
AWS
re:Invent
DECEMBER 2 – 6, 2024 | LAS VEGAS, NV

SEC337-R

Scaling IAM: Advanced administration and delegation patterns

Swara Gandhi
Sr. Identity Solutions Architect
AWS

Jeremy Ware
Sr. Security Solutions Architect
AWS

Agenda

Challenges

- Scaling challenges
- Managing access
- Securing identities

Solution

- Delegation strategies
- Centralizing and automating IAM management
- Implementing permissions guardrails
- Permissions management tools

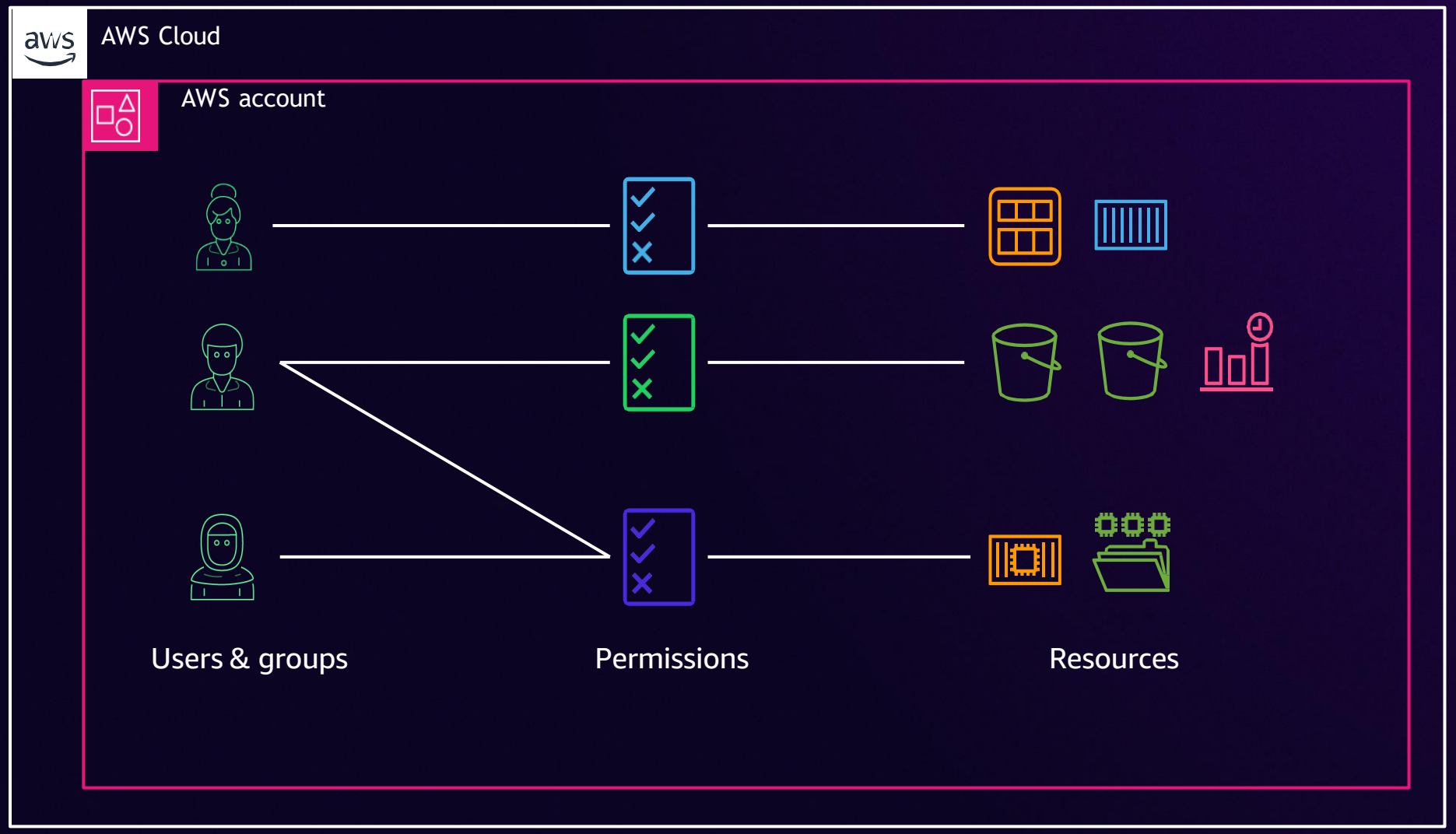
Assessing your IAM posture

- Where are you starting from?
- Tying it all together

Wrap up

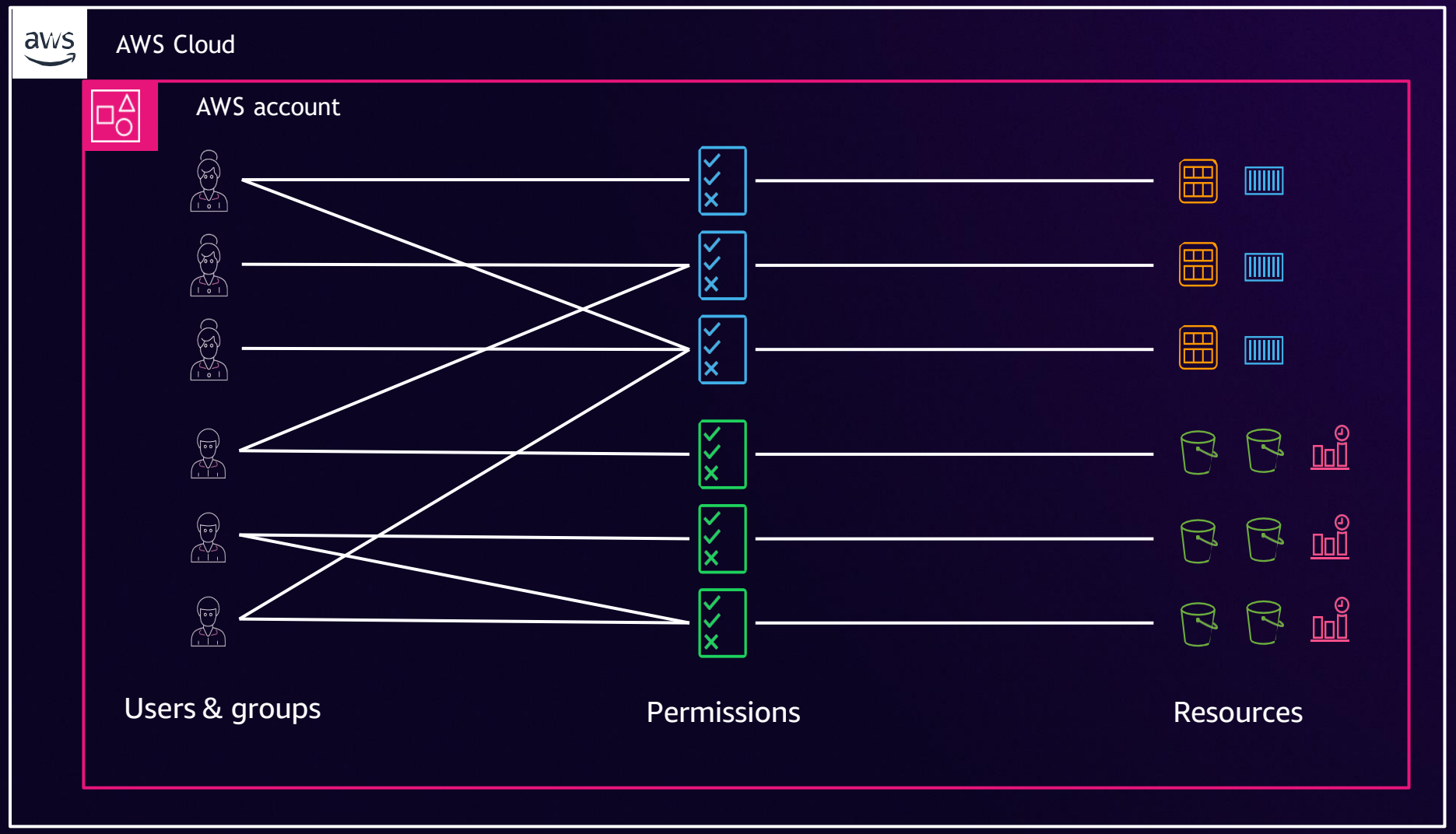
- Discussion
