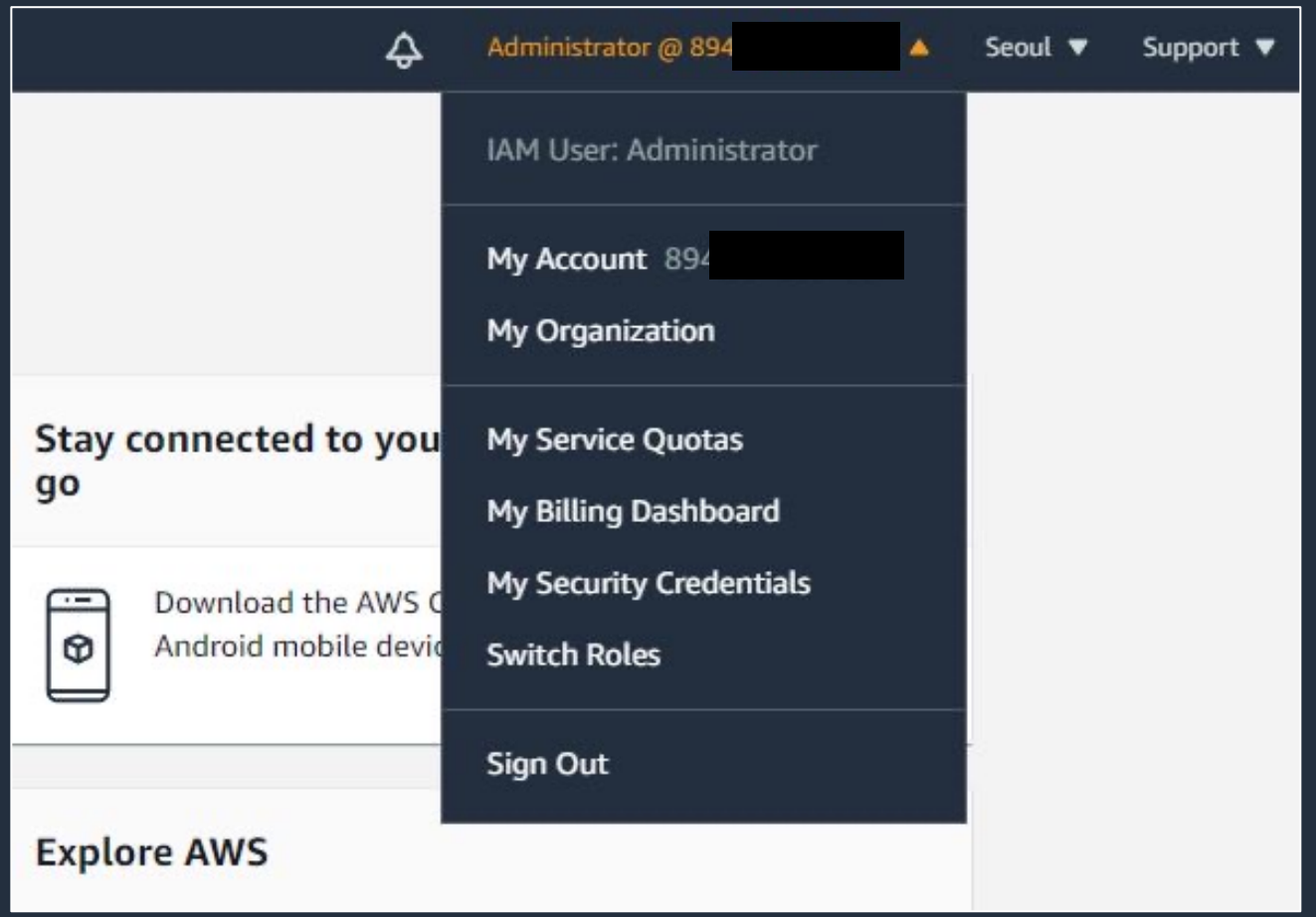
# AWS 계정(Account)

AWS를 사용하기 위해서는 계정(account)을 생성하여야 하며, 계정은 빌링과 리소스 관리의 기준이 됩니다

**AWS 계정**

- 12자리 숫자로 표현됨
- 가입 시 이메일 주소, Root 암호, 신용카드 정보, 연락처가 필요함
- 빌링의 기준이 됨
- AWS에 로그인을 위해서는:
  A. 가입 시 입력한 이메일 주소와 Root 암호를 사용 → “Account Root”
  B. 또는 “IAM User”를 생성한 후 이를 통해 로그인

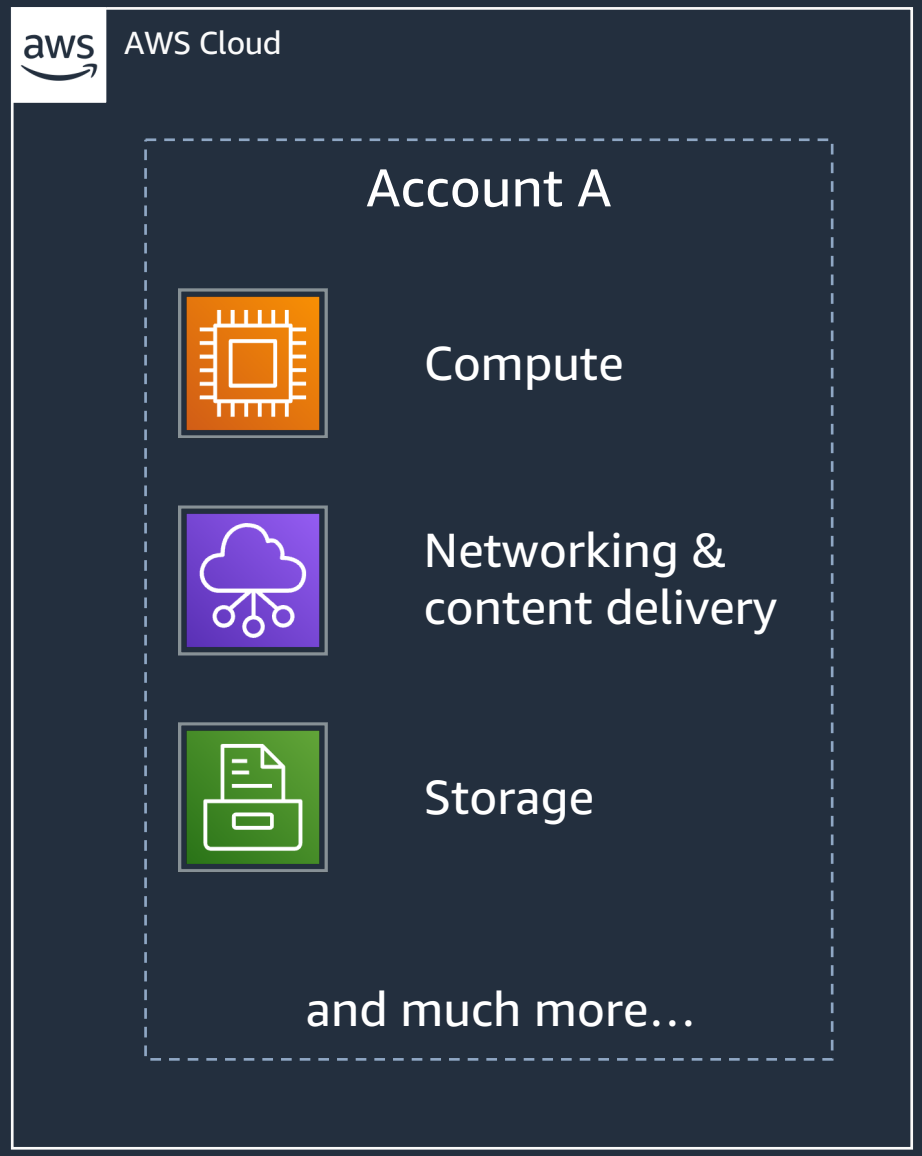
**Enterprise 환경에서는 AWS 계정을 여러 개 사용하는 Multi-Account 구조가 권장됨**



# AWS 계정(Account)

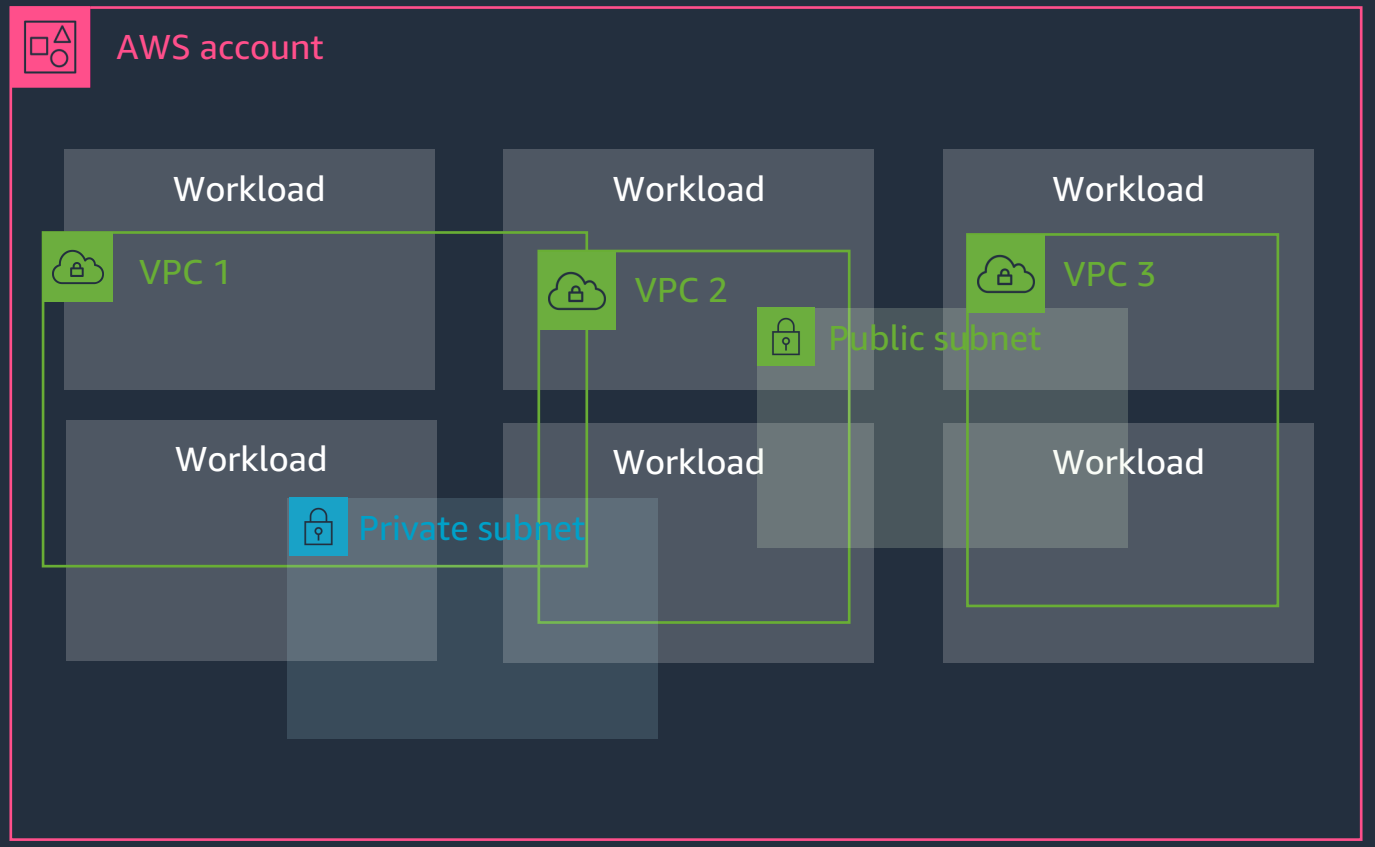
Each AWS Account:

- 리소스 컨테이너(resource container)
- 명시적인 보안 경계
- 빌링 및 비용 추적을 위한 단위(container)
- AWS 리소스 제한 기준(Limits, Thresholds)
    - e.g. Service Quotas and API thresholds