- Further reading material

Permissions to resources in one account



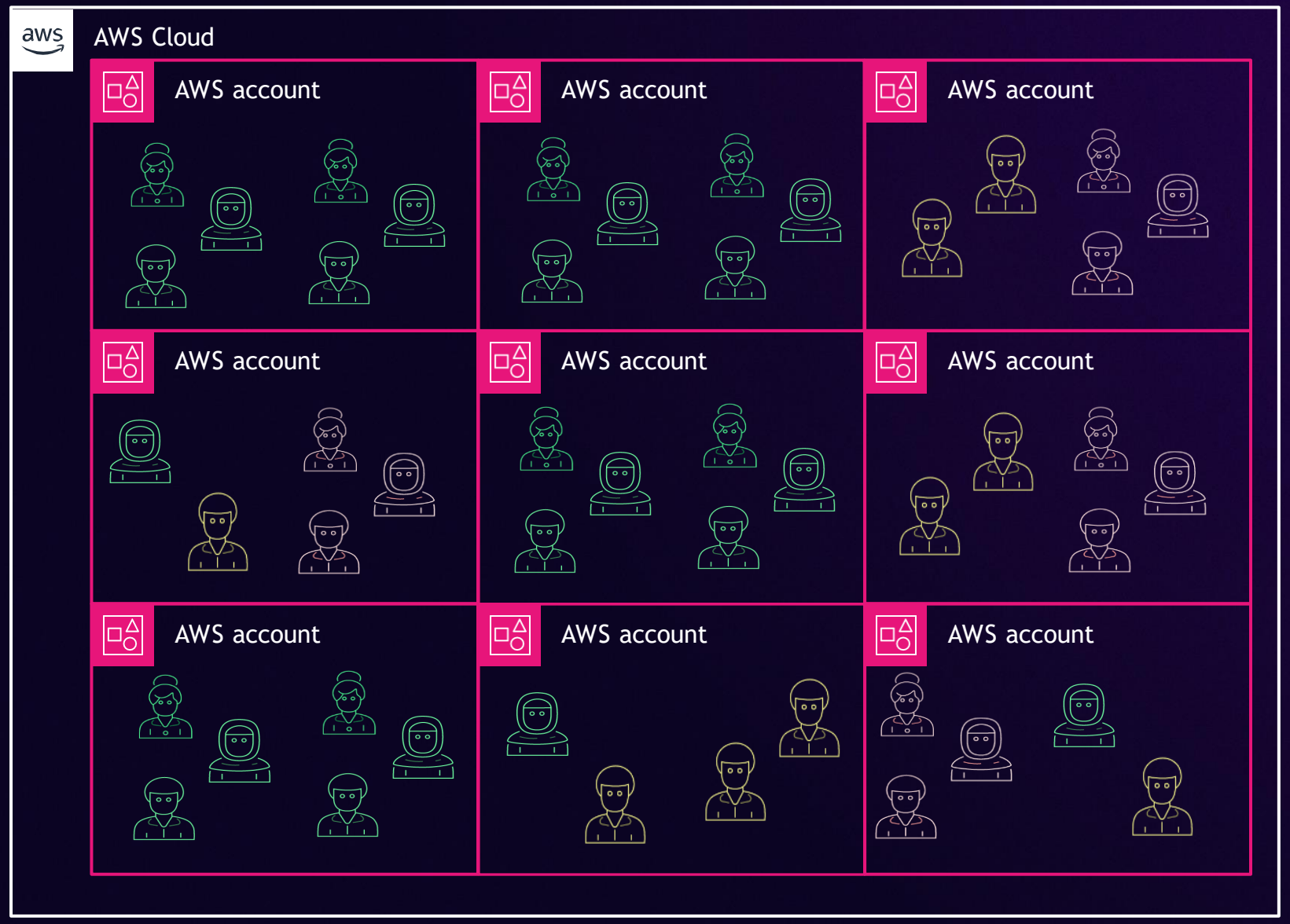
Central Security Team

More realistic scenario



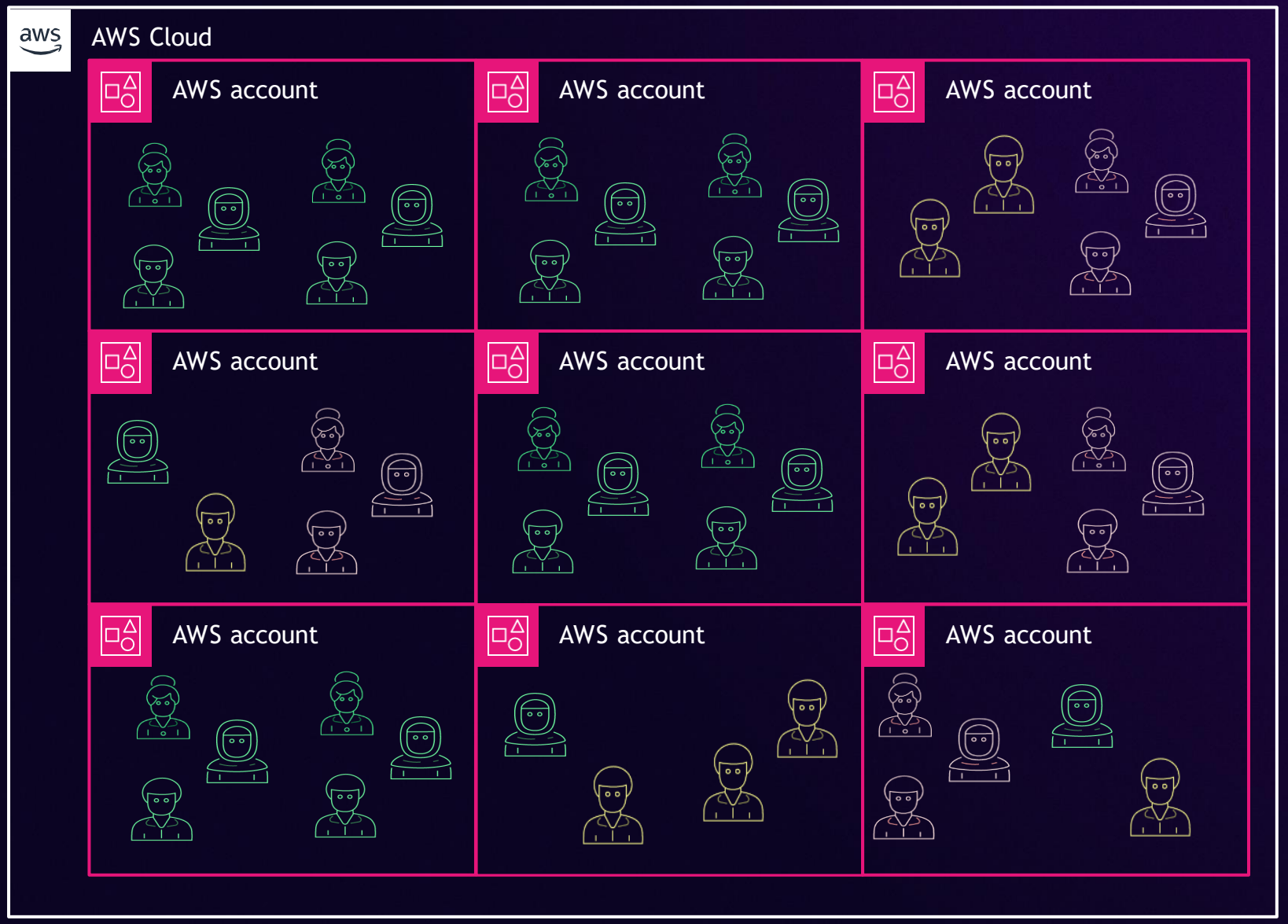
Central Security Team

A common customer scenario



Central Security Team

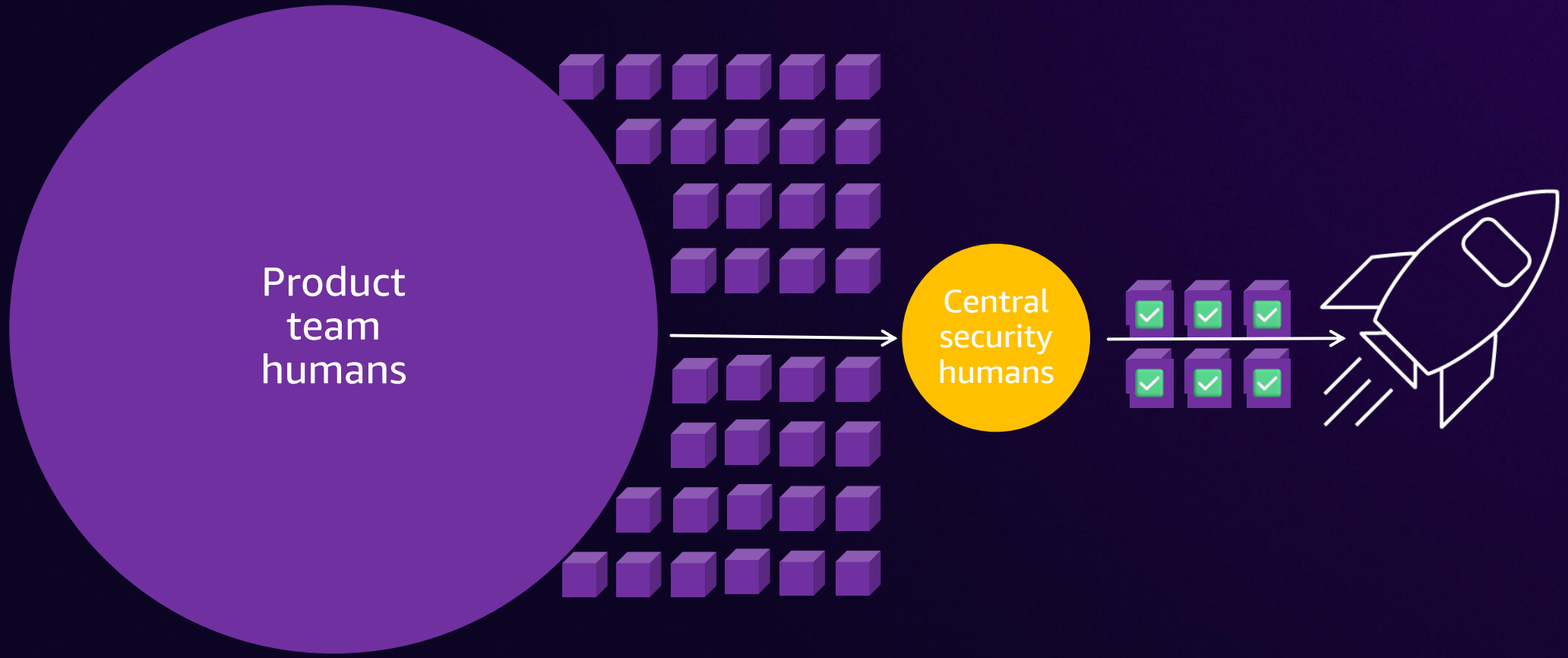
Central team managing access



Everyone needs to ask me for IAM permissions

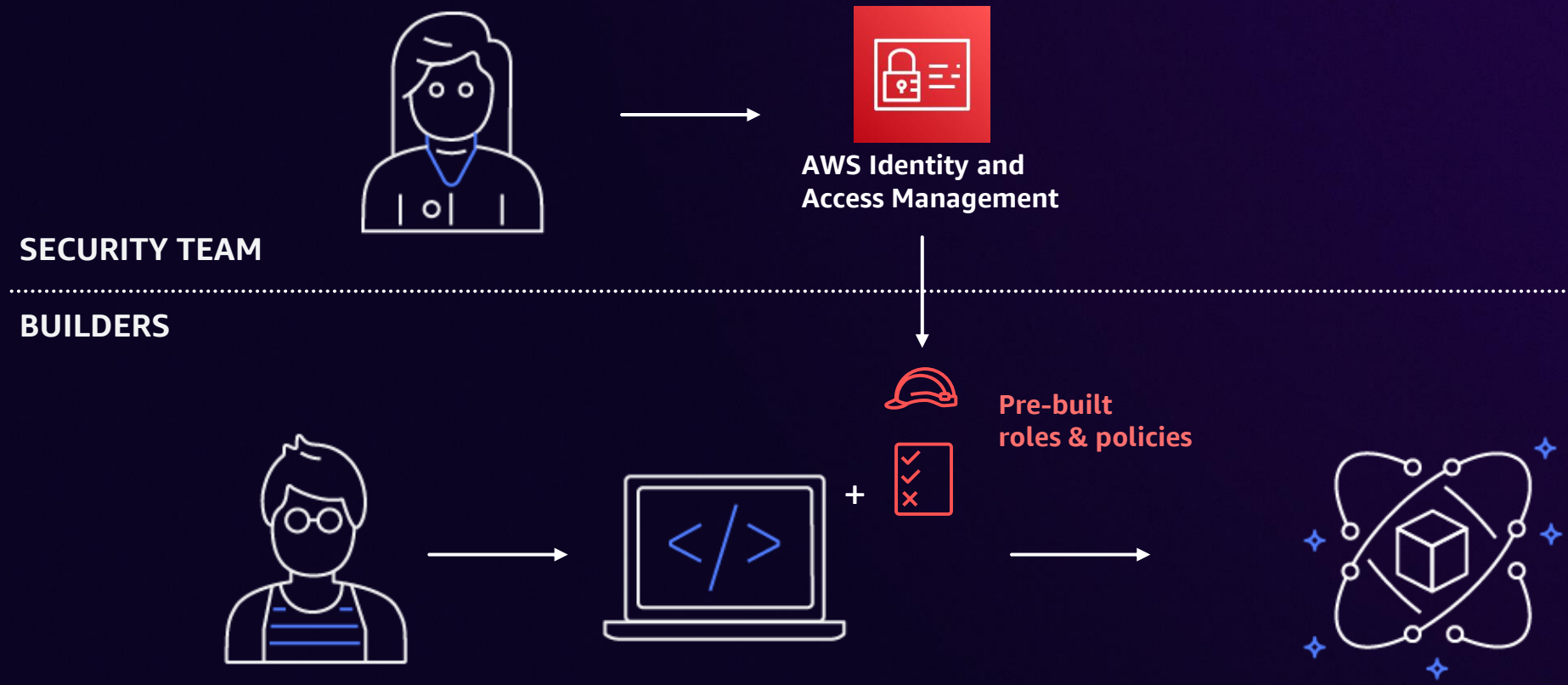
Central Security Team

Central team becomes a bottleneck



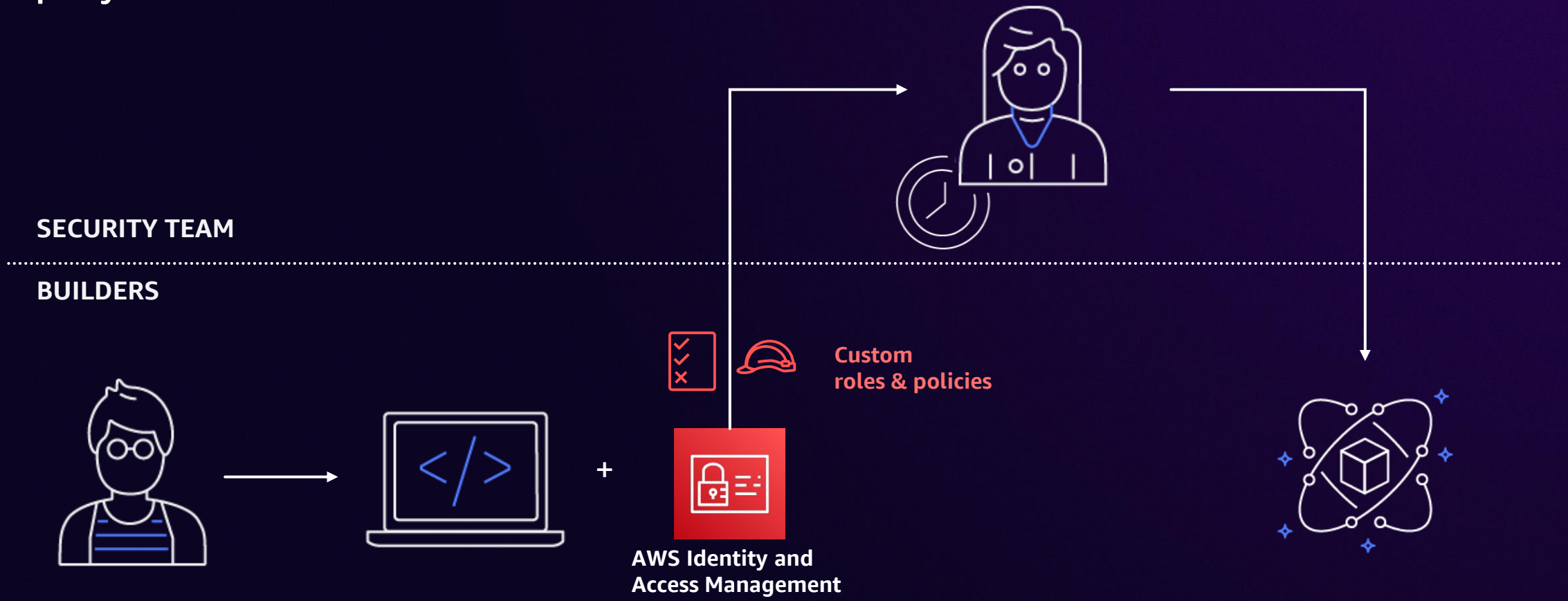
Centralized IAM Approach

Security team creates and deploys all IAM roles/policies; builders have no access to IAM

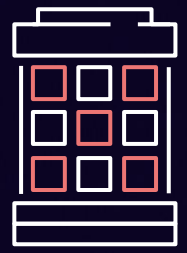


Gated IAM Approach

Builders codify the IAM roles and policies, but need Security to approve and deploy



Challenges



Lack of visibility

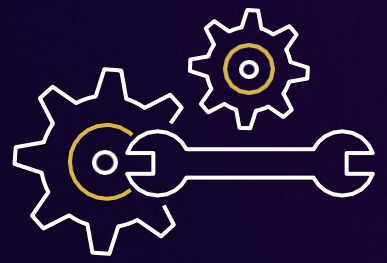
Who has access to what?

Identifying users with privileged actions

Scaling

Bottle-necked central team

Provisioning



Decentralized setup

Managing access

Operations

Security

Lack of governance and controls

User Experience

Agility to move quickly

Freedom for builders

Solution

Path to self-service

Productized approach to IAM

Centralize IAM Management with IdC

- Centrally manage ~~users & groups~~
- Centrally manage permissions

Automate

- Machine identity management
- Resource creation
- Account delivery with defined feature set

Delegation

- Role vending
- Policy validation

Security controls

- Permissions guardrails
- Base set of policies

Delegation

What do we mean by delegation?

Builder Bob

I need a role to run my application!

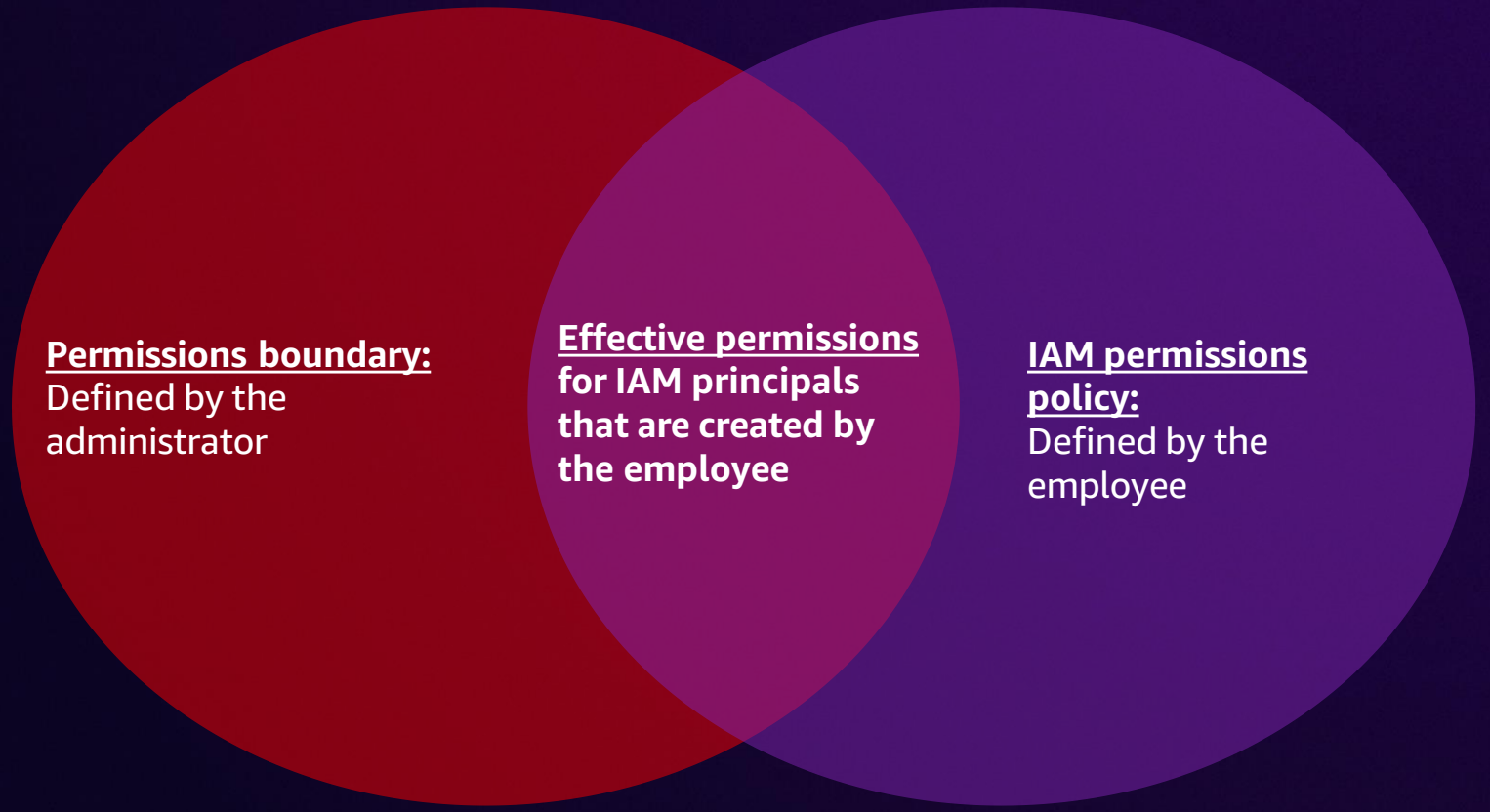
Security Susan

Create it yourself!

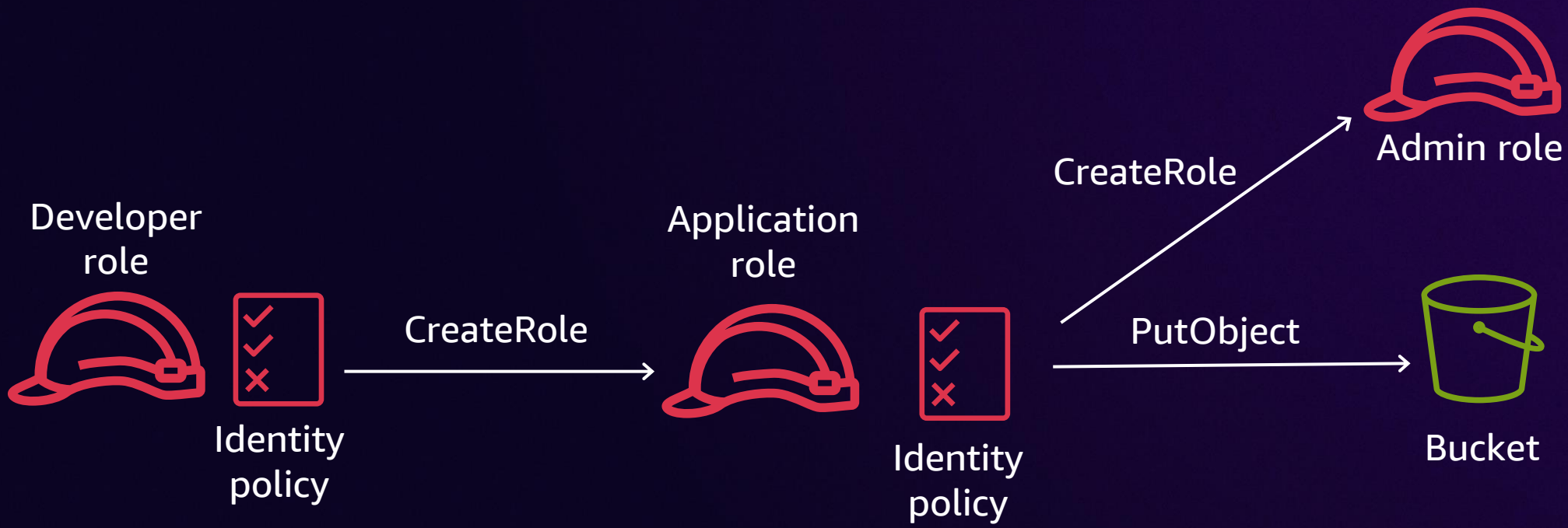
Permissions boundaries

Apply permissions boundaries to the **IAM roles created by developers**, rather than to the developers themselves

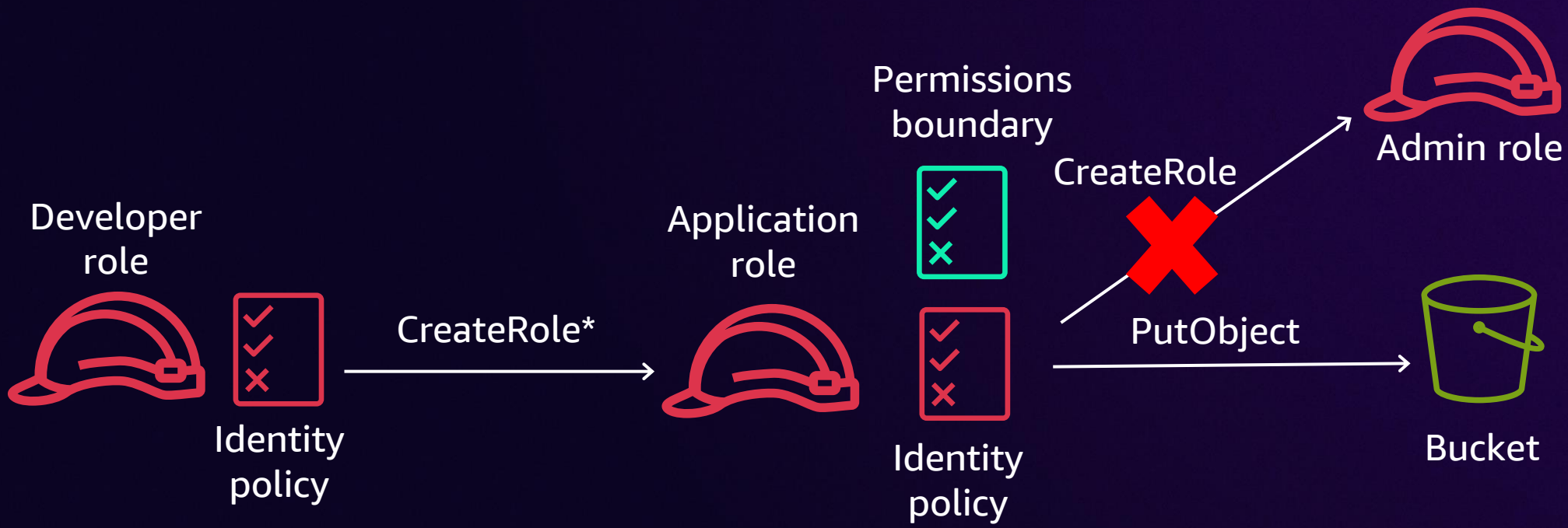
Group your **permissions boundaries into categories** that fit the scope of similar application functions (such as system automation and analytics)



Without permissions boundaries



With permissions boundaries



* But only if a permissions boundary is attached

Permissions boundary walkthrough

Account Admin Step 1: Create the permissions boundary only allowing certain AWS services (CompanyBoundary)