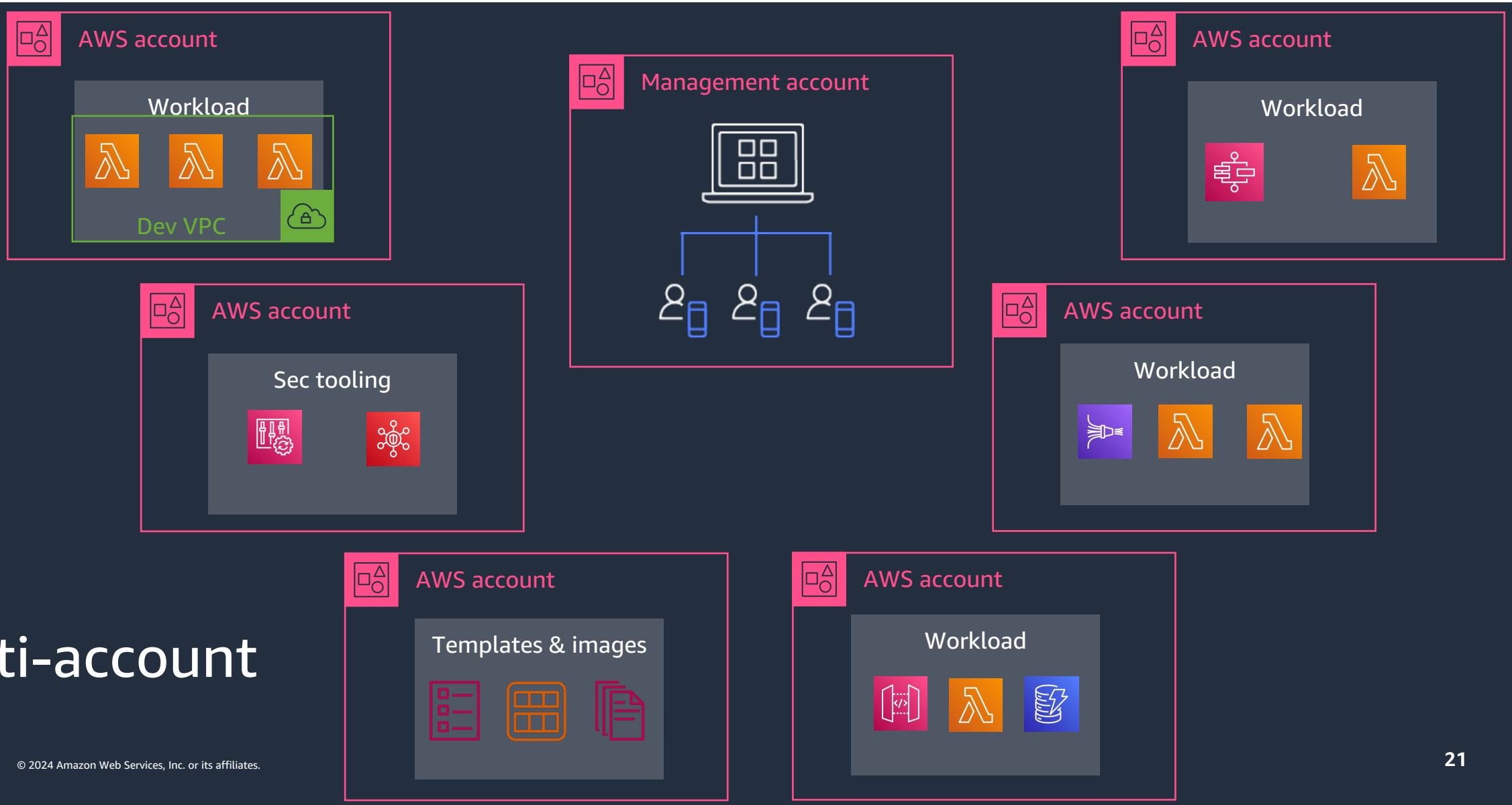


# AWS 계정(Account)

## Single account



# AWS 계정(Account)




Multi-account

# Multi-account 모델로 확장

Many teams

팀별 빠른 리소스 제공 및 독립적 사용 보장으로 혁신 가속화

Billing

AWS 계정 내에서 사용된 리소스를 해당 계정을 담당하는 사업부에 할당할 수 있는 청구 간소화

Business process

다양한 운영, 규제, 비용제한이 있는 비즈니스 프로세스에 맞게 계정 구성

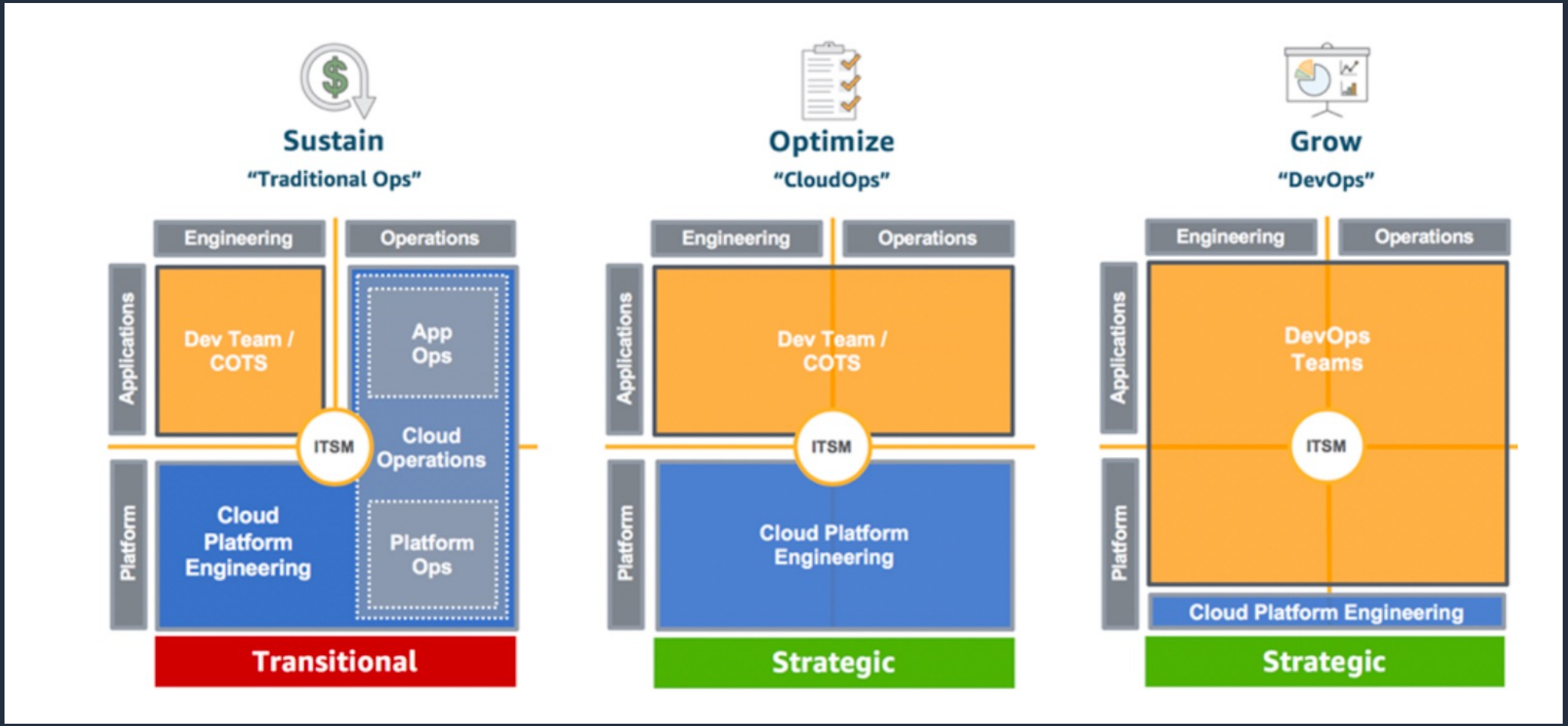
Isolation & security

계정 간 기본적인 격리를 통해 보안 경계를 엄격하게 설정하고 유사한 리스크 유형을 가진 워크로드를 통합

# AWS Multi-Account의 장점

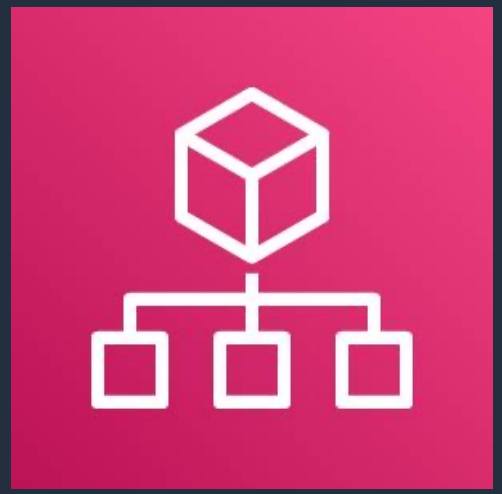
| | |
|---|---|
| • 공통된 비즈니스 목적을 가진 워크로드들을 그룹화 | 계정과 워크로드와의 소유 및 의사결정 과정을 일치시켜 다른 워크로드와의 운영, 보안, 충돌 이슈 방지<br>- 가드레일: 보안, 운영 규정준수를  규칙으로 적용 |
| • 운영환경 별 고유한 보안 정책 적용 | 워크로드간 별도의 보안 제어정책 및 관리 매커니즘 적용 |
| • 민감정보에 대한 접근 제한 | 데이터 저장소에 접근 사는 사람 및 시스템을 쉽게 제한(ex: S3) |
| • 비즈니스 혁신과 민첩성 향상 | 상용환경보다 넓은 접근성을 제공하는 보안 가드레일 및 비용 예산이 적용된 샌드박스 계정 , 개발 계정을 통해  비즈니스 속도 향상 |
| • 장애 영향 범위 제한 | 계정간 리소스 격리를  보장하기 위한 보안, 접근, 빌링 경계(Boundary)를 제공하여 잘못된 작업의 영향 범위를 최소화 |
| • 다양한 IT 운영 모델 지원 | 다양한 ITSM 모델(Transitional, CloudOps, DevOps)별 운영 계정을 들 분리하여  지원 |
| • 비용관리 | 서로 다른 비즈니스, 워크로드 그룹을 계정으로 분리하여 클라우드 지출을 쉽게 모니터링, 제어, 예측, 예산 관리 |
| • AWS 서비스 할당량 및 API 요청 제한 | 기본적으로 계정 별 사용할 수 있는 리소스 및 API요청 량이 할당(Service Quoatas)되어 있으며 잠재적인 영향을 분산할 수 있음 |

# *Example operating models*



# AWS Organizations

# AWS Organizations



클라우드 환경을 중앙에서 관리할 수 있는 기능을 제공

계정 생성, 자원 할당 통해 빠르게  확장 가능

운영환경에 맞는 보안정책(Service Control Policy)적용

통합 보안 및 감사 관리

등록된 계정의 통합 비용관리

# AWS Organizations 이용한 중앙 거버넌스 관리

다중 계정 환경에서 리소스를 중앙에서 프로비저닝

- AWS CloudFormation
- AWS Systems Manager
- AWS Service Catalog

OU내 리소스를 공유하고 계정, 리전 및 서비스에 대한 액세스를 제어

- AWS Personal Health Dashboard
- AWS Resource Access Manager
- AWS Backup & Backup Policies
- Tag Policies

비용 최적화 및 비용 절감 관리

- AWS Trusted Advisor
- AWS Compute Optimizer
- AWS Cost Explorer
- AWS License Manager
- S3 Storage Lens

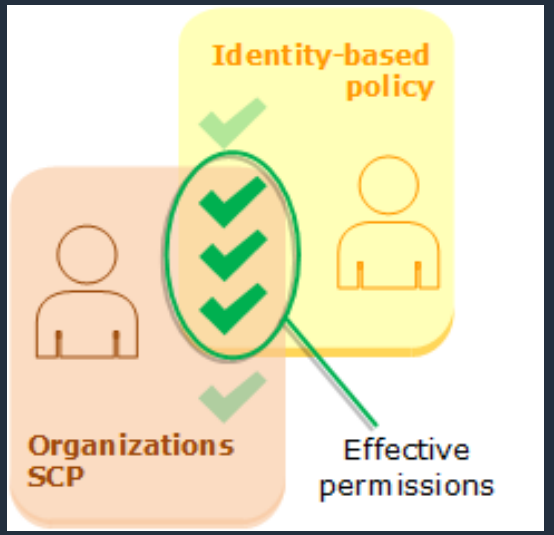
AWS 보안 서비스와 통합 관리

- AWS Audit Manager
- Amazon GuardDuty
- Amazon Cloud Directory
- AWS Firewall Manager
- Amazon Macie
- AWS Security Hub
- AWS IAM Access Analyzer
- AI/ML Policies

https://docs.aws.amazon.com/organizations/

# Service Control Policies (SCPs)

- Enables you to control which AWS service APIs are accessible
- Permission
  - Intersection between the SCP and IAM permissions
  - IAM policy simulator is SCP aware
- Applied at
  - The Account or Organizational Unit

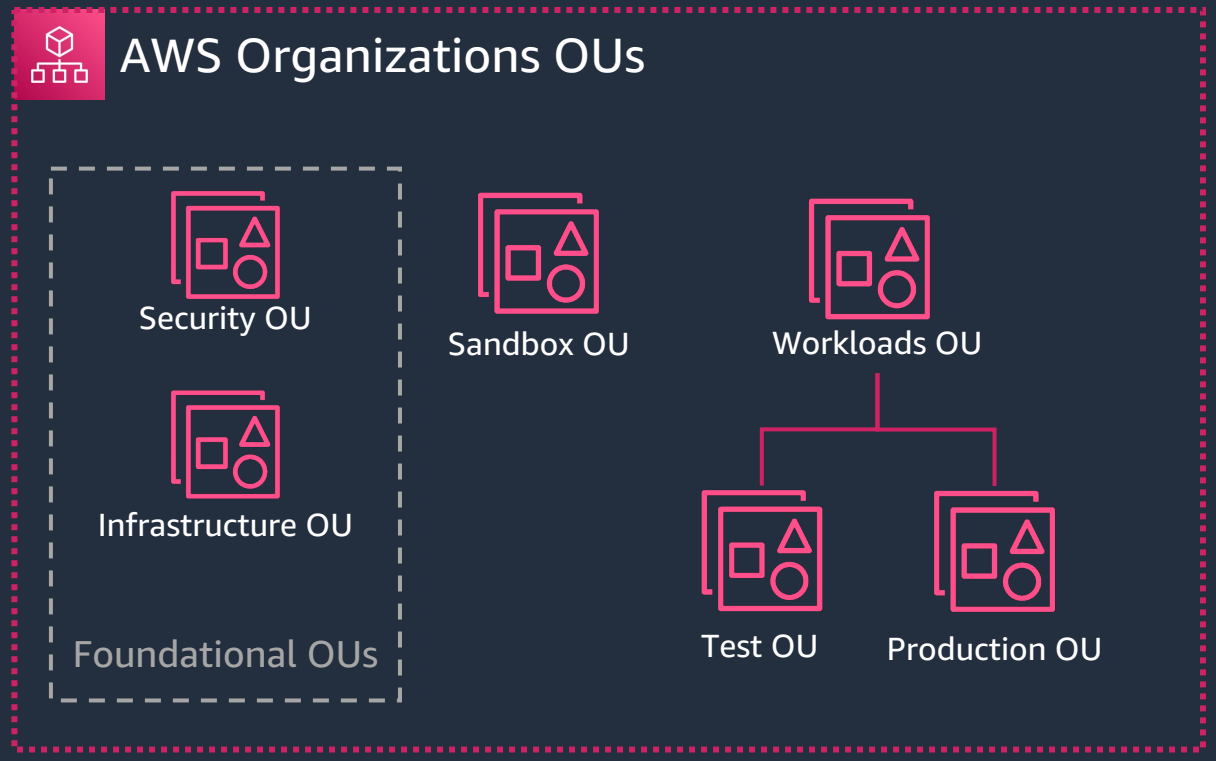


# Organizational Units

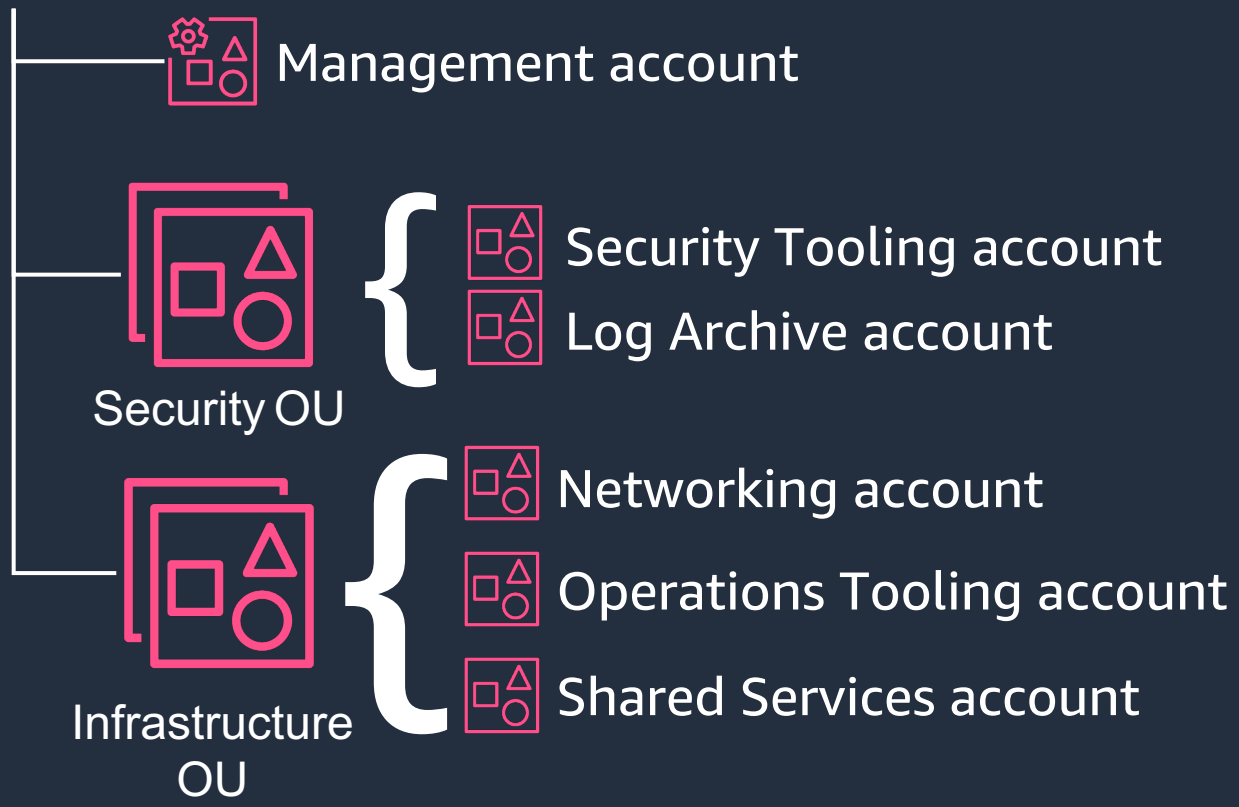
- Grouping of AWS Accounts
- Service Control Polices (SCP) to the groups
- Use permission grouping (NOT corporate structure)

How likely is the group to need a set of similar policies?

# Start small – Production starter Organization

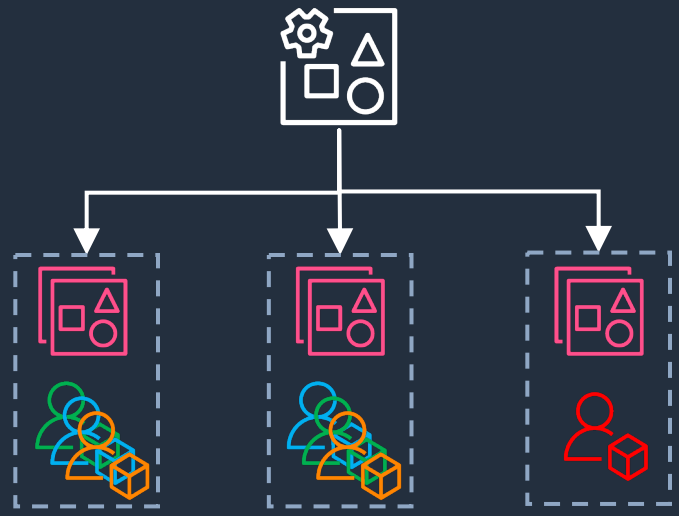


# Foundational accounts



# Management Account

- AWS Organizations
- Service control policies
- Tag policies
- AWS CloudTrail

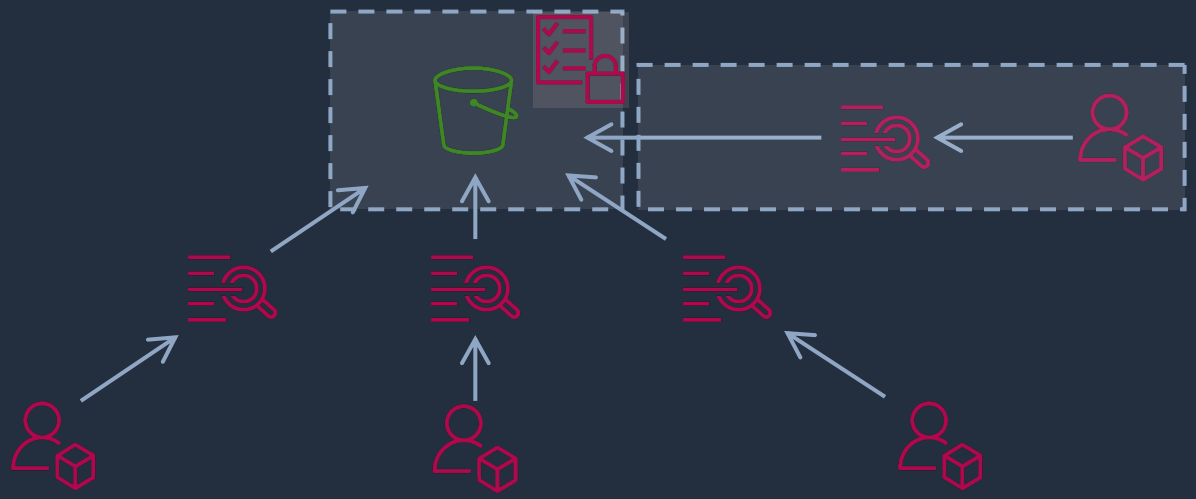


# Security Tooling Account

- Security Hub
- GuardDuty
- Macie
- Detective
- AWS Config
- IAM Access Analyzer

Delegated administrator account

# Log Archive Account



Access logs

DNS logs

Flow logs

CloudTrail events

# Networking Account

AWS Transit Gateway

Cross-account VPCs

IP Address Manager (IPAM)