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": ”ServiceBoundaries",
         "Effect": "Allow",
         "Action": [
            ”s3:*",
            ”cloudwatch:*",
            ”ec2:*”
         ],
         "Resource": "*”
      }
   ]
}
```

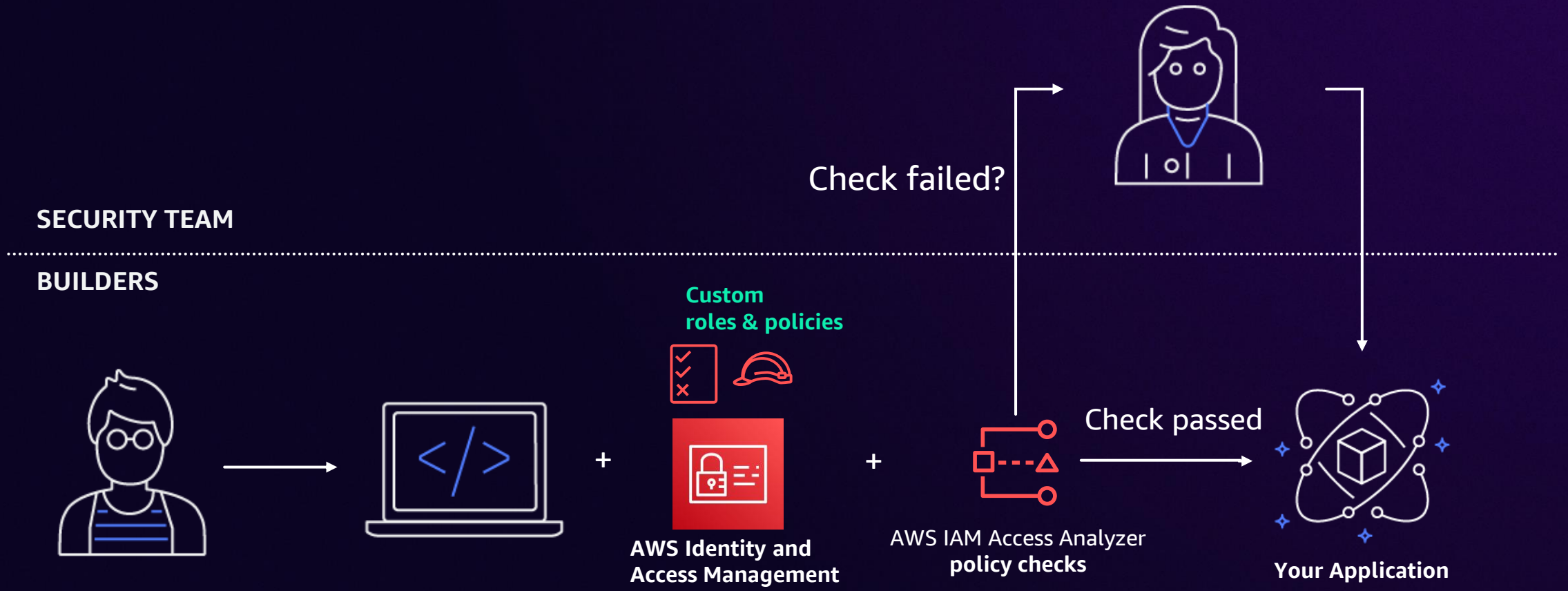
Permissions boundary walkthrough

Account Admin Step 2: Allow role creation, but only if CompanyBoundary is specified. Attach permission policy to delegated IAM Admin principal

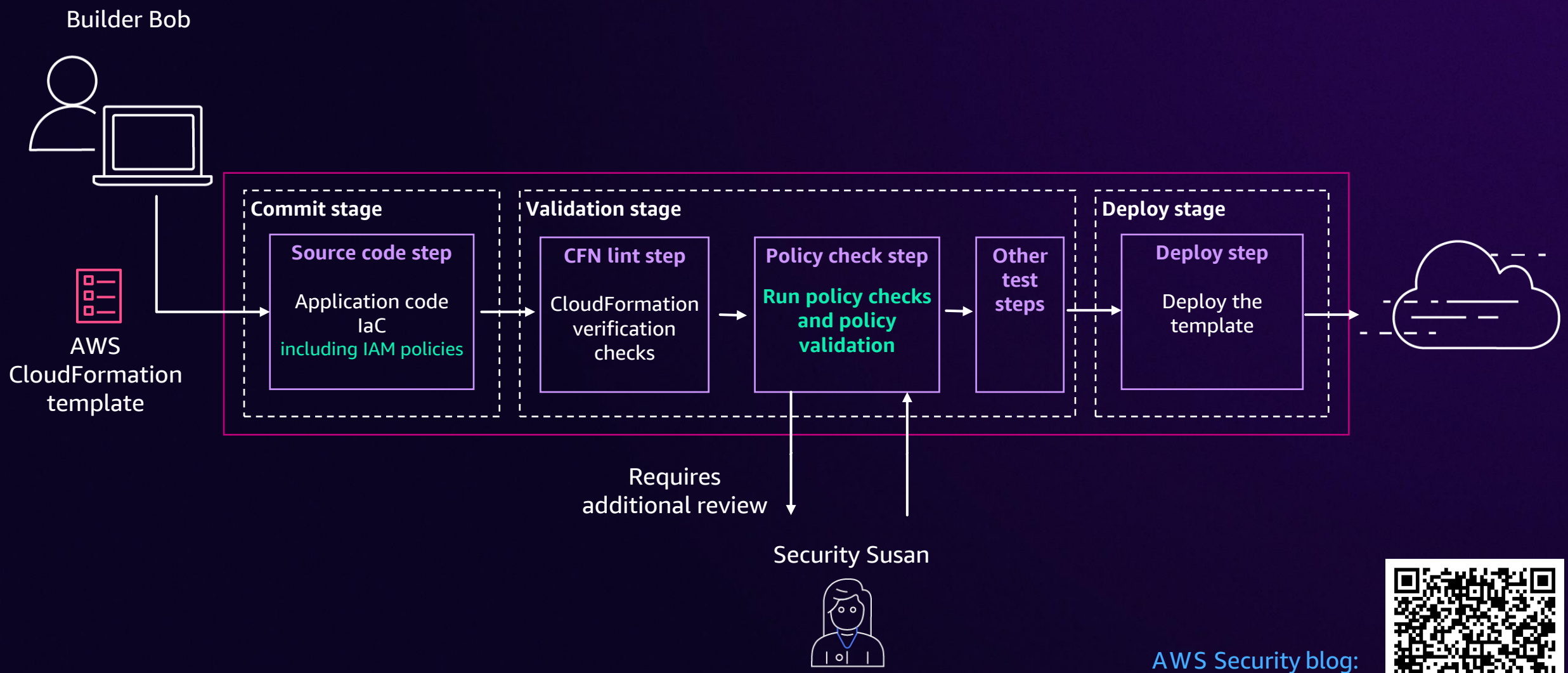
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": ”SetPermissionsBoundary",
            "Effect": "Allow",
            "Action": [
                "iam:CreateRole",
                "iam:AttachRolePolicy",
                "iam:DetachRolePolicy”
            ],
            "Resource": "*",
            "Condition": {
              "StringEquals": {
                "iam:PermissionsBoundary": "arn:aws:iam::123456789012:policy/CompanyBoundaries"
               }
            }
        },
        {
            "Sid": ”CreateAndEditPermissionsPolicy",
            "Effect": "Deny",
            "Action": [
                "iam:CreatePolicy",
                "iam:CreatePolicyVersion",
                "iam:DeletePolicyVersion"
             ],
            "Resource":  "arn:aws:iam::123456789012:policy/AppPolicies/*”

    }
  ]
}
```

Delegated approach

Builders codify the IAM roles/policies. Security team uses automated policy validation to perform the security checks that were previously manual

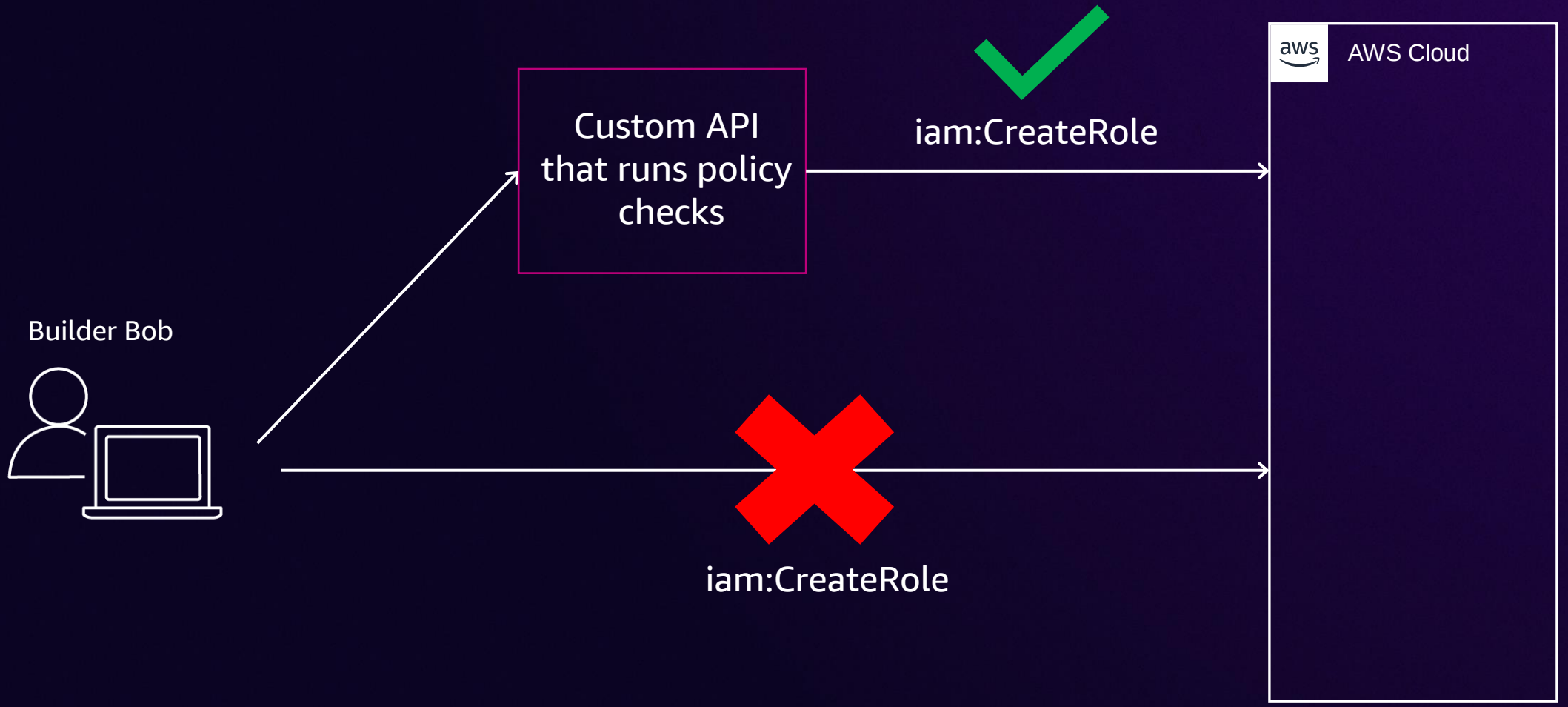


CI/CD pattern



AWS Security blog:
Validate IAM policies in CloudFormation templates using IAM Access Analyzer

API to create roles pattern



CI/CD vs API

Controls in CI/CD Pipeline

Pros:

- AWS offered tooling
- Get started quickly
- Backwards compatible with existing tooling

Cons:

- Difficult to make consistent at scale
- Difficult to ensure all pipelines are covered

API to create roles

Pros:

- Limit scope for CreateRole* permission (API)
- Centralized evaluation
- Flexibility on responding to failed policy checks/validations via API response

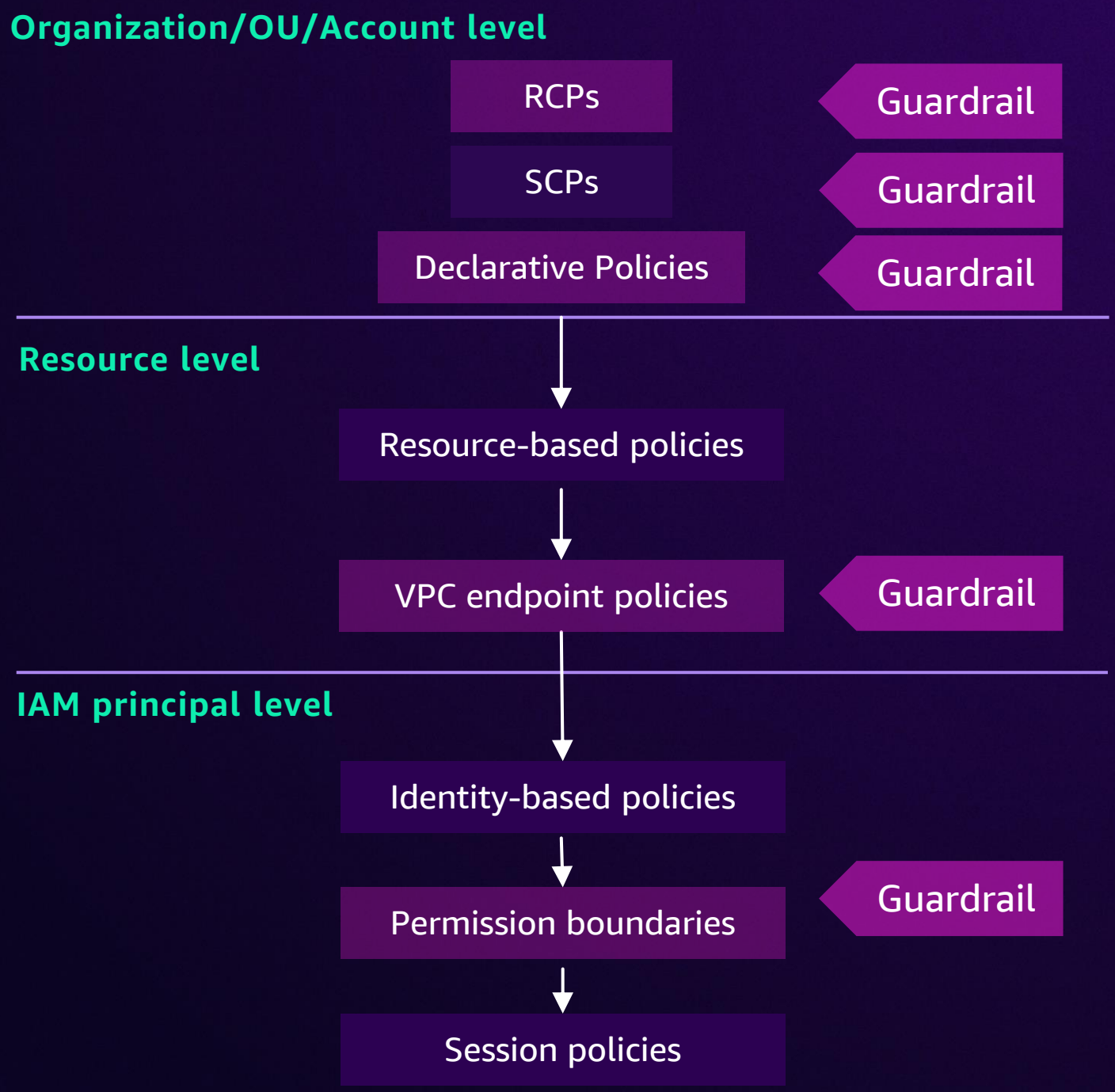
Cons:

- Custom built

While maintaining security controls

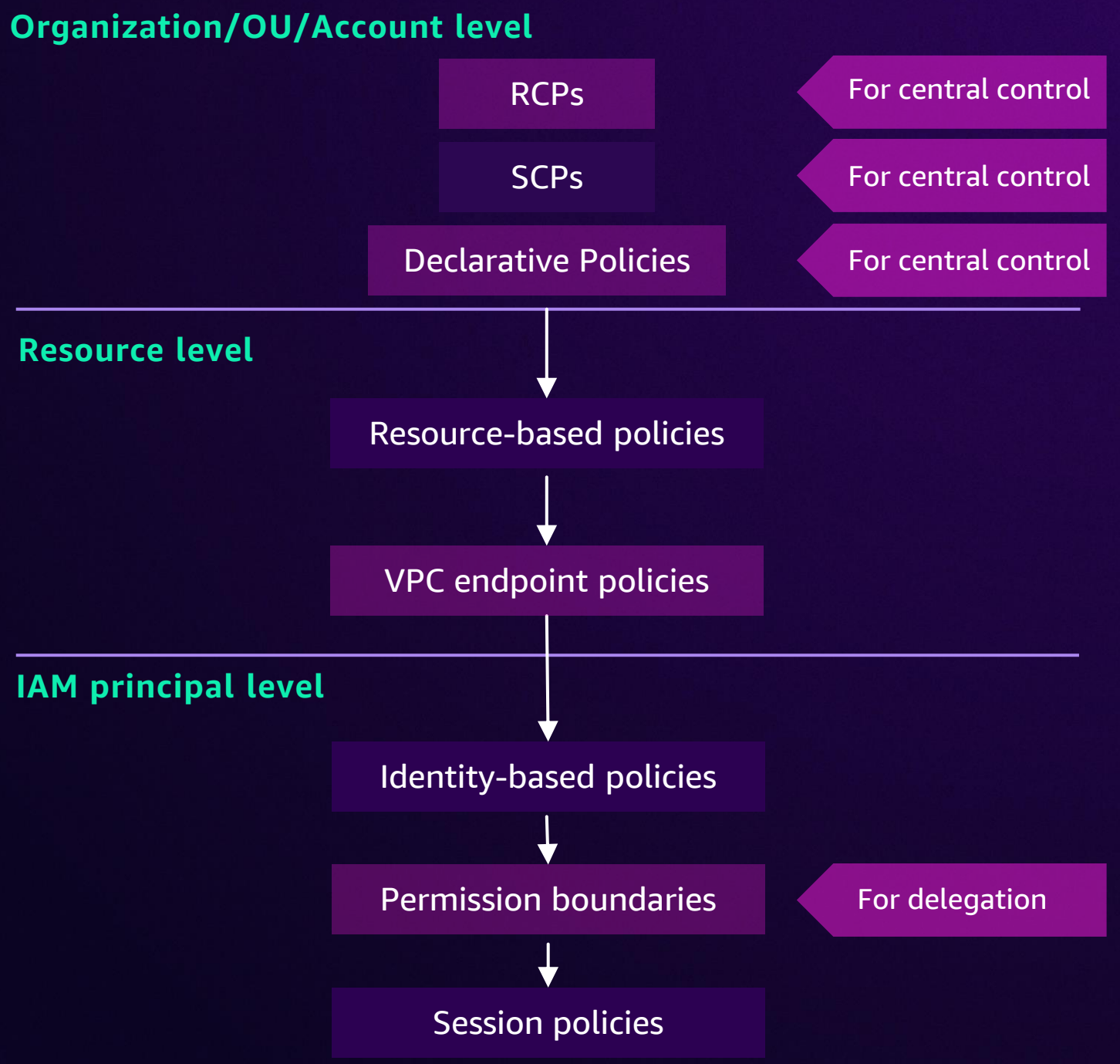
AWS IAM policy types

Impact radius
Intent
Guardrail vs. Grant



AWS IAM policy types

- Impact radius
- Intent
- Guardrail vs. Grant
- Should be used by Central team



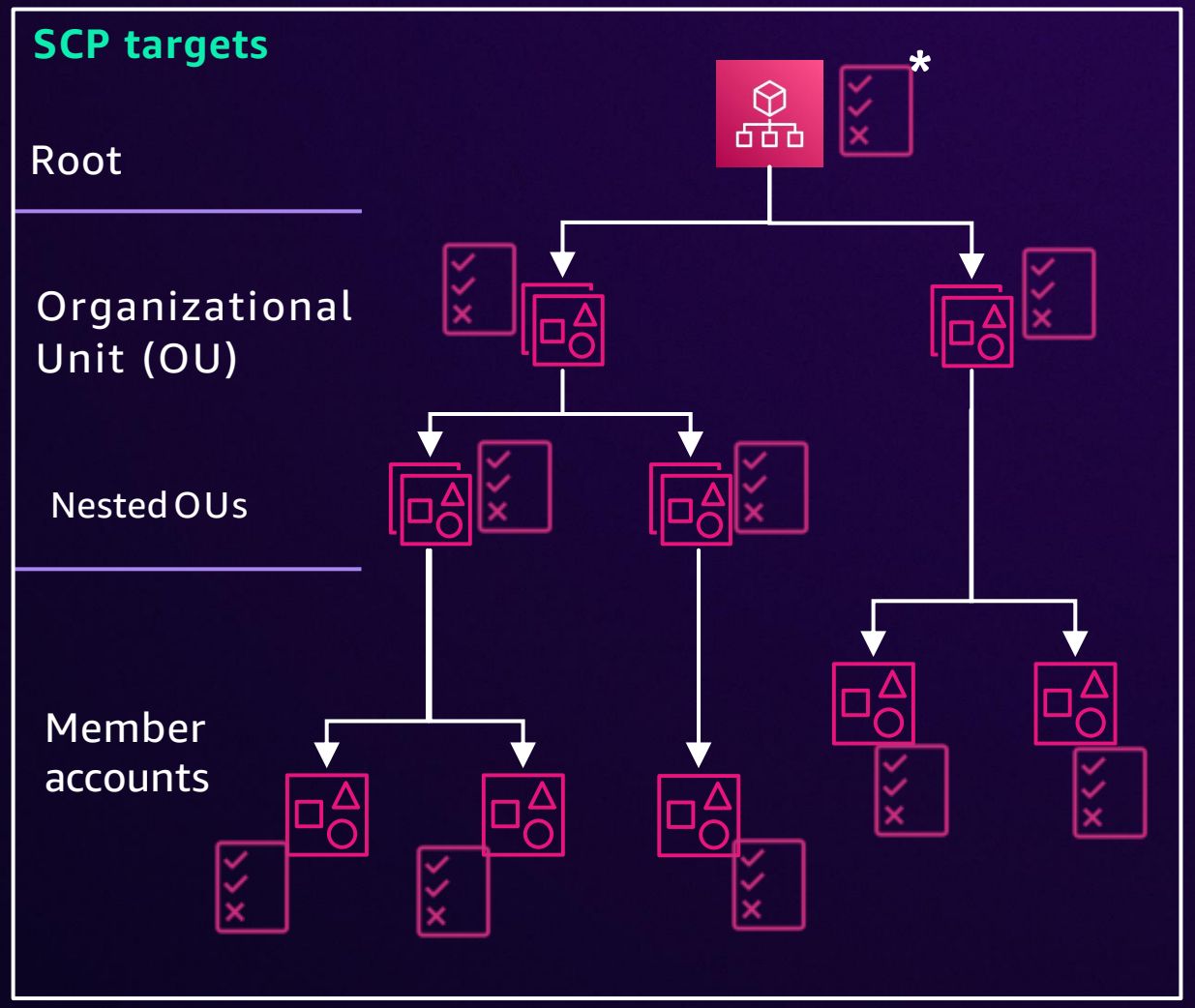
Service control policies (SCPs)

CENTRALLY CREATE AND APPLY PREVENTIVE GUARDRAILS FOR YOUR ORGANIZATION

Define maximum available permissions of the IAM principals for all accounts in your organization