Delegated administrator account

# Operations Tooling Account

- AWS CloudFormation
- AWS Systems Manager
- Amazon CloudWatch
- AWS Backup

Delegated administrator account

# Shared Services Account

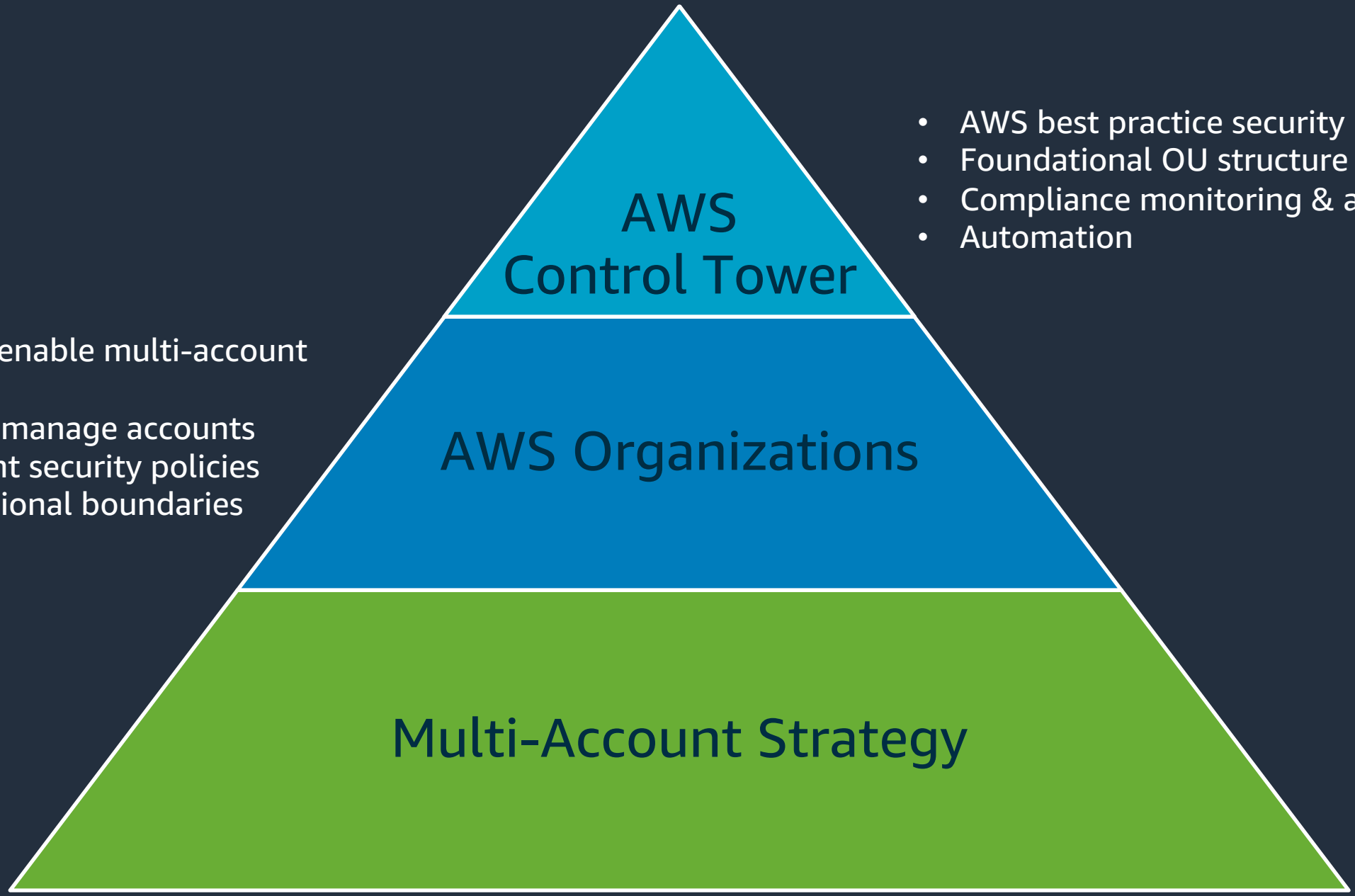
AWS SSO

AWS License Manager

Delegated administrator account

# Control Tower and Service Catalog
# For secure acceleration

- Define & enable multi-account strategy
- Centrally manage accounts
- Implement security policies and functional boundaries




- AWS best practice security policies
- Foundational OU structure
- Compliance monitoring & alerting
- Automation

# 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 of account activity

Account factory for account provisioning


Built-in monitoring and notifications

Dashboard for visibility and actions

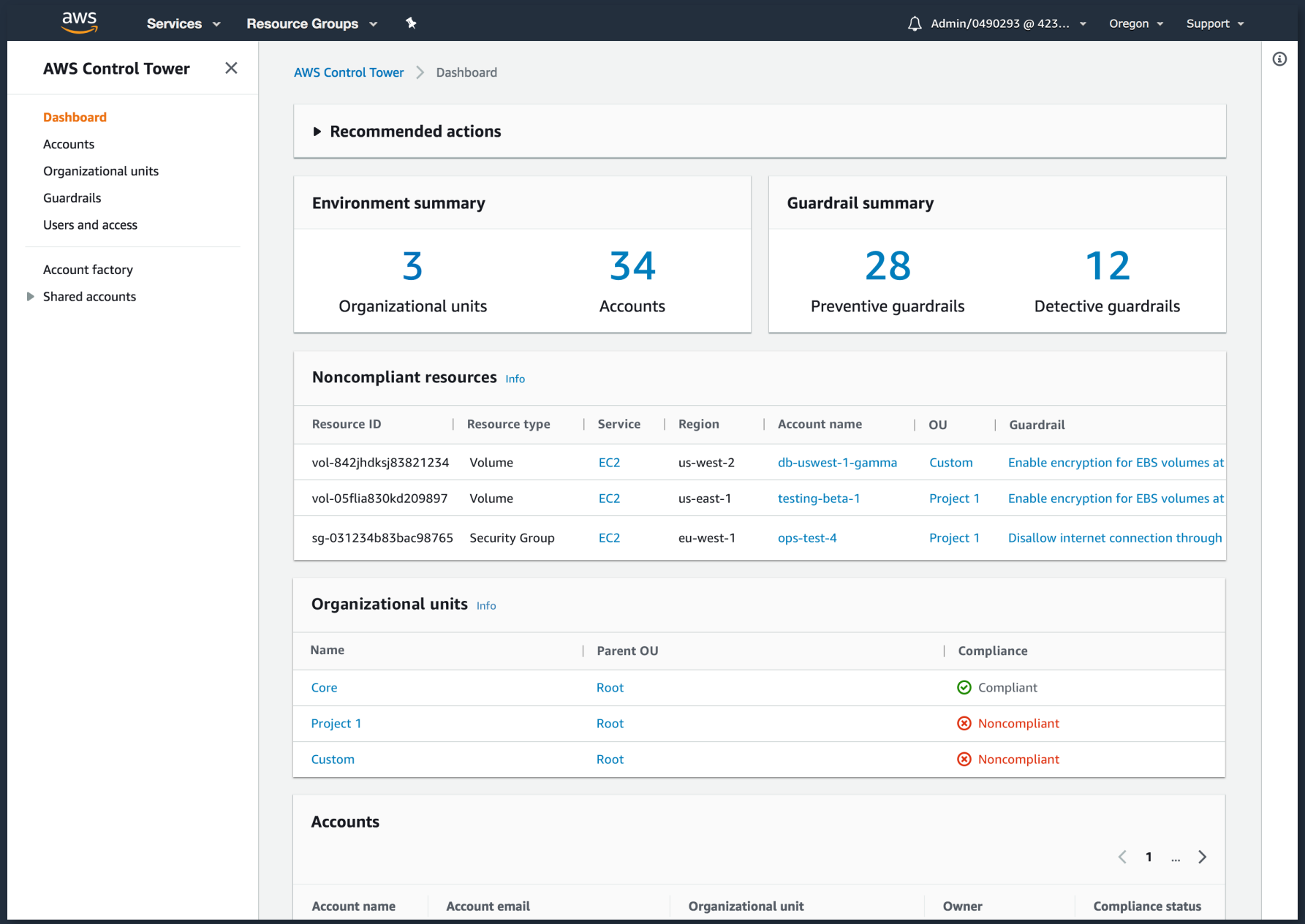
Automatic updates

*The easiest self-service solution to create a governed **AWS multi-account environment***