Can be applied to organization root, OUs, or member accounts

Supports delegated administration



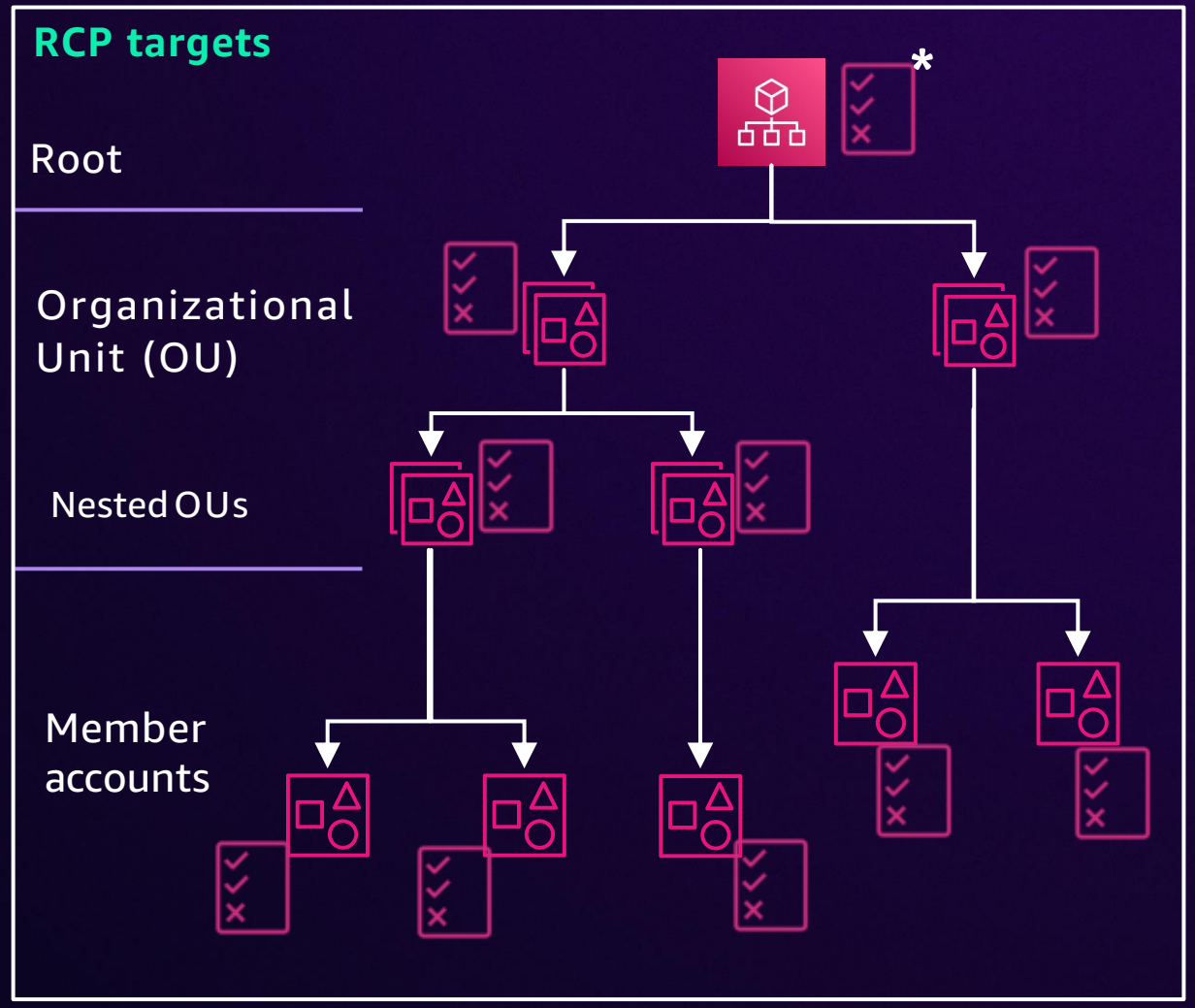
Resource control policies (RCPs)

CENTRALLY CREATE AND APPLY PREVENTIVE GUARDRAILS FOR YOUR ORGANIZATION

Define uniform access controls on resources across your AWS environment

Can be applied to organization root, OUs, or member accounts

Supports delegated administration



RCPs are only supported with few services today

NEW

Declarative policies for EC2

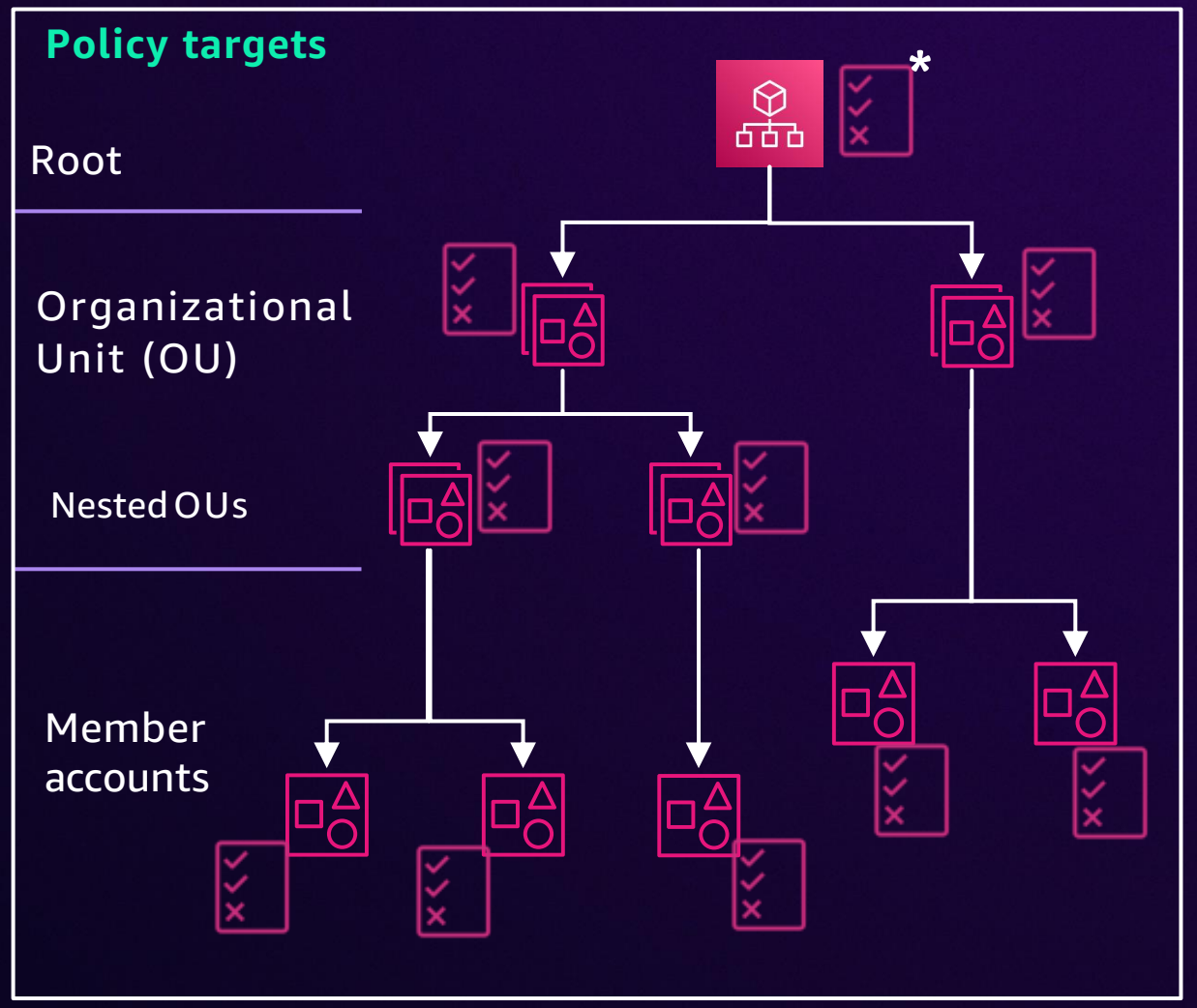
DECLARE AND ENFORCE DESIRED CONFIGURATION FOR A GIVEN AWS SERVICE

Once attached, the configuration is always maintained when EC2 adds new features or APIs

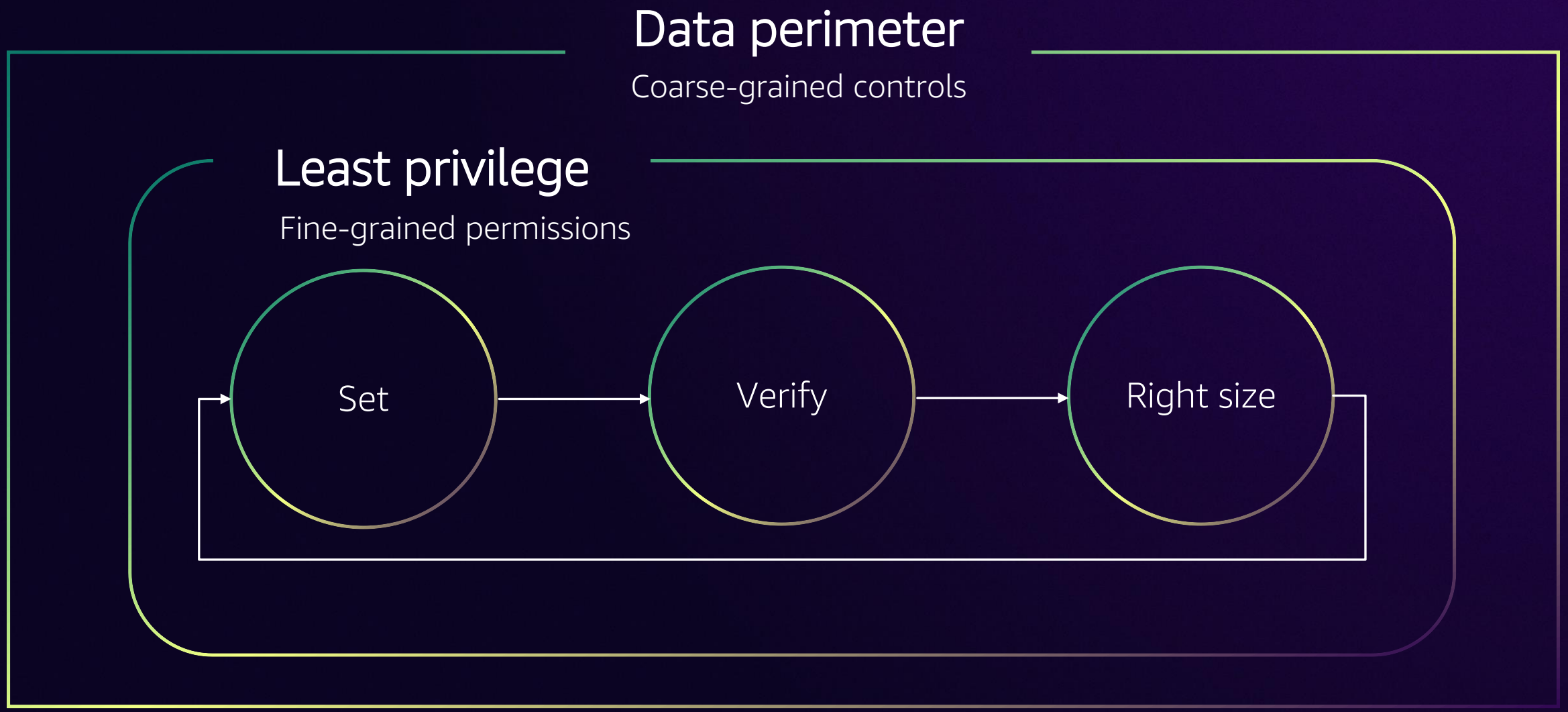
Can be applied to organization root, OUs, or member accounts

Examples:

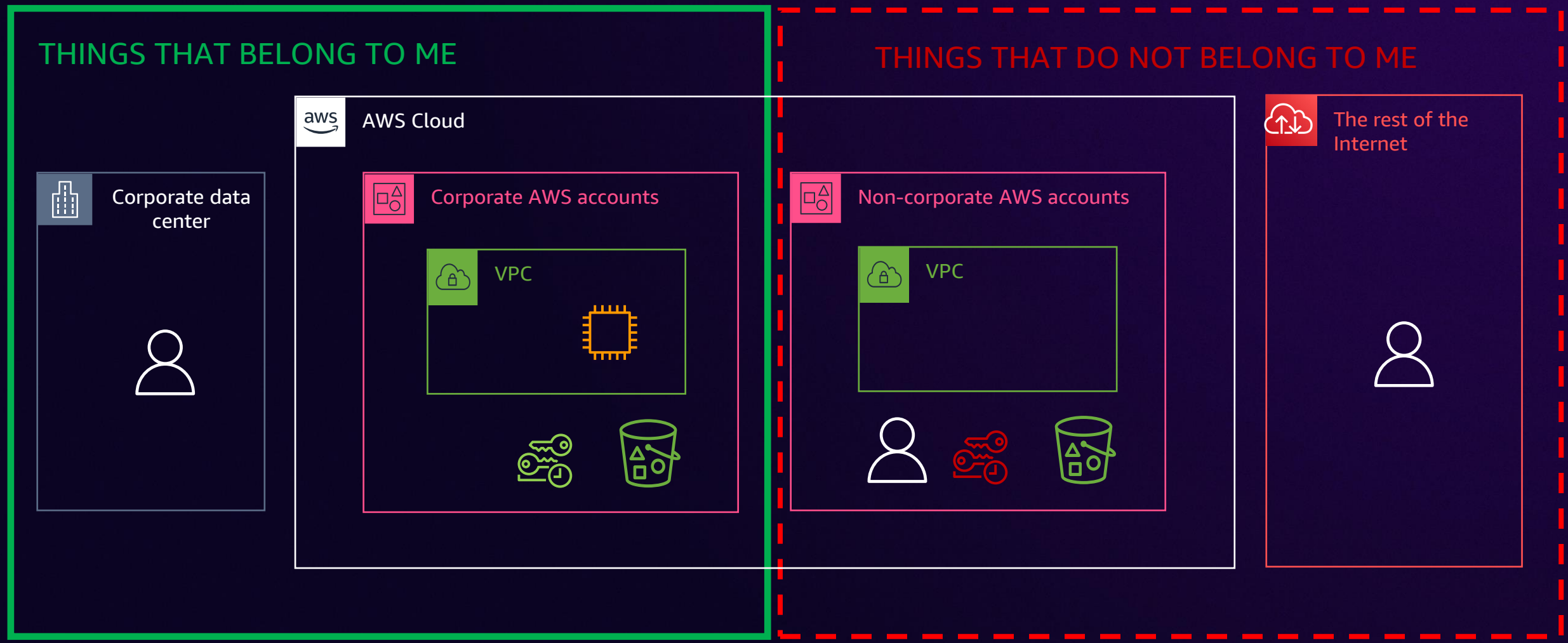
- EC2 image block public access
- Snapshot block public access
- VPC block public access



Access management strategy



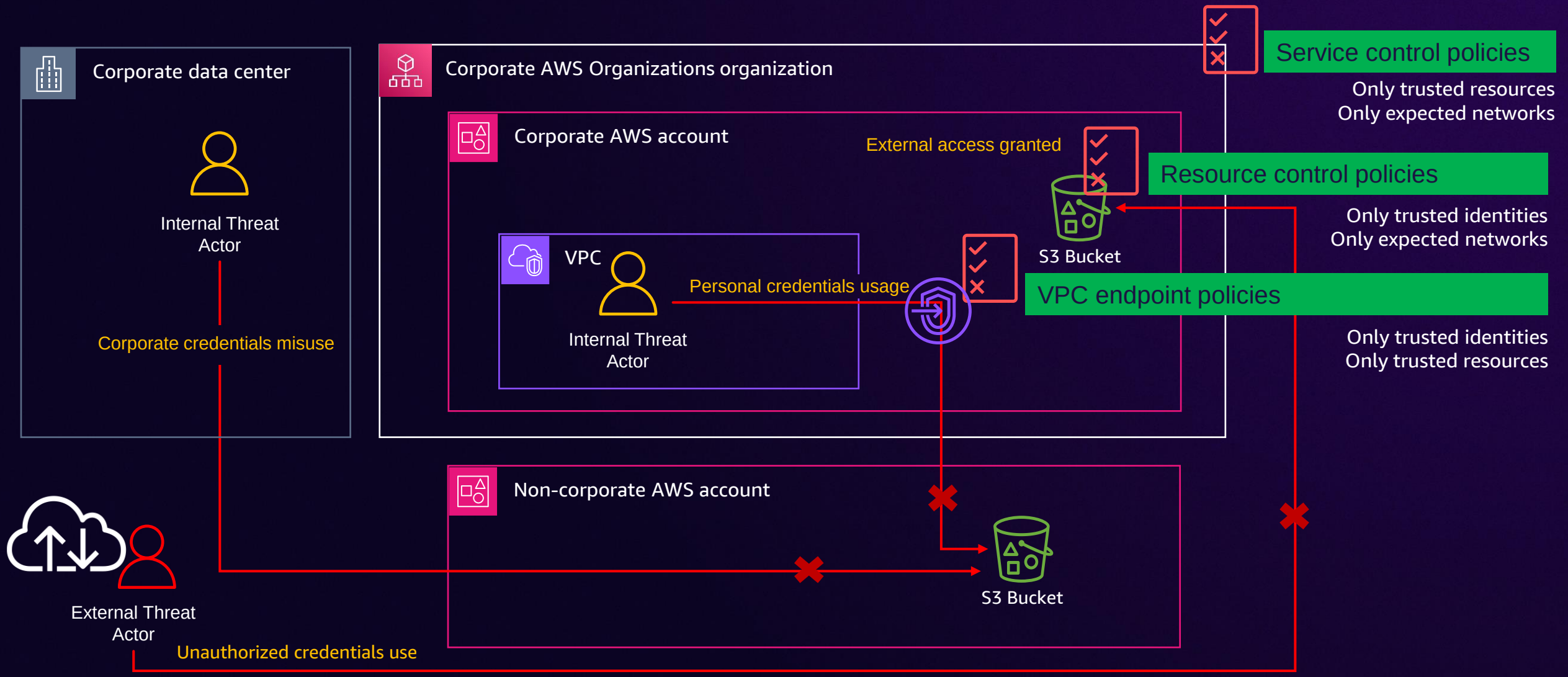
What is a data perimeter?



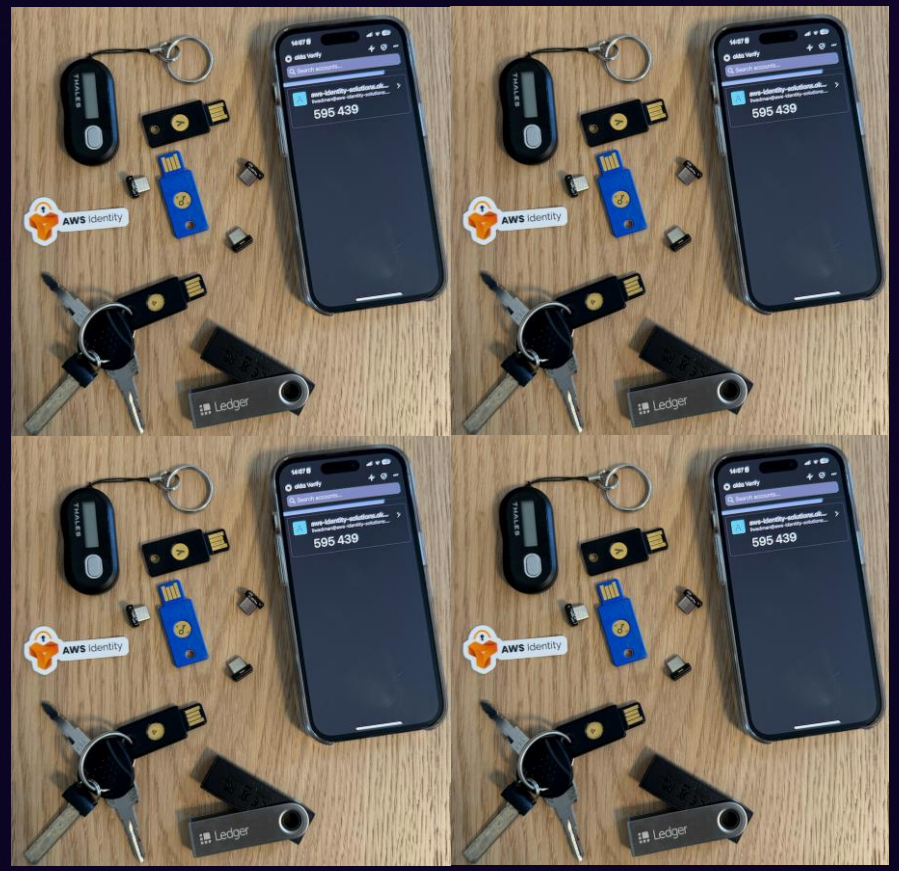
Data perimeter controls

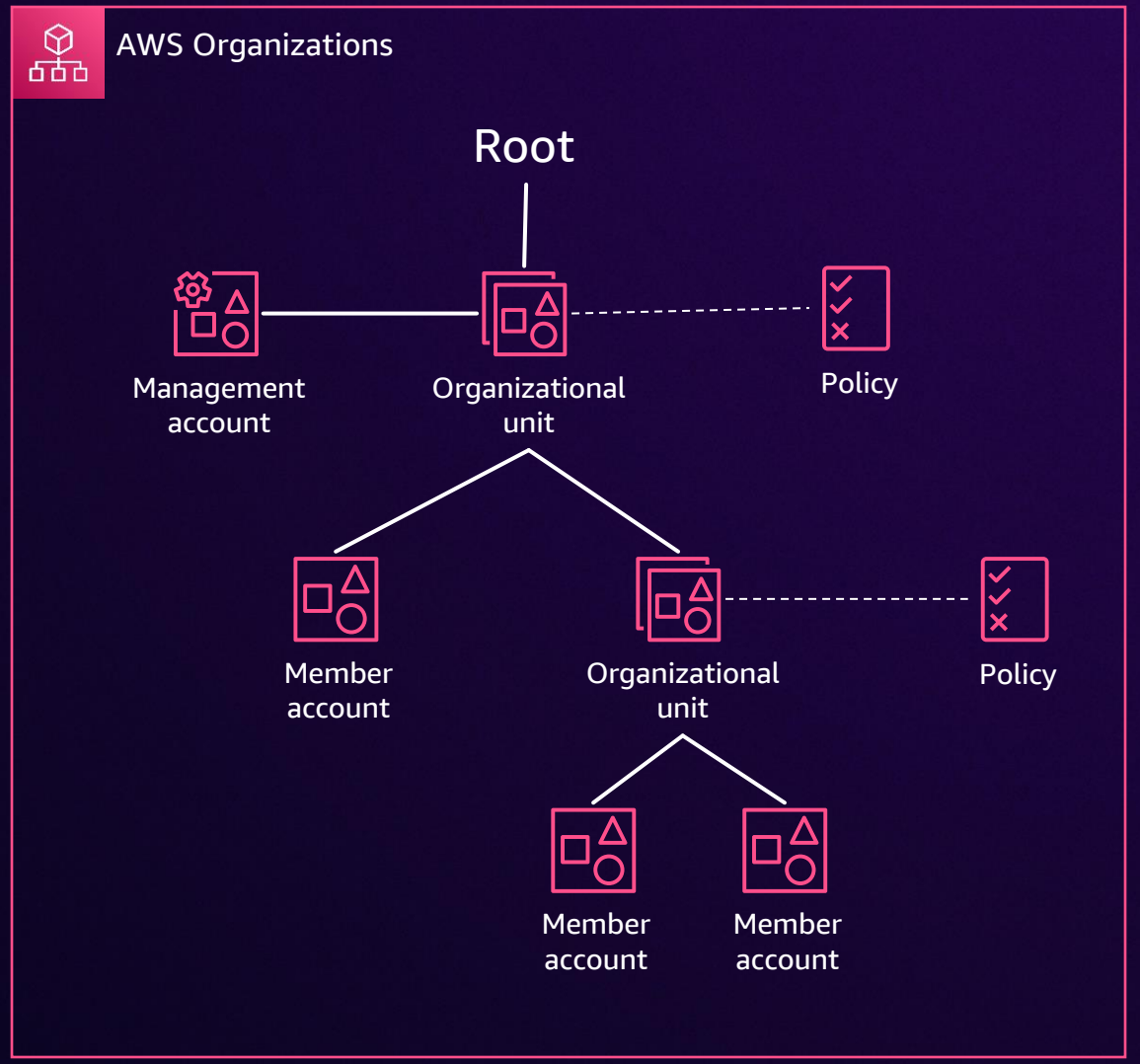
Perimeter	Intent / Control Objective	Applied on	Using	Primary IAM feature
Identity	Only trusted identities can access my resources	**Resources**	RCP	aws:PrincipalOrgID aws:PrincipalIsAWSService
	Only trusted identities are allowed from my network	**Network**	VPC endpoint policy	aws:PrincipalOrgID aws:PrincipalIsAWSService
Resource	My identities can access only trusted resources	**Identities**	SCP	aws:ResourceOrgID
	Only trusted resources can be accessed from my network	**Network**	VPC endpoint policy	aws:ResourceOrgID
Network	My identities can access resources only from expected networks	**Identities**	SCP	aws:SourceIp aws:SourceVpc/aws:SourceVpce aws:ViaAWSService
	My resources can only be accessed from expected networks	**Resources**	RCP	aws:SourceIp aws:SourceVpc/aws:SourceVpce aws:ViaAWSService aws:PrincipalIsAWSService