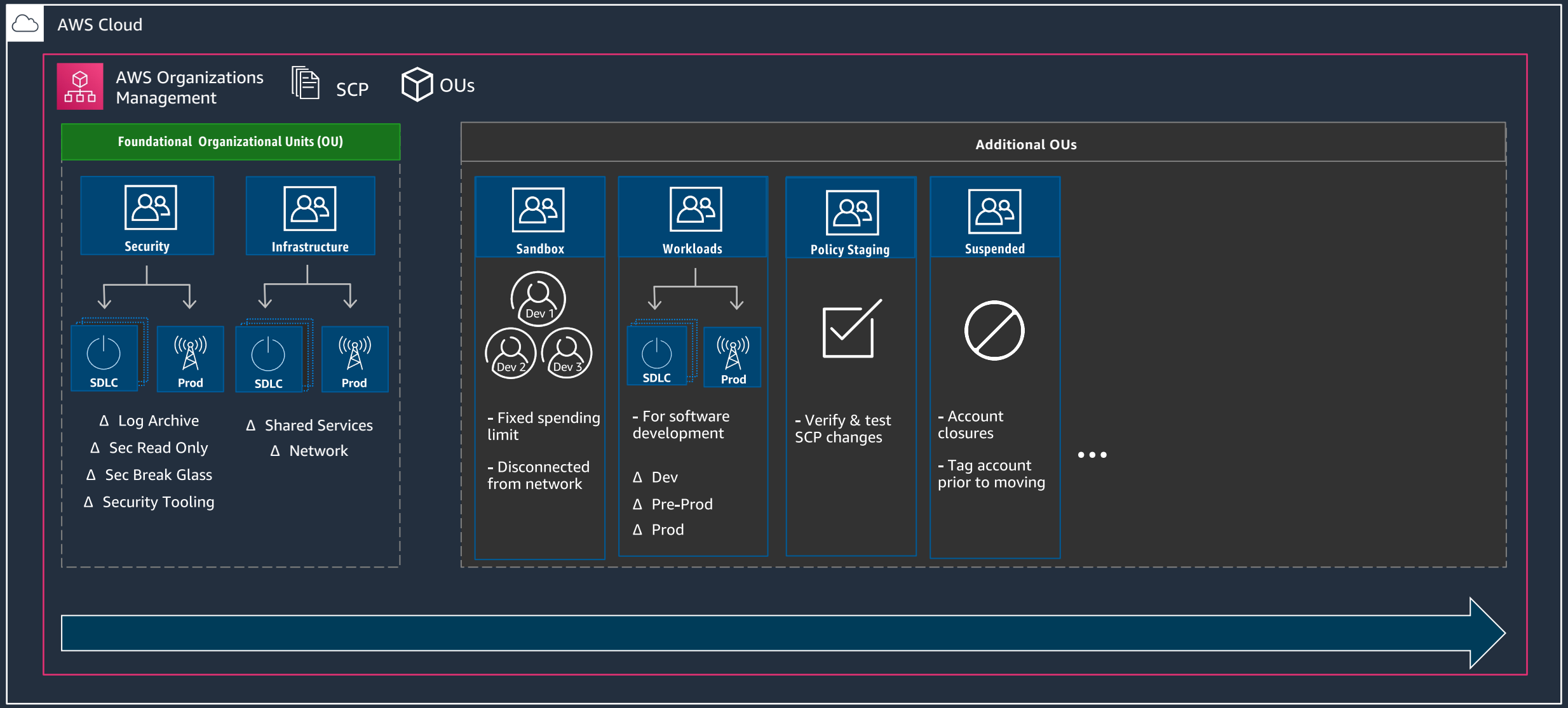


aws

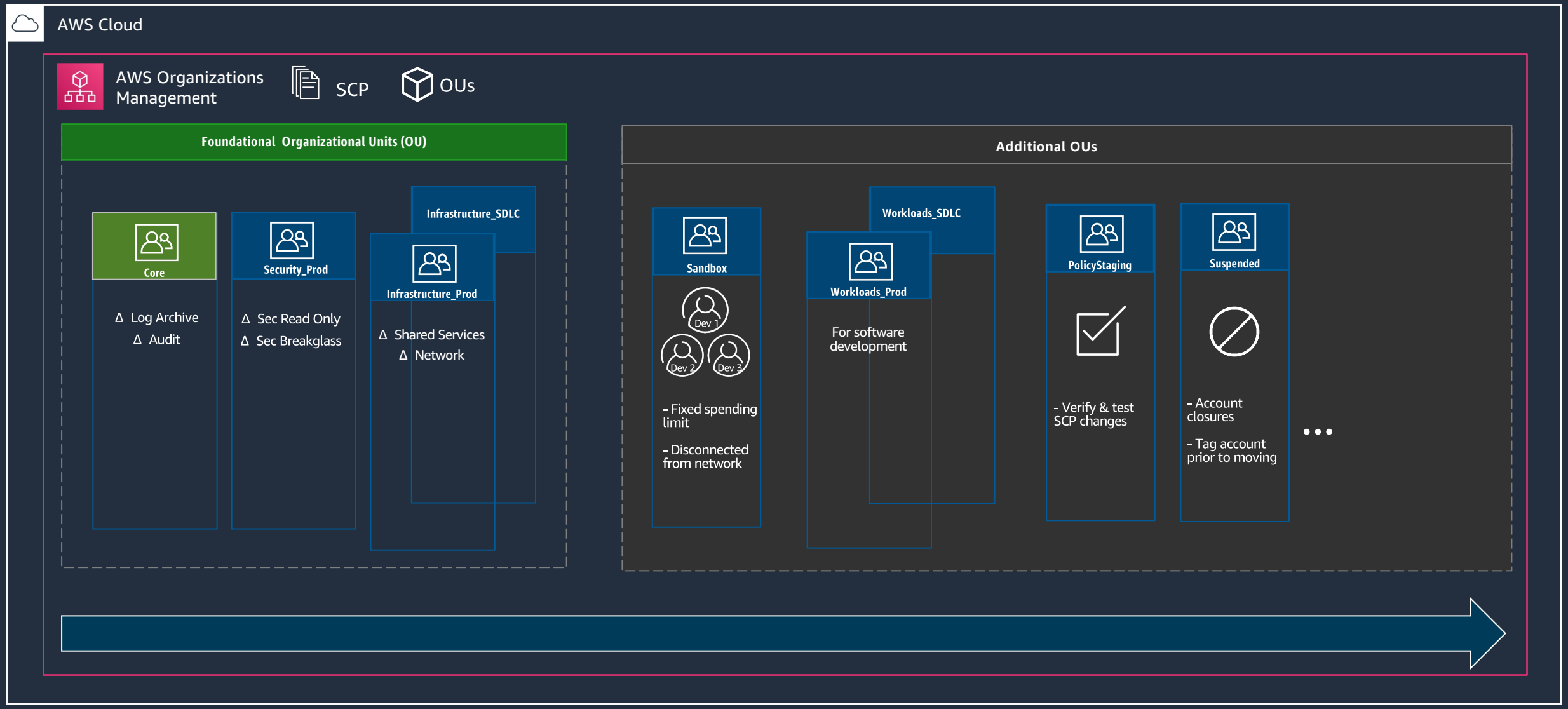
# Dashboard for oversight



# AWS Organizations nested OU structure

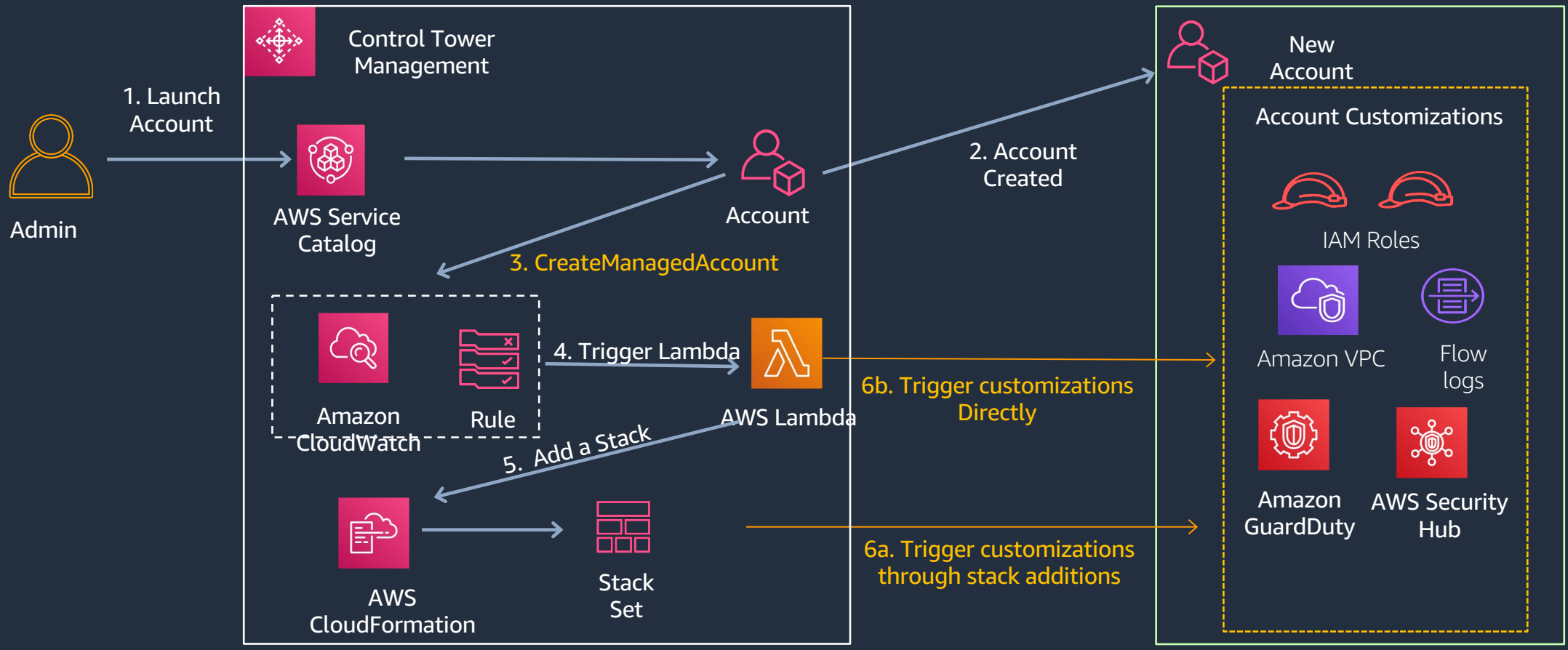


# AWS Control Tower flat OU structure

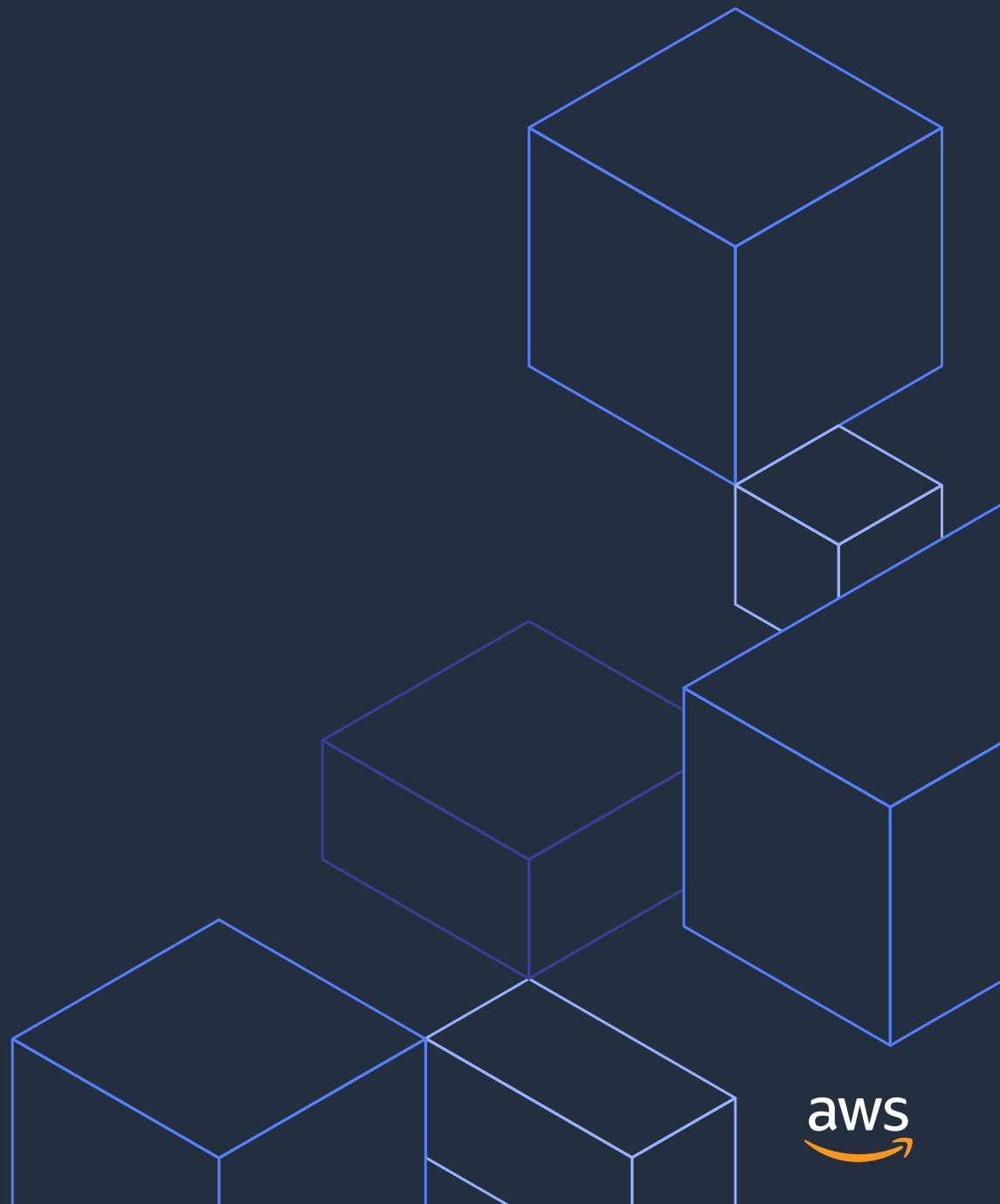


# Extend via AWS Control Tower Lifecycle Events

- **CreateManagedAccount:** The log records whether AWS Control Tower successfully completed every action to provision a new account using account factory.



# AWS Service Catalog



# AWS Service Catalog

Standardize
Secure
Control

Agility
Self-Service
Time to Market

*…allows organizations to create and manage catalogs of IT services on AWS*

**Organizations**

**Developers**

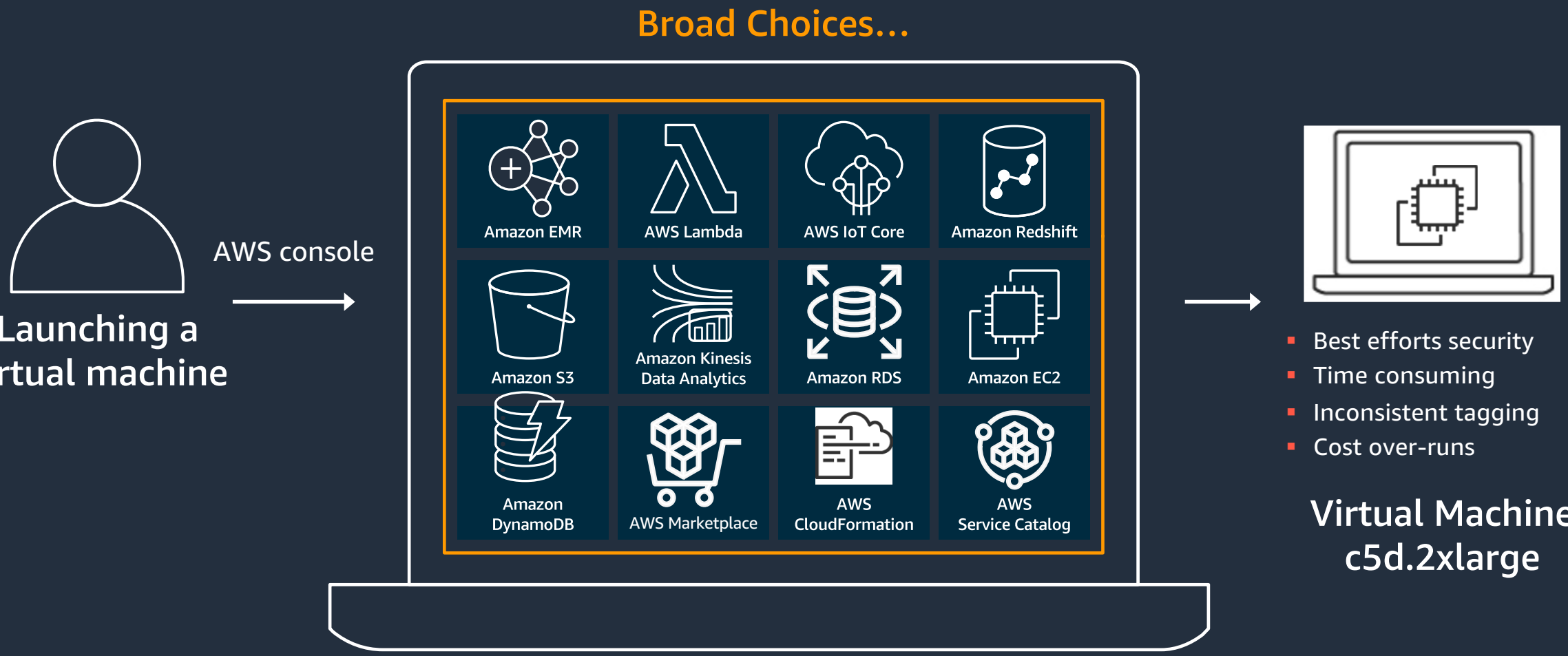


aws

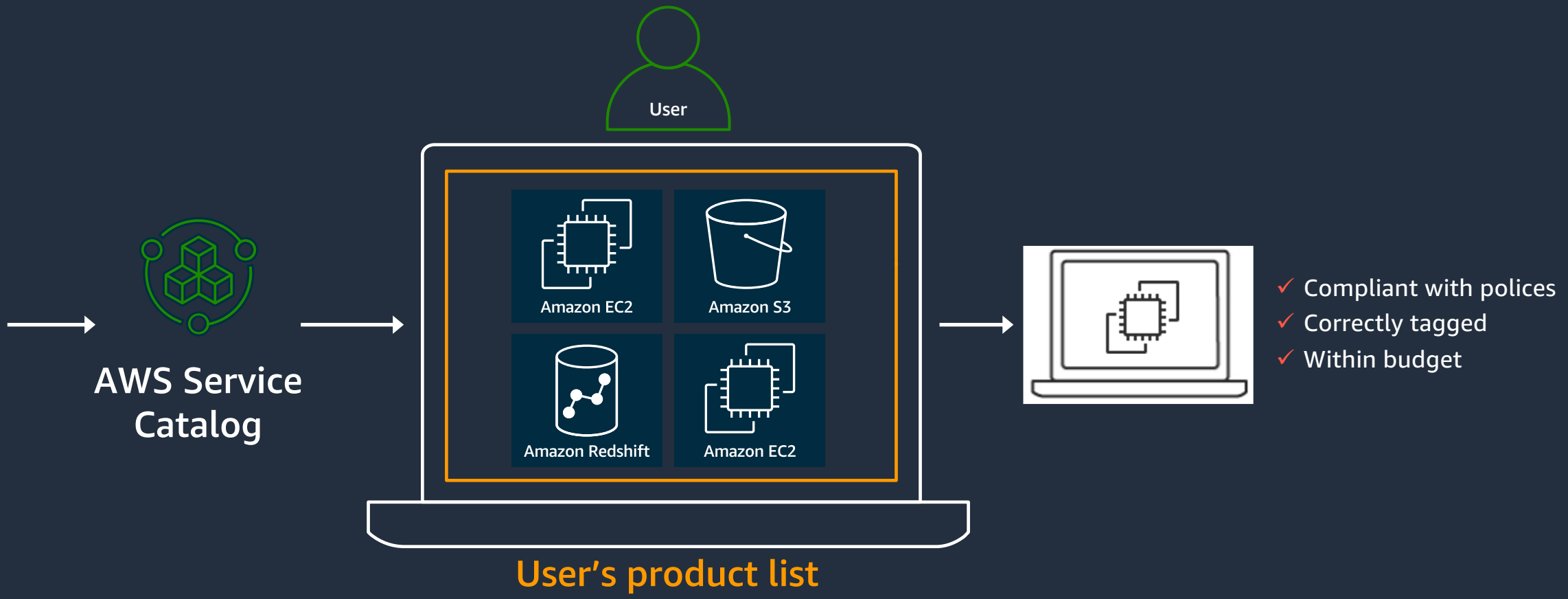
# Self-service with preconfigured compliance


Standardizes best practices

# Self-service *without* AWS Service Catalog



# Self-service *with* AWS Service Catalog



# AWS Service Catalog Benefits

*Secure Acceleration*

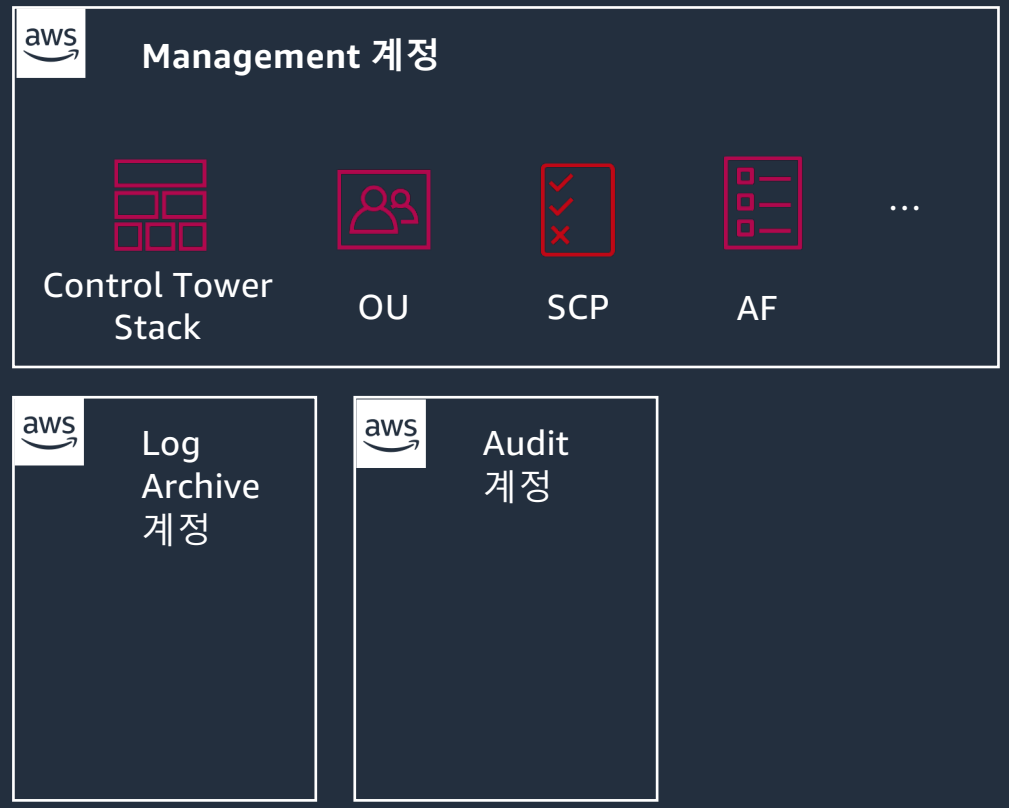
- Control AWS Provisioning (Cost, Security, Governance)
- Enforce Governance and Compliance proactively
- Self Service Portal – Developer Acceleration
- Standardized Deployments (no re-inventing the wheel)
- Centrally Manage IT Service Life Cycle
- Versioned Products
- Integrate with ITSM tools



aws

# AWS Control Tower(Core) Account – Management

**Management 계정**은 Control Tower 자체를 관리하기 위한 계정으로서 필수입니다




**목적**

- Control Tower 구성 및 운영 관리
- Organizational Unit (OU) 관리
- Service Control Policy (SCP) 관리
- 통합 빌링 정보 확인
- Account Factory (AF)를 실행하여 AWS 계정 생성

**특징**

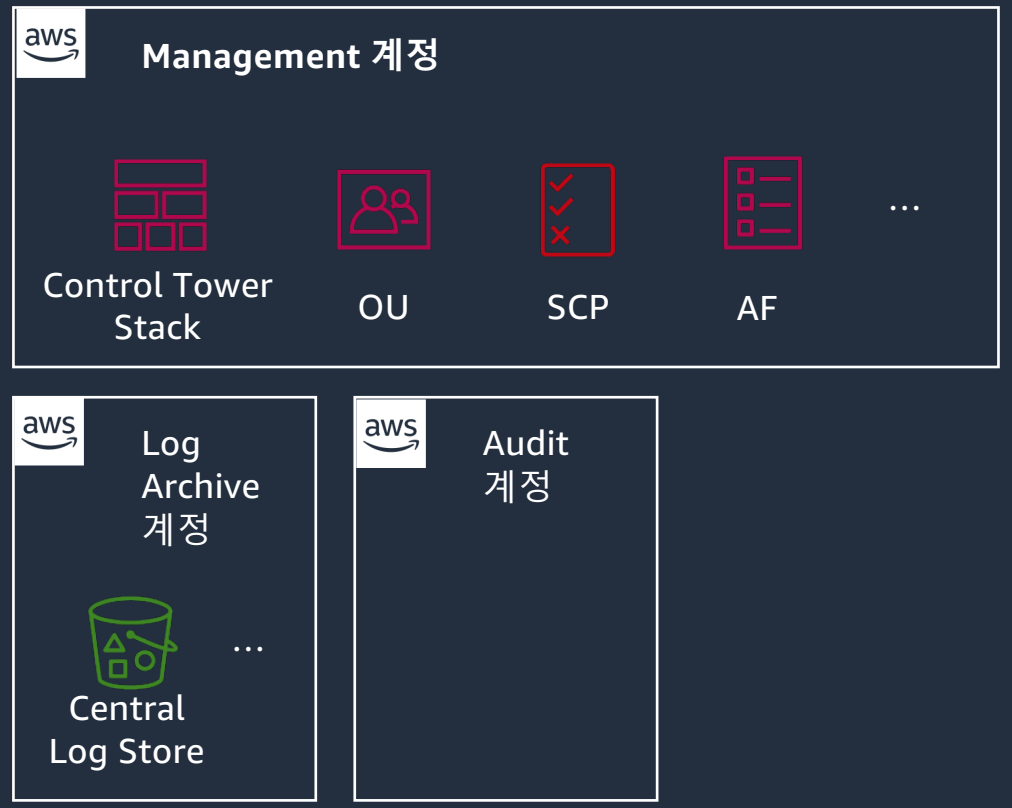
- 필수 인원만 제한적으로 접근함
- On-prem DC와 연결하지 않음
- **Control Tower 외에 AWS 리소스는 최소한으로 함**

**필수 계정**

* "**Master 계정**", "**Payer 계정**", 또는 단순히 "**Management 계정**"도 같은 표현

# AWS Control Tower(Core) Account – Log Archive

**Log Archive 계정**은 Control Tower 자체를 관리하기 위한 계정으로서 필수입니다




**목적**

- S3 버킷 (제한된 액세스 및 삭제 시 MFA 적용)
- CloudTrail 로그 적재
- 그 외 보안 관련 로그 적재
- 이를 통한 보안 로그의 중앙 적재 (centralized logging)
- 보안 관련 Single Source of Truth (immutable security log)

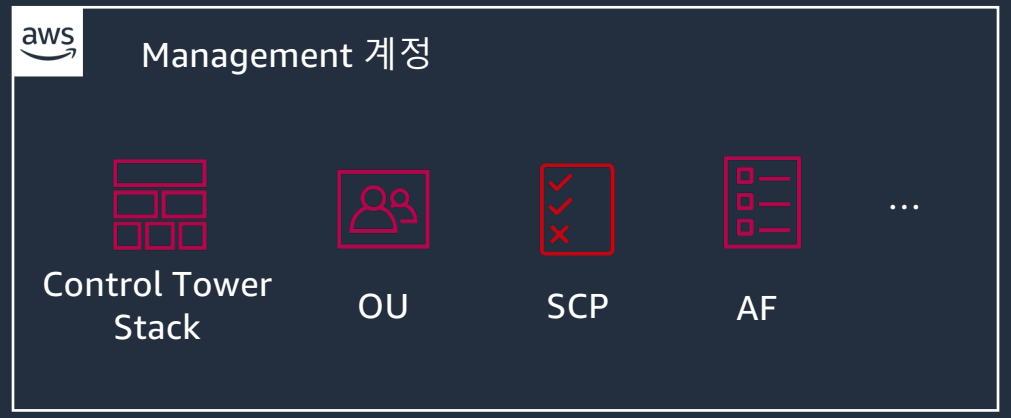
**특징**

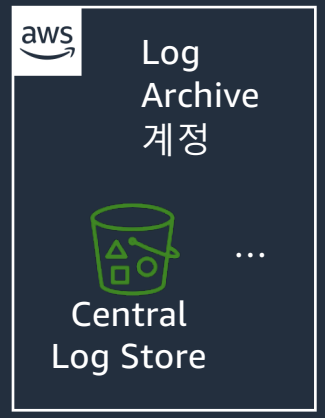
- 필수 인원만 제한적으로 접근하며 로그인 자체를 통제
- 로그 관련 툴을 설치하고 실행하기 위한 계정은 아님

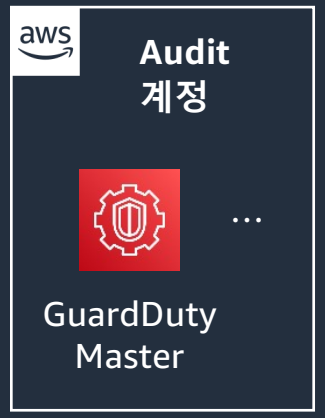
**필수 계정**

# AWS Control Tower(Core) Account – Audit

**Audit 계정**은 보안 관련 AWS 서비스 및 툴이 위치하는 필수 계정입니다




**목적**

- **GuardDuty master** – GuardDuty 서비스 상에서 다른 계정의 결과를 조회하고 관리하는 master의 역할을 수행
- **Security notifications** – SNS 서비스를 활용한 보안 관련 알림 기능이 위치
- 향후 출시되는 **AWS 보안 관련 서비스**가 위치
- 향후 필요 시 **3rd Party 보안 모니터링 툴**이 위치

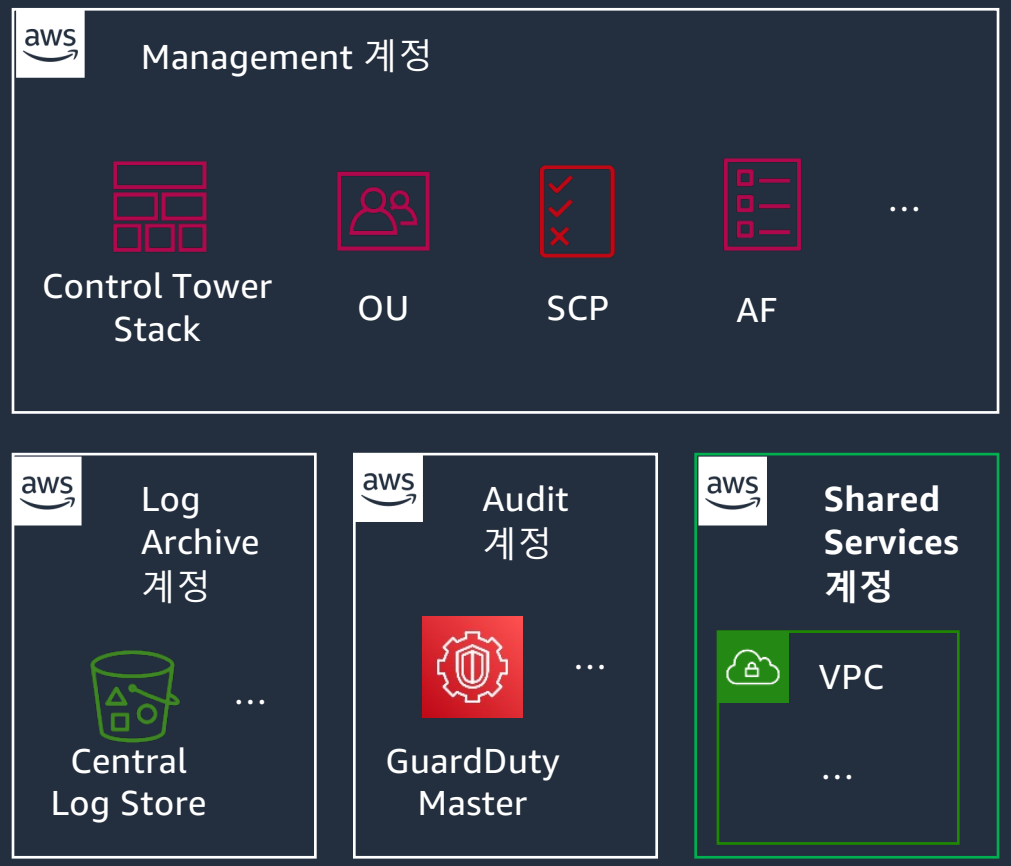
**특징**

- 보안 담당이 관리하며 제한적으로 접근함

**필수 계정**

# AWS Control Tower Additional Account – Shared Services

**Shared Services 계정**은 전사 공용 인프라가 위치하는 계정으로 사용할 수 있습니다.




**목적**

- **전사 공용 서비스 용도 – 개별 애플리케이션 공통 컴포넌트를 위한 용도는 아님**
- **Shared Services VPC** – 공용 서비스 중 VPC 기반으로 구현되는 컴포넌트들을 위한 VPC가 위치
- **Active Directory** – AWS 콘솔에 대한 SSO를 Microsoft AD 통해 구성될 경우 관련 리소스 위치
- **DNS** – Hybrid DNS에 필요한 리소스가 위치
- **배포 관련 공용 서비스**가 구성될 경우 관련 리소스가 위치 (Golden AMI, 전사 CI/CD 파이프라인)

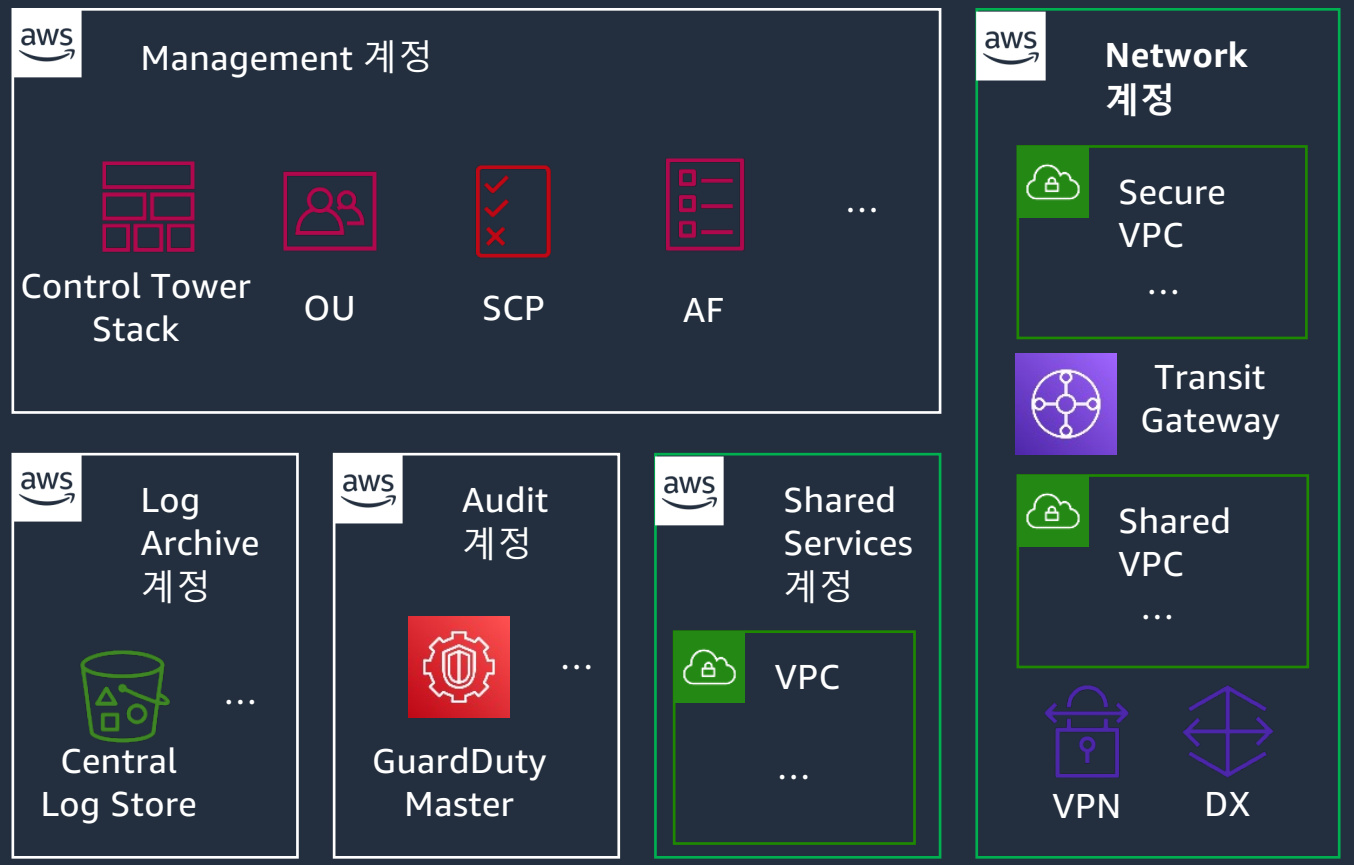
**특징**

- 필수 인원만 제한적으로 접근함
- 필요 시 On-prem DC와 연결됨

**추가 계정**

# AWS Control Tower Additional Account – Network

**Network 계정**은 전사적 네트워크 연결성을 제공하기 위해 사용할 수 있습니다.




**목적**

- **<u>전사적 연결성 제공</u>** – 외부 (North), on-prem (South), AWS 내(East-West)를 모두 포함함
- **Secure VPC** – 외부로부터 들어오는 North 트래픽이 거치는 VPC임
- **Shared VPC** – (Optional)서비스/어플리케이션 Account로 Subnet을 Share하는 VPC임
- **Transit Gateway** – AWS VPC 간 연결성 제공
- **VPN 및 DX** – on-prem DC를 오가는 South 트래픽에 대한 보안성을 갖춘 연결 제공

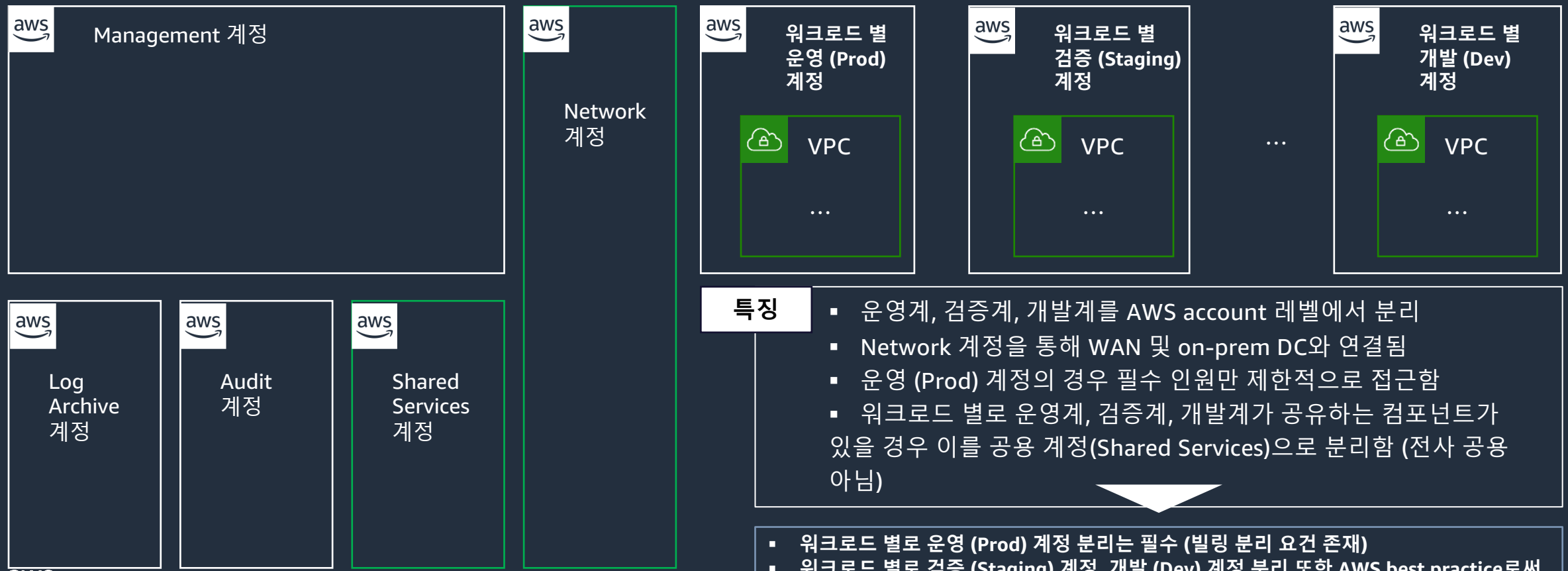
**특징**

- 필수 인원만 제한적으로 접근함

**추가 계정**

# 개별 워크로드 용 Account

워크로드(애플리케이션) 별로 **운영, 검증 및 개발 계정**을 생성하는 것이 AWS best practice입니다



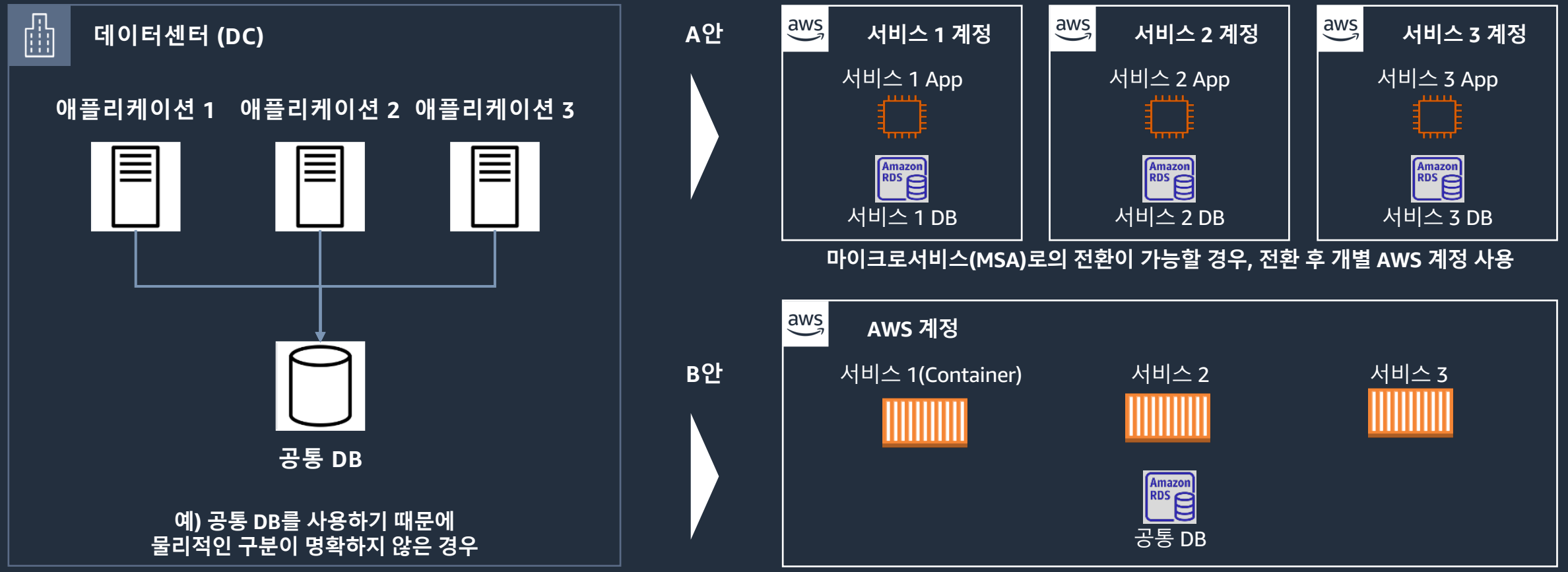

**특징**

- 운영계, 검증계, 개발계를 AWS account 레벨에서 분리
- Network 계정을 통해 WAN 및 on-prem DC와 연결됨
- 운영 (Prod) 계정의 경우 필수 인원만 제한적으로 접근함
- 워크로드 별로 운영계, 검증계, 개발계가 공유하는 컴포넌트가 있을 경우 이를 공용 계정(Shared Services)으로 분리함 (전사 공용 아님)

- **워크로드 별로 운영 (Prod) 계정 분리는 필수 (빌링 분리 요건 존재)**
- **워크로드 별로 검증 (Staging) 계정, 개발 (Dev) 계정 분리 또한 AWS best practice로써 강력히 권장됨**
- **공용 (Shared Services) 계정은 해당 워크로드에서 필요 시 사용 권장됨**

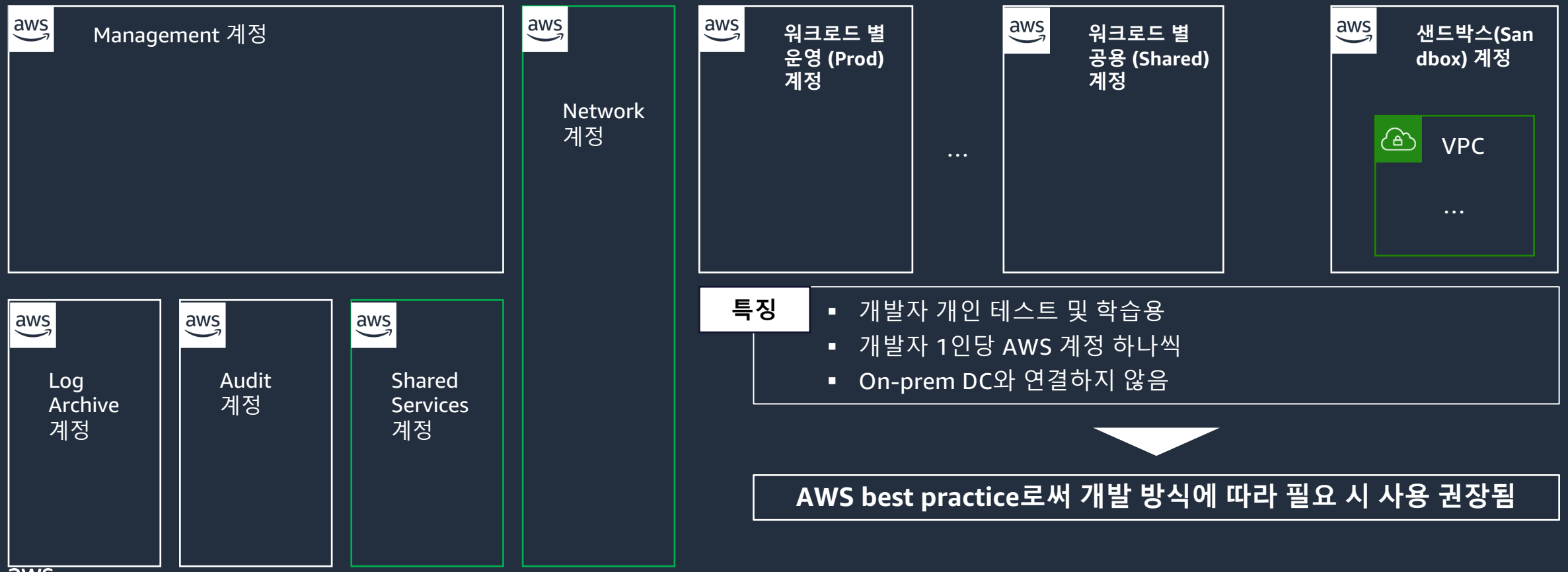
# 개별 워크로드 용 Account

AWS 이관 시 MSA 전환 여부에 따라 계정 분리 여부를 결정할 수 있습니다.



# Sandbox - 개발자 별 Account

필요 시 **개발자가 사용하는 AWS 계정**도 관리하는 것이 AWS best practice입니다.



# AWS Control Tower Multi-Account 요약

초기 구성은 다음과 같은 Account 구조가 권고되며, 이후 **필요 시 유연하게 확장 구성을** 권장 드립니다.

| 계정 | 목적 | 사용주체 |
|---|---|---|
| **Management 계정-필수** | ▪ **Control Tower 서비스**, **OU**, **SCP** 구성 및 운영 관리 | ▪ **기획, 인프라, 보안** (단, 로그인 최소화) |
| **Log Archive 계정 - 필수** | ▪ 보안 로그의 중앙 적재를 위한 **S3 버킷** 위치 | ▪ **보안** (단, 로그인 최소화) |
| **Audit 계정 – 필수** | ▪ 보안 관련 AWS 서비스 (**GuardDuty**) 및 향후 **보안 툴** 도입 시 해당 툴 설치 | ▪ **보안** |
| **Shared Services 계정** | ▪ 전사 공용 성격의 애플리케이션 구성 및 향후 보안 목적 외 툴 도입 시 해당 툴 설치<br>▪ **메타 관리 서버**, **형상/배포 서버** 위치 | ▪ **인프라** |
| **Network 계정** | ▪ **Secure VPC**, **접근제어 VPC** 위치<br>▪ **Transit Gateway**, **Direct Connect**, **VPN** 위치 | ▪ **보안*** – Secure VPC, 접근제어 VPC, VPN<br>▪ **인프라*** – Transit Gateway, Direct Connect |

# AWS Control Tower Multi-Account 요약 – 전체

| 구분 | 계정 이름 | 필수 여부 | 개수 | 생성 방식 | 생성 시기 |
|---|---|---|---|---|---|
| 핵심 (Core) | **Management** | 필수 | **1** | 기본 포함 | 기본 포함 |
| | **Log Archive** | 필수 | **1** | 기본 포함 | 기본 포함 |
| | **Audit** | 필수 | **1** | 기본 포함 | 기본 포함 |
| 추가 | **Shared Services** | 선택 | **1** | AF* 실행 | 필요시 |
| | **Network** | 선택 | **1** | AF* 실행 | 필요시 |
| 워크로드 별 | **운영 (Prod)** | 필수 | **N**<br>(애플리케이션 수) | AF* 실행 | 필요 시 수시로 |
| | **검증 (Staging)** | AWS BP – 권장 | **N**<br>(애플리케이션 수) | AF* 실행 | 필요 시 수시로 |
| | **개발 (Dev)** | AWS BP – 권장 | **N**<br>(애플리케이션 수) | AF* 실행 | 필요 시 수시로 |
| 개발자 별 | **샌드박스 (Sandbox)** | AWS BP – 선택 | **N**<br>(개발자 수) | AF* 실행 | 필요 시 수시로 |

# AWS Organizations Unit 설계

# Recommended OUs and accounts (1/3)


https://docs.aws.amazon.com/ko_kr/whitepapers/latest/organizing-your-aws-environment/recommended-ous-and-accounts.html

# Recommended OUs and accounts (2/3)

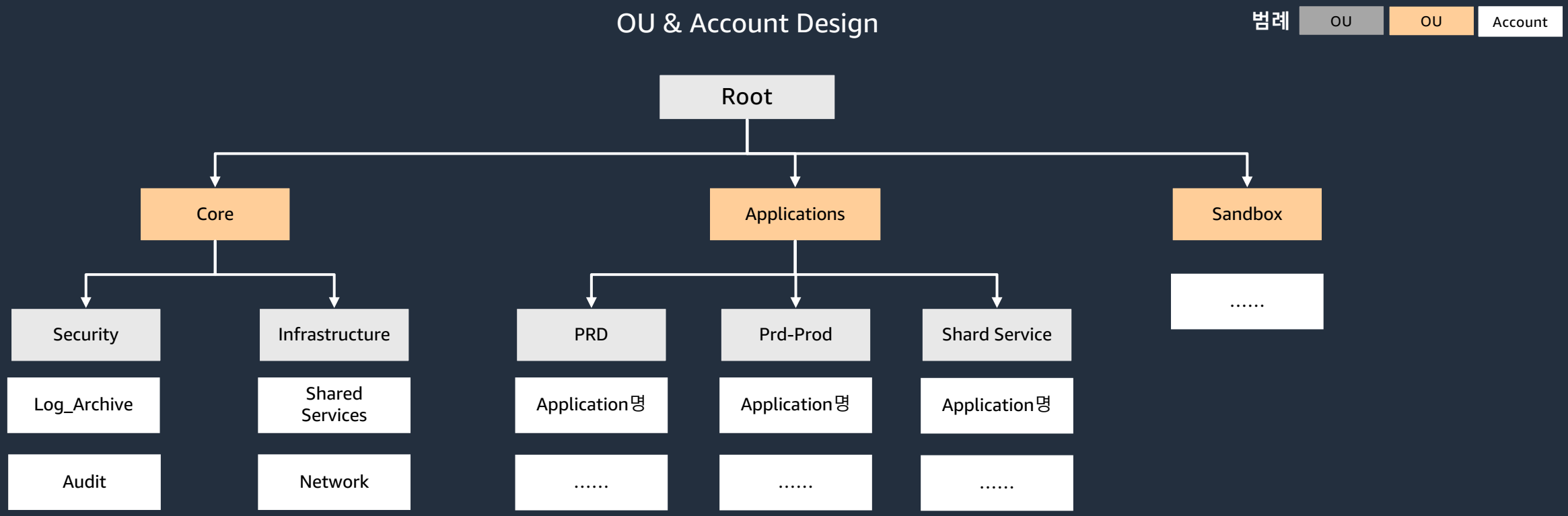
| OU | OU 설명 | account | Account 설 명 |
|---|---|---|---|
| Security | 기본 OU, 보안 관련 계정 관리 | Log archive | 조직의 모든 계정에서 수집되고 주로 보안, 운영, 감사 및 규정 준수팀에서 사용되는 로그 데이터를 통합 하는 계정(AWS CloudTrail, AWS Config 등) |
| | | Security tooling | 보안 서비스, 도구 등을 관리하는 계정 ( AWS Security Hub, Amazon GuardDuty, AWS Firewall Manager, Amazon Detective, Amazon Inspector, IAM Access Analyzer 등) |
| Infrastructure | 기본 OU, 공유 인프라 서비스 관리 | Network | 네트워킹 리소스를 관리하는 계정(Amazon VPC, AWS Transit Gateway, Amazon Route 53, AWS Firewall Manager, AWS Network Firewall 등) |
| | | Operations tooling | 운영 자동화 활동을 관리 하는 계정(AWS Systems Manager, AWS Systems Manager 등) |
| | | Shared Services | 전체 조직에서 공유되는 서비스를 관리하는 계정(AWS IAM Identity Center, AWS License Manager, SSH or RDP Bastion s 등) |
| Sandbox | 허용 가능한 정책에 따라 AWS 서비스와 기타 도구 등을 자유롭게 테스트 허용 | developer | 임시적인 리소스 환경, AWS 서비스,및 인터넷접속 허용,기업 리소스 사용제한, 비공개데이터 사용 제한 |

# Recommended OUs and accounts (3/3)

| OU | OU 설명 | account | Account 설 명 |
|---|---|---|---|
| Workloads | 상용, 비상용 환경 포함 비즈니스별 워크로드 계정 관리 | Prod, Stg, Test | 상용, 스테이징, 테스트 계정 |
| Policy Staging | 정책을 적용하기 전 테스트 계정관리 | policy test | IAM Role, SCPs, Tag 규칙 등 적용 전 사전 영향도 테스트 |
| Suspended | 일시적 또는 영구적으로 사용을 중지해야 하는 계정 등을 임시보관 | suspended account | AWS 리소스를 사용할 수 없도록 SCPs를 이용하여 제한하며 애플리케이션 레벨에서도 접근제한을 설정하여 보안 및 비용을 제어하고 정지된 계정에 Tagging을 설정하여 자동화된 종료처리 및 내부 추적,감사에 적용가능 |
| Individual Business Users | AWS 리소스를 외부에서 사용하는 팀, 사용자들을 관리 | business team account | 비즈니스, 마켓팅을 위해 파트너와 S3를 통해 자료 및 BI Tool(AWS Quick Sight 등)공유함. 보안을 위해 SCPs, IAM 퍼미셧 설정하여 사용제어 |
| Exceptions | Workloads OU 적용 보안 정책에 대한 예외가 필요한 계정관리 | policy exception account | 적용 계정 최소로 관리 |
| Deployments | Workloads 리소스 빌드, 검토,배포 계정 관리 | CI/CD management | AWS 내부서비스, 외부 CI Jobs, Cd build stage 관리 |
| Transitional | 조직으로 이동하는 기존 계정 및 워크로드를 임시 보관 하는 영역 | moving accounts | 예: AWS Account를 사용하고 있는 회사 인수, 기존 사용하고 있는 AWS Account 들을 이전, 3rd Party 에서 사용하고 있는 Account 이전 |
| Business Continuity | 계정간 재해 복구 전략 구현용 | disaster recovery accounts | workload, data 중요도, RPO, RTO 고려 Account 분리 |

# OU & Account Design (1/2)

- AWS account 그룹별로 SCP를 적용하기 위하여 고객의 상황에 맞는 전략으로 OU와 계정을 구성 합니다.



# OU & Account Design (2/2)

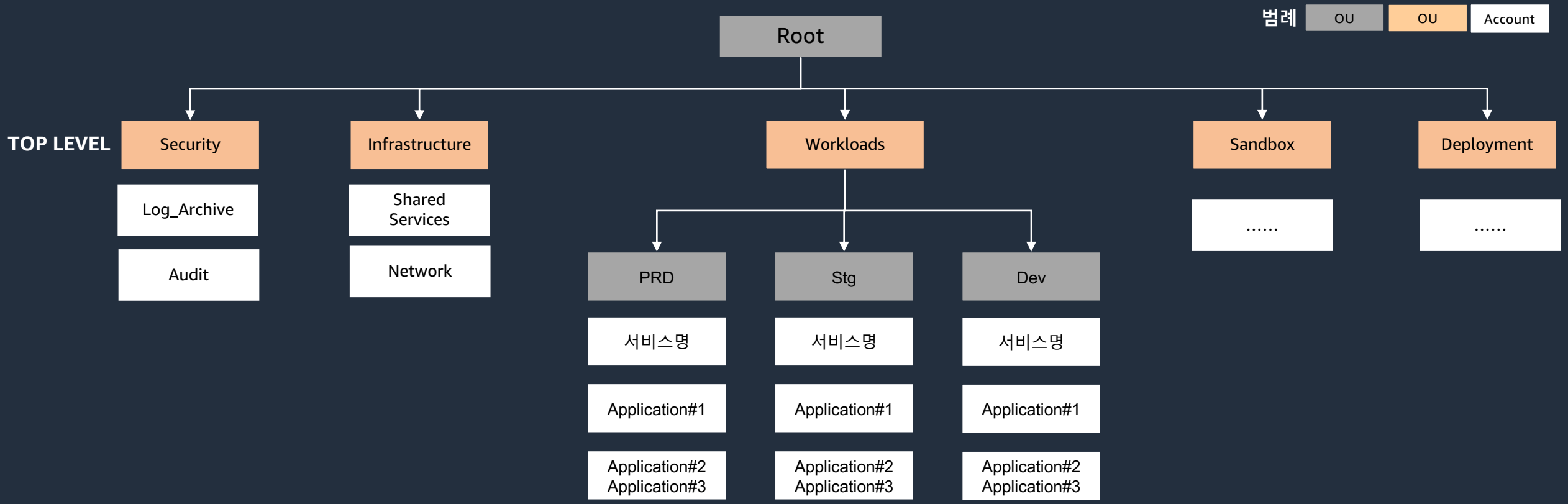
- 각 계정의 역할을 쉽게 이해를 할 수 있도록 Naming rule을 적용하여 계정을 생성합니다.

| Account Name | OU Name | Role | Email | Account ID | Org ID |
|---|---|---|---|---|---|
| Payer | Root | Master Billing OU & Account | TBD | TBD | TBD |
| Log_Archive | Security | Central security log aggregation account | *[aaa_logarchive]*@aaabbb.com | TBD | TBD |
| Audit | Security | Central security account for GuardDuty, Security Notification, 3'rd party monitoring, etc | *[aaa_security]*@aaabbb.com | TBD | TBD |
| Shared Services | Infrastructure | Central Shared Services/platform workloads | *[aaa_shared]*@aaabbb.com | TBD | TBD |
| Network | Infrastructure | Central networking resources. DX, TGW, VPN, Secure VPC, etc. | *[aaa_network]*@aaabbb.com | TBD | TBD |
| *[Application name_PRD]* | *[Applications:PRD]* | Production workloads for applications | *[application name_prd]*@aaabbb.com | TBD | TBD |
| *[Application name_STG]* | *[Applications:STG]* | Staging workloads for applications | *[application name_stg]*@aaabbb.com | TBD | TBD |
| *[Application name_SHR]* | *[Applications:SHR]* | Shared services account for applications | *[application name_shr]*@aaabbb.com | TBD | TBD |
| *[Sandbox_test]* | Sandbox | Development Account | *[sandbox_test]*@aaabbb.com | TBD | TBD |

# Organizational units (OUs) Design 전략 (1/4)

**OU는 유사한 요구 사항을 가진 계정에 공통의 중요한 정책을 더 쉽게 적용할 수 있도록 구성할 수 있는 방법을 제공**

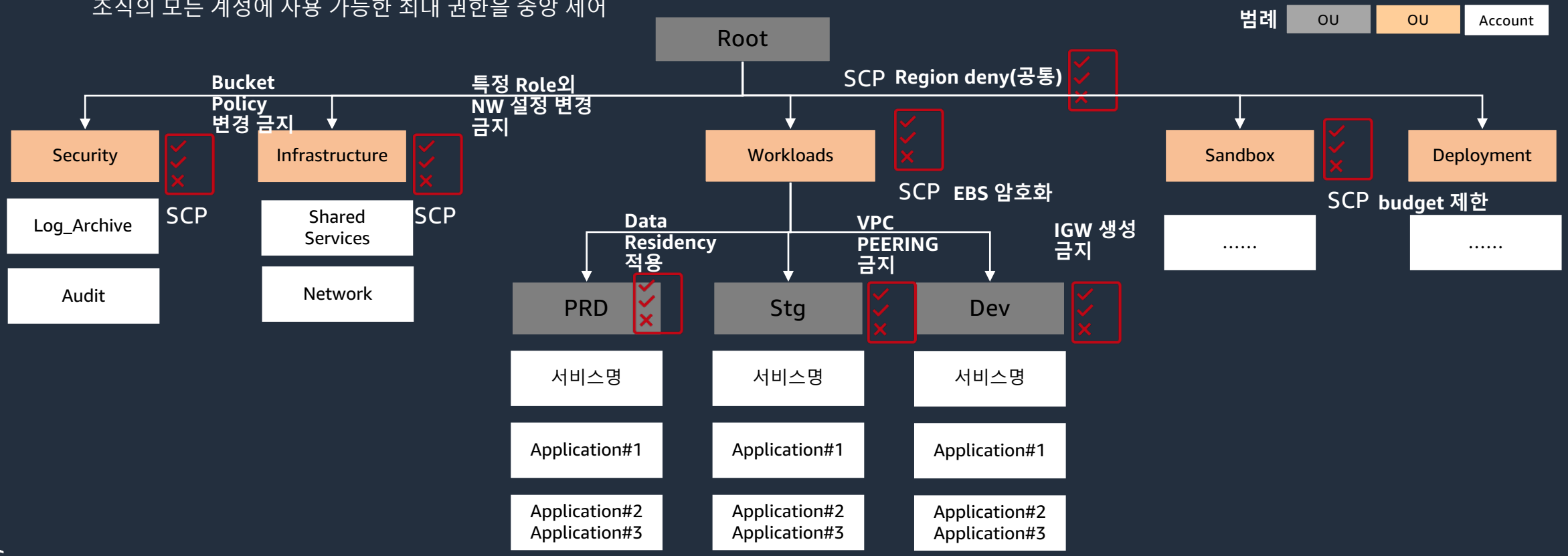
- 비슷한 기능을 가진 계정들을 명확한 의미를 가진 Top-Level OU밑으로 그룹화 ( AWS Recommended OU 권고안)



# Organizational units (OUs) Design 전략 (2/4)

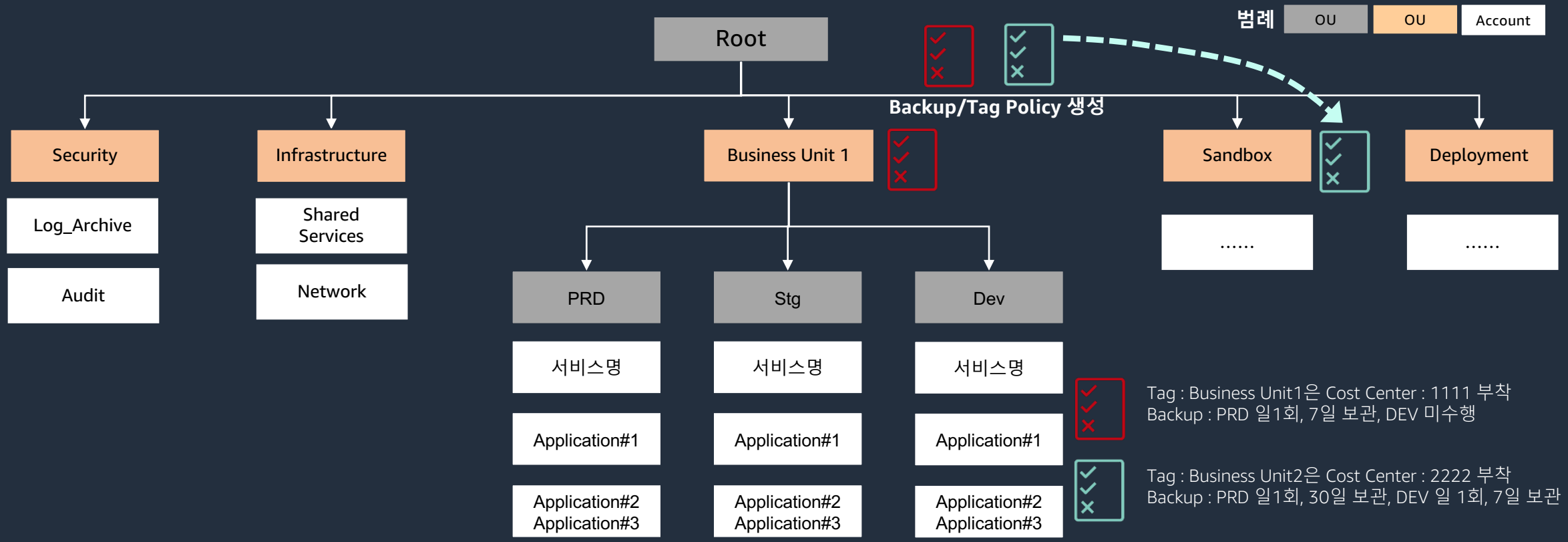
- **공통 Policy 적용 - SCP(Service Control Policy) 권한 정책 적용**

※ SCP : 조직의 권한을 관리하는 데 사용할 수 있는 정책,
조직의 모든 계정에 사용 가능한 최대 권한을 중앙 제어



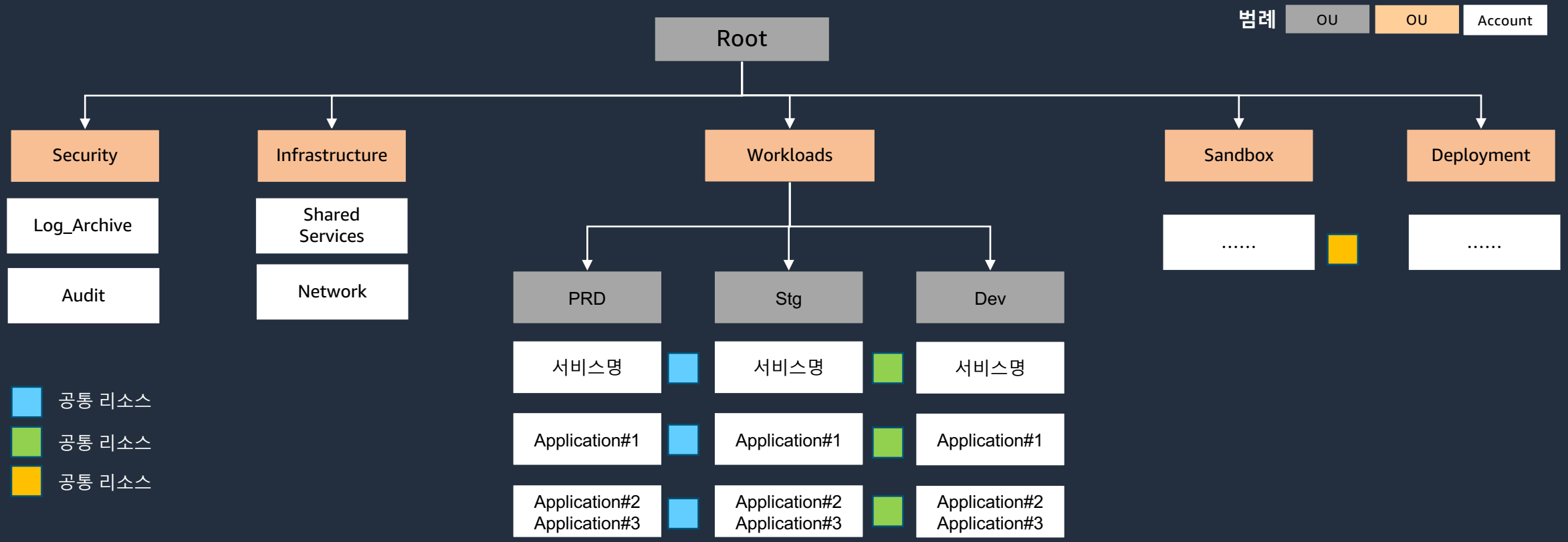
# Organizational units (OUs) Design 전략 (3/4)

- 공통 Policy 적용 - Management/Governance 정책 적용 단위 (ex. Tag Policy , Backup Policy)



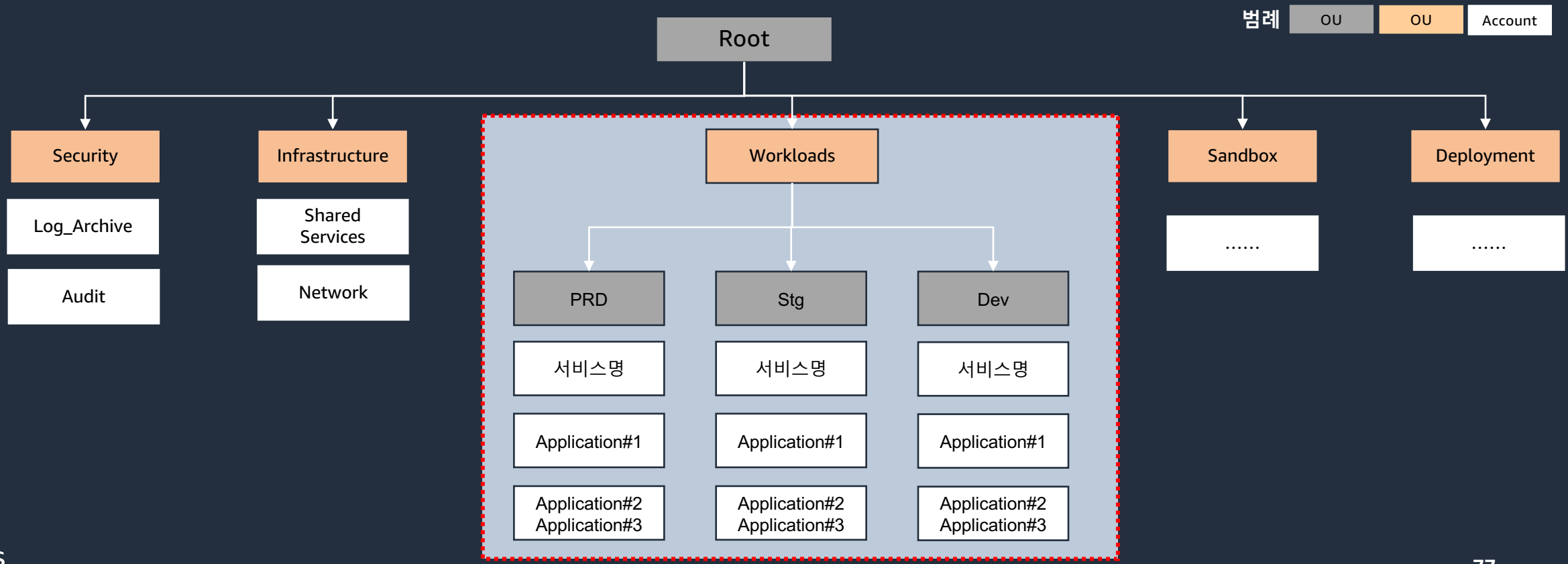
# Organizational units (OUs) Design 전략 (4/4)

- 공통 Resource (Security Group,Transit GW,Firewall Rule 등) 를 공유(Share) / 생성(Provisioning) / 관리 편의를 위한 단위로 사용



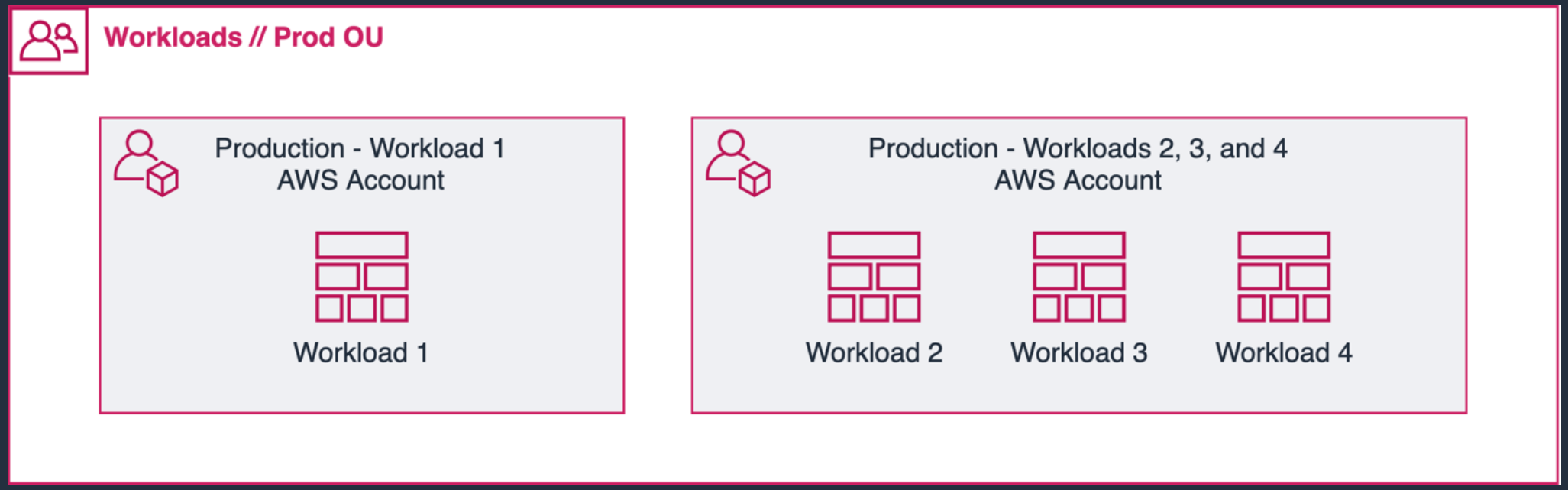
# Workload-oriented OUs Design (1/3)

- **Workload 를 위한 OU 정의하기 위한 추가 고려사항**



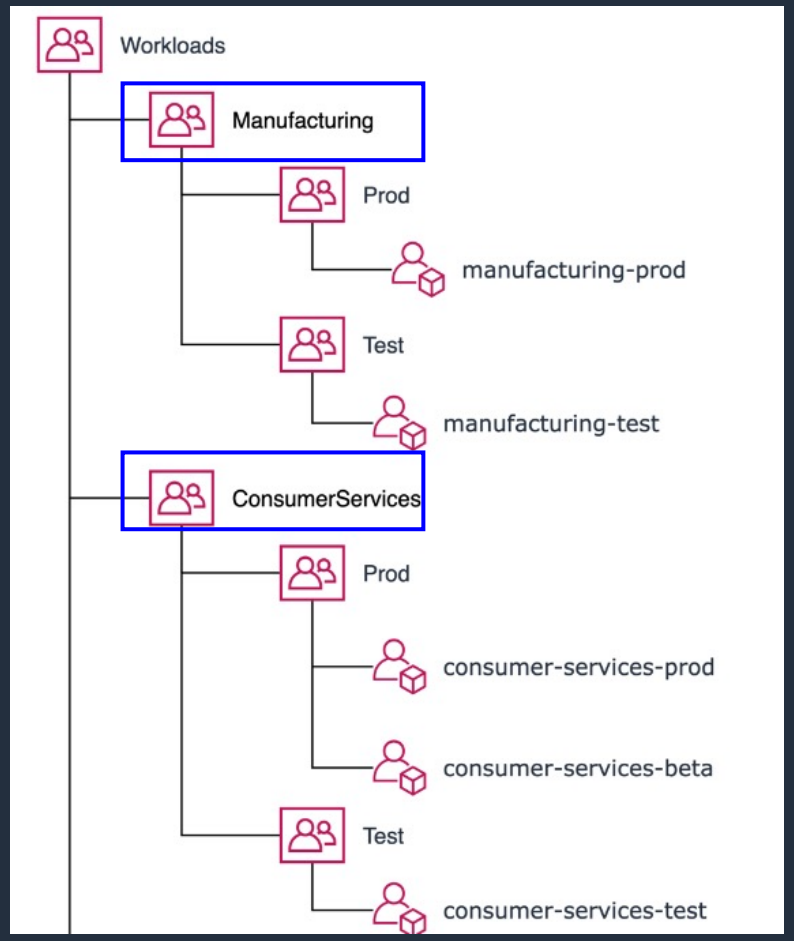
# Workload-oriented OUs Design (2/3)

- 워크로드 Account 생성 시, 단일 워크로드별 전용 환경이거나, 여러 워크로드 유형이 단일 계정인 경우, 모두 가능 후자의 경우, 계정 Naming 개발 필요



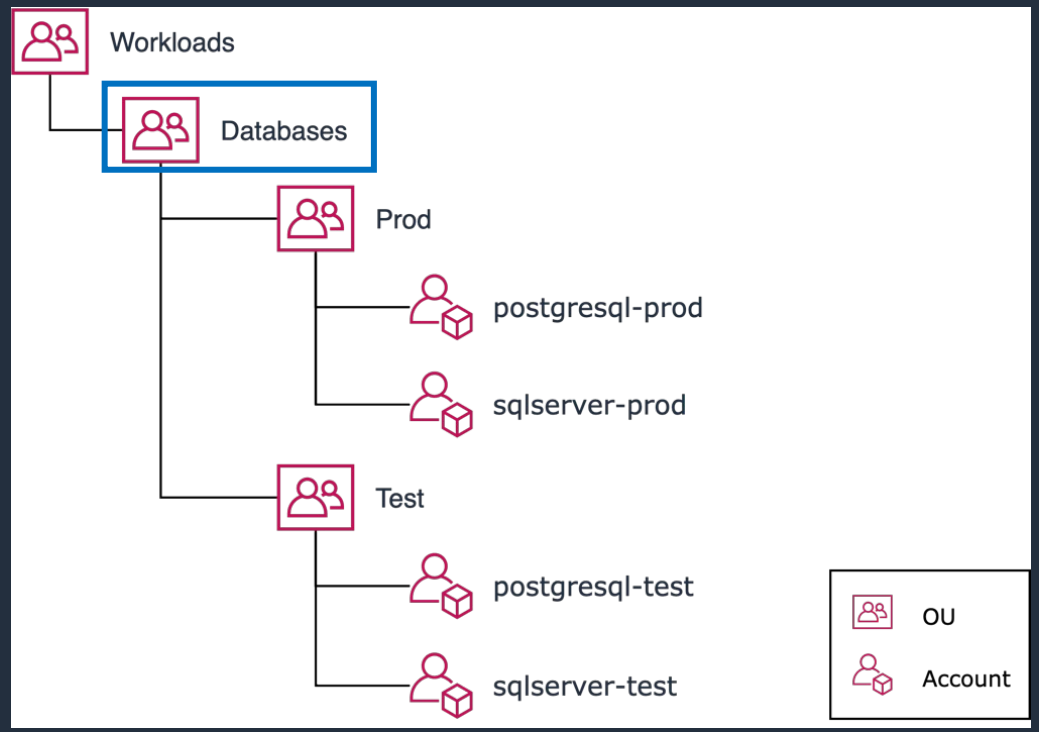
# Workload-oriented OUs Design (2/3)

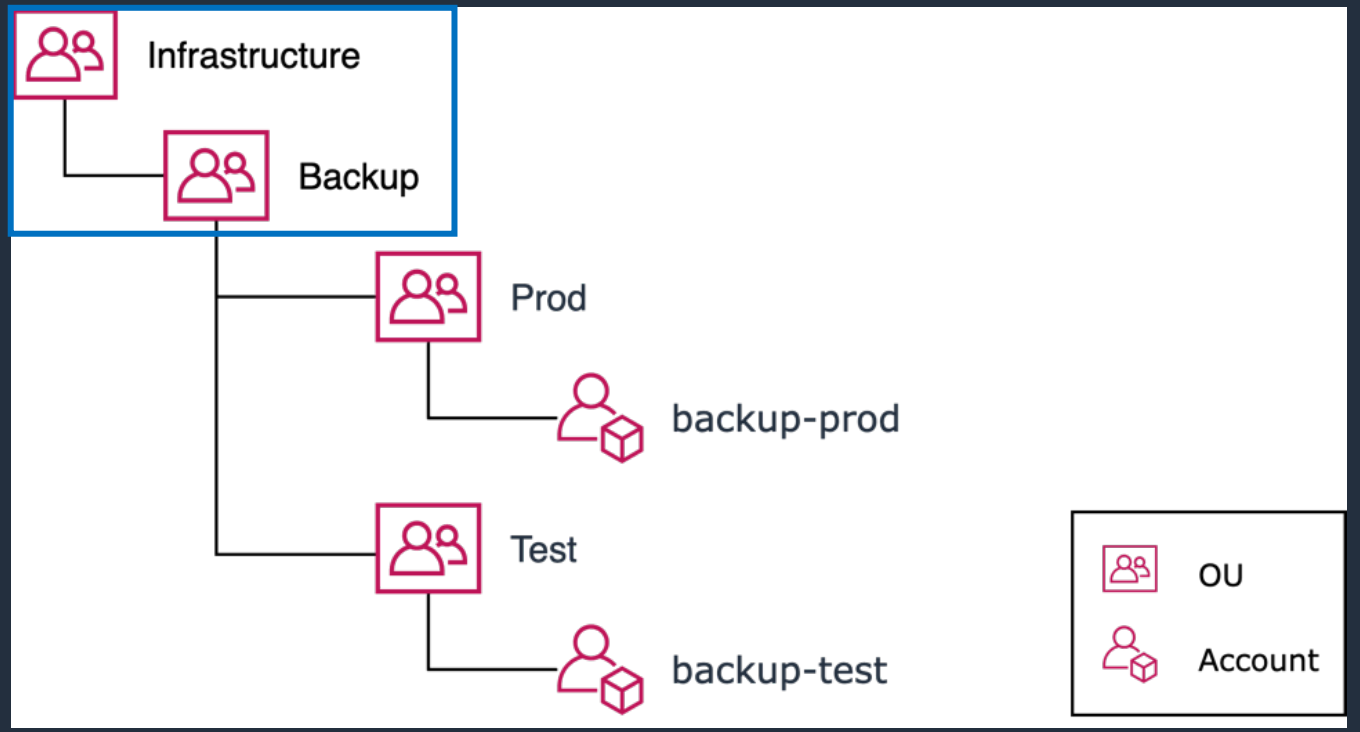
- 대규모 기업 환경에서 워크로드를 관리하는 자율적인 비즈니스 단위(BU) 가 있을 경우,



# Workload-oriented OUs Design (3/3)

- 전사적 관점의 특정 IT기능을 담당하도록 명확히 분리된 경우, 별도의 구별된 정책을 적용하기 위해 OU 분리 가능



# EKS workload 고려사항 (1/2)

• 관리 대상 서비스/워크로드/클러스터가 같은 Organizations 에 위치하는가?
✓ 워크로드와 환경에 따른 다양한 고려사항을 종합적으로 판단하여 결정

- **Security**

✓ by default, all pods, in all namespaces, can communicate to each other
✓ tenant 간 공유되는 정보(non-namespaced objects) 고려

- **Resource Shares**

✓ CPU, memory, network 자원의 공유
✓ 노드의 자원은 한정. 어떤 워크로드가 얼마만큼의 리소스를 사용할 것인가?
✓ 특정 tenant/워크로드의 리소스 점유가 다른 서비스에 영향을 주어도 되는가?

- **관리 조직 형태**

✓ 복수 서비스 중앙 관리 팀
✓ 개발, 운영 통합 DevOps팀

- **비용**

✓ Billing 분리 처리
✓ RI/SP 등 계정 별 할인 정책에 따른 구분

# EKS workload 고려사항 (2/2)

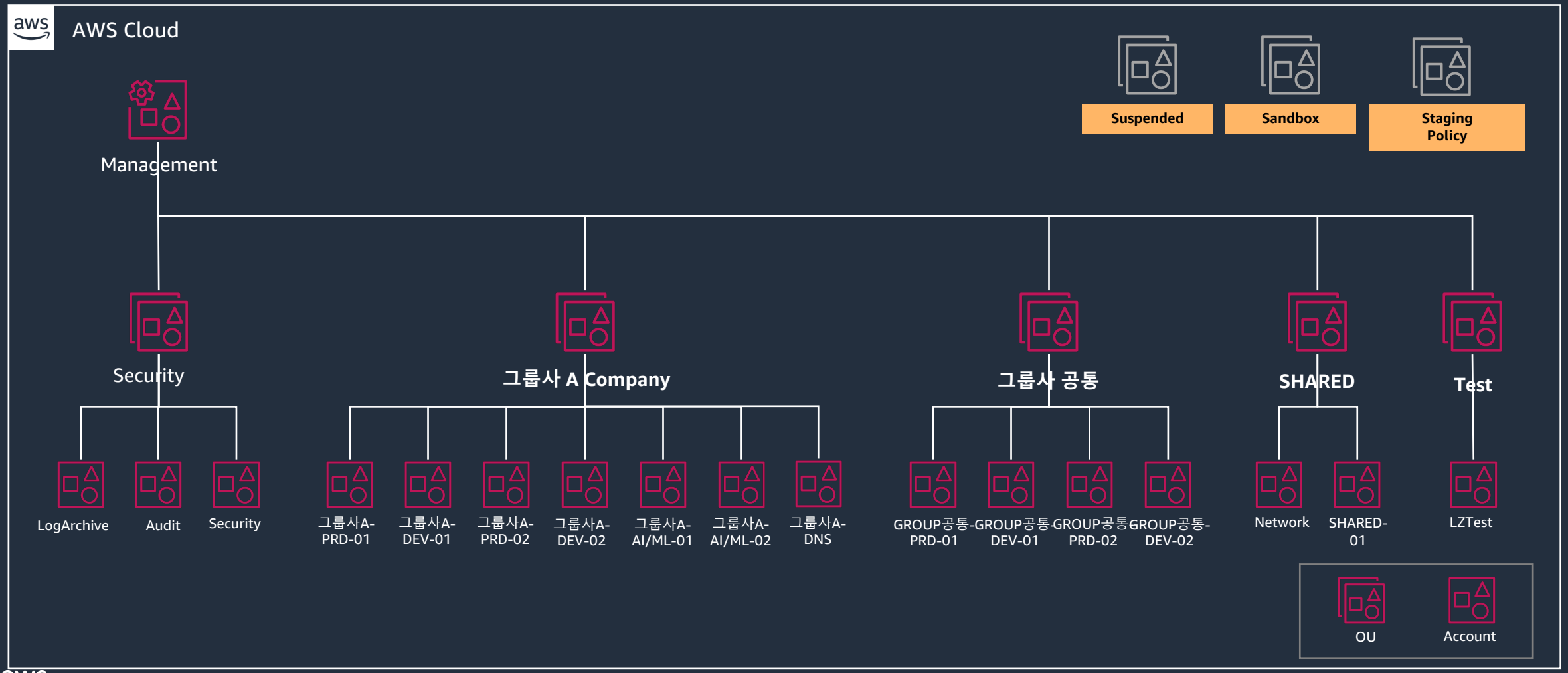
- Account 분리 원칙
✓ 장기적으로 트래픽 증가, 보안정책이 상이할 경우 서비스 별 계정 분리 권장
✓ 적은 트래픽, 자원 사용량, 통합 운영 고려 시 단일 Account로 사용 고려
✓ 단 SLDC레벨에서 Account 분리 권장 (DEV/STG & PRD)

- EKS 클러스터 분리 원칙
✓ SDLC 레벨에 대해 분리(DEV/STG/PRD 는 각각의 클러스터로 분리)

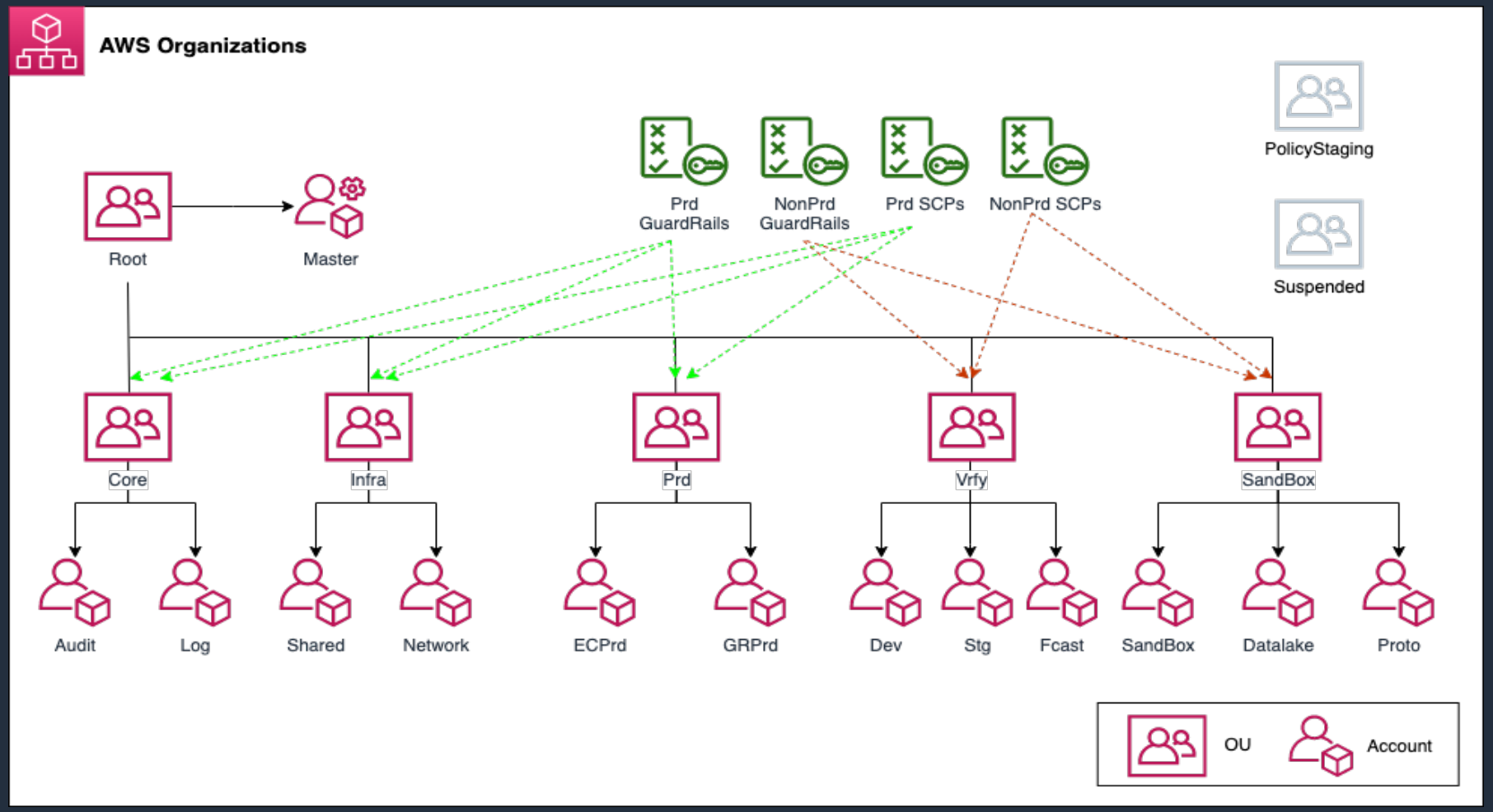
- Namespace 단위 분리 원칙
✓ 동일 클러스터 내 서비스 별로 각각의 namespace 로 구분

- 기타 고려사항
✓ Namespace 간 pod security 제어 고려
✓ Namespace 간 resource quota 제어 고려

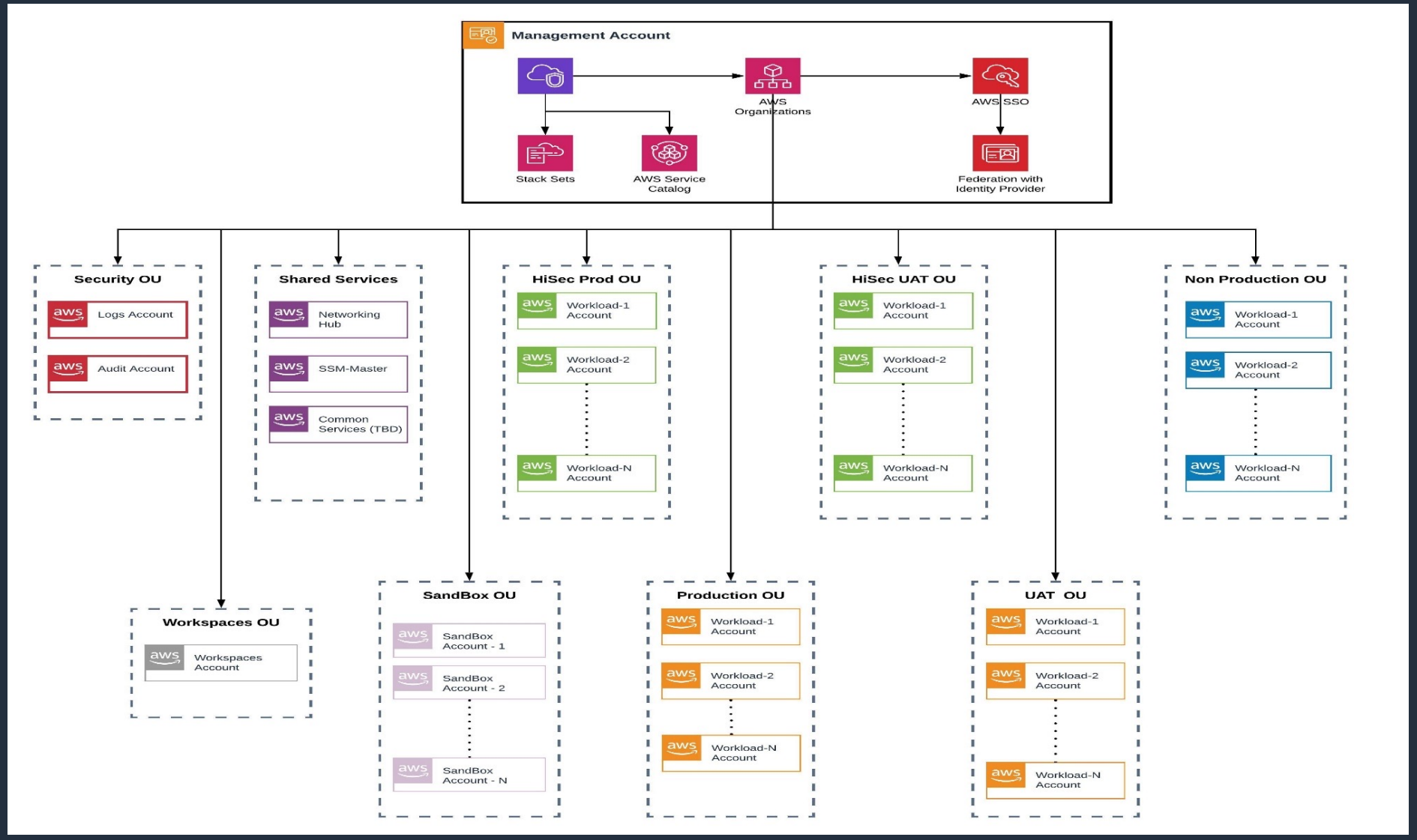
# 사례 **A**



# 사례 **B**



# 사례 C



# Thank you!