Mitigating security risks with data perimeter controls



Managing root users in AWS Organizations



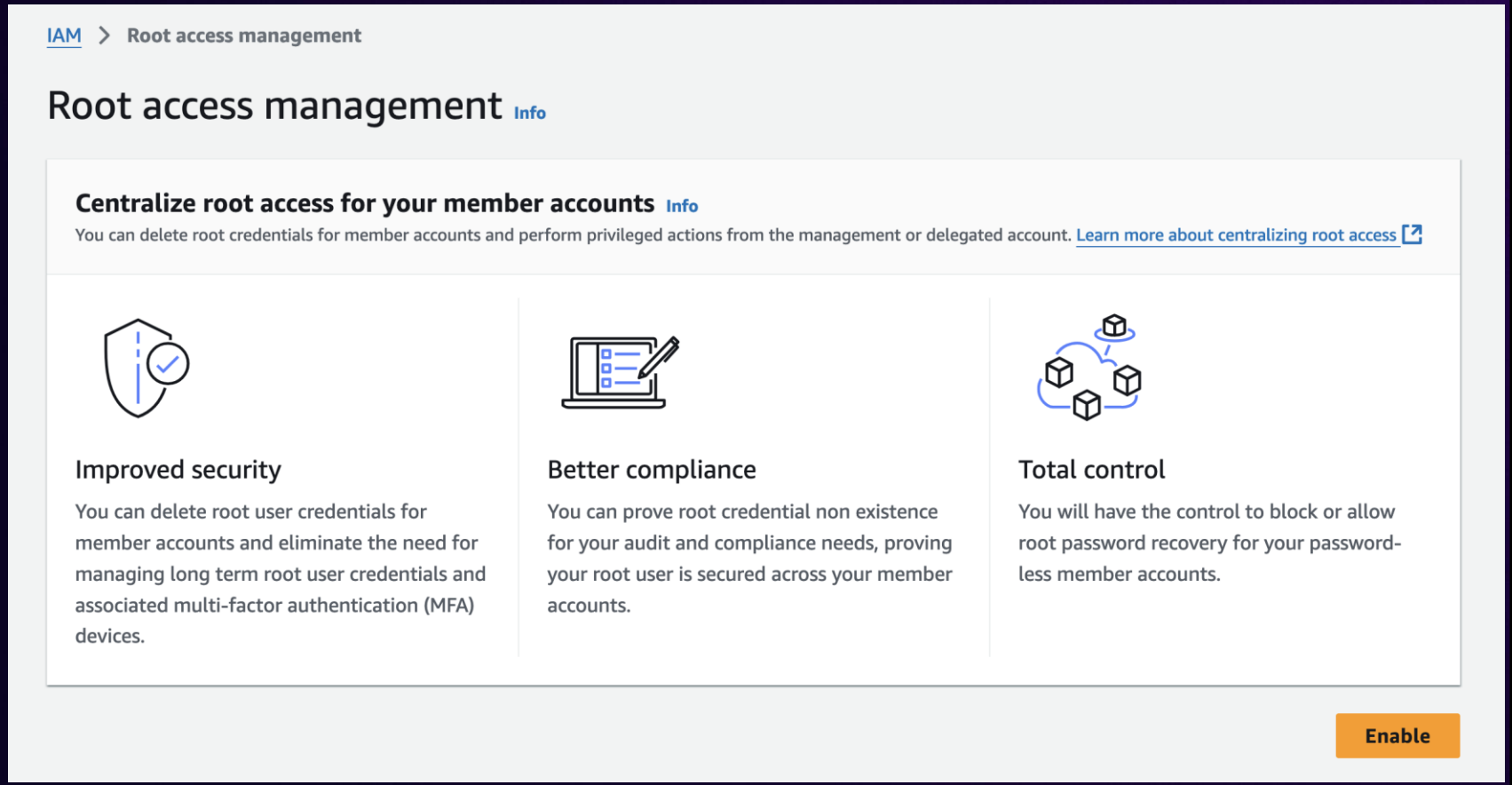


NEW

Central management for root access

PROVIDES CUSTOMERS WITH A SIMPLIFIED WAY TO MANAGE, SECURE, AND CONTROL HIGHLY PRIVILEGED ACCESS TO THEIR AWS MEMBER ACCOUNTS

- Simplify audit and compliance
- Easily manage root credentials
- Tightly scoped privileged access



Access Analysis with IAM Access Analyzer

EXTERNAL AND UNUSED ACCESS ANALYSIS

The unused access analyzing tool can be **enabled** in your account or with Organizations

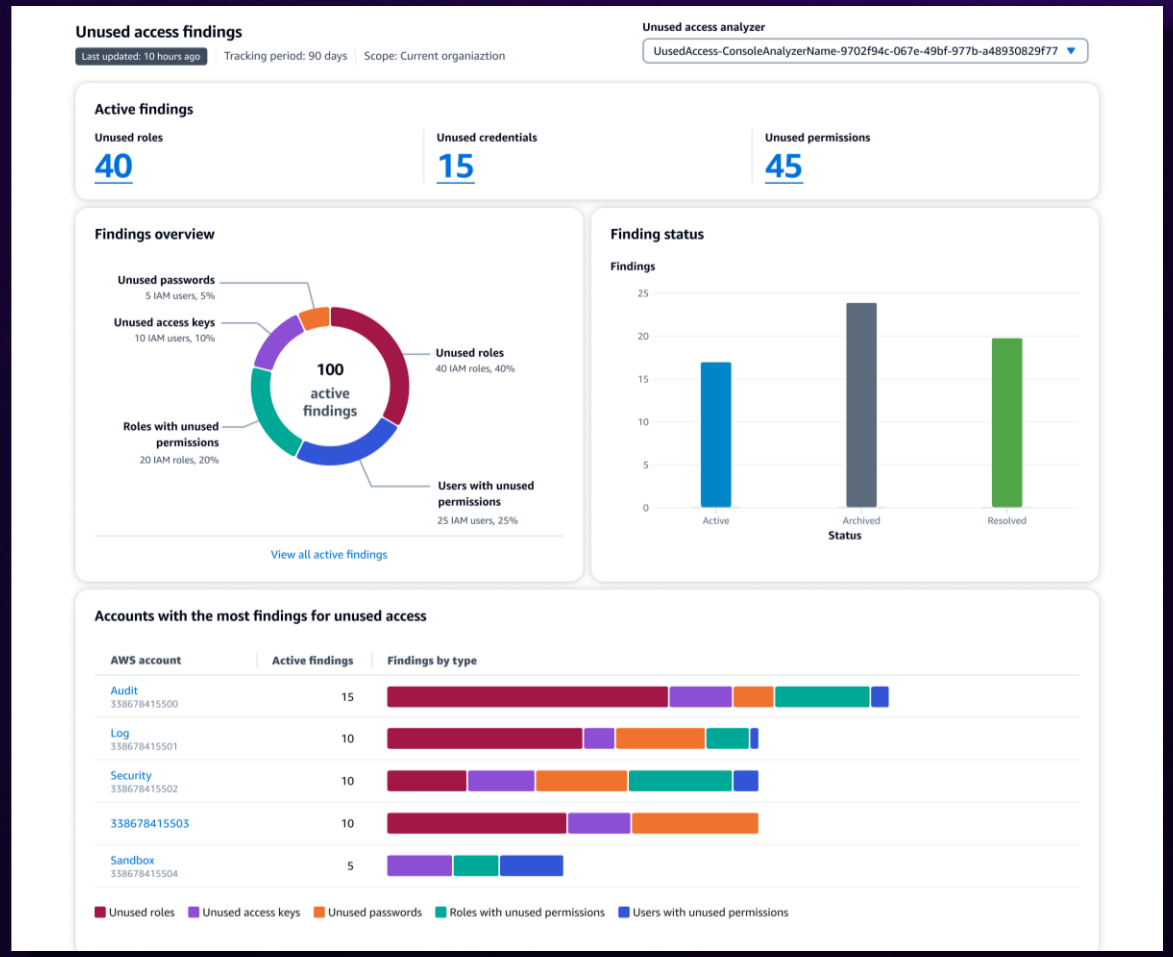
Continually **monitors and reviews** access activity across all IAM roles and users

Identifies unused **roles**, unused user **credentials**, and unused **permissions**

Generates findings for you to review over-permissive IAM roles and users

Review dashboard to identify accounts that need attention

Customers use action last accessed by individual roles for no additional charge; optionally, customers use this **paid offering** for continual monitoring and centralized review capabilities



NEW

Unused access analysis – Analyzer scoping

Specific accounts – using account ID

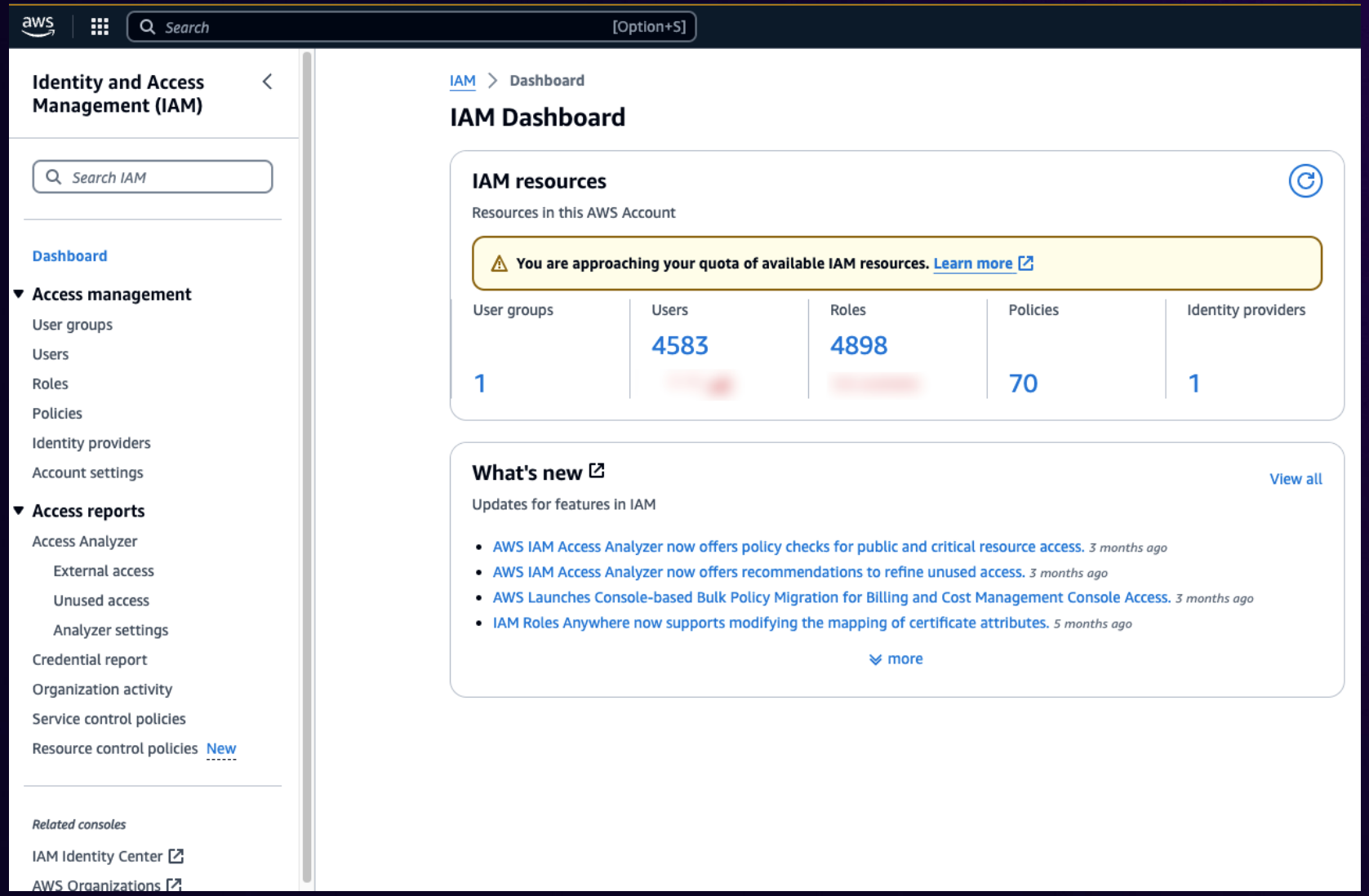
Specific IAM principals – using principal tags

How does it help?

- ❑ Focus unused access analysis on important AWS accounts and IAM principals
- ❑ “Prioritized” unused access findings based on specific accounts and principals
- ❑ Optimize the cost for unused access analysis

Assessing your IAM posture

Assess your AWS IAM posture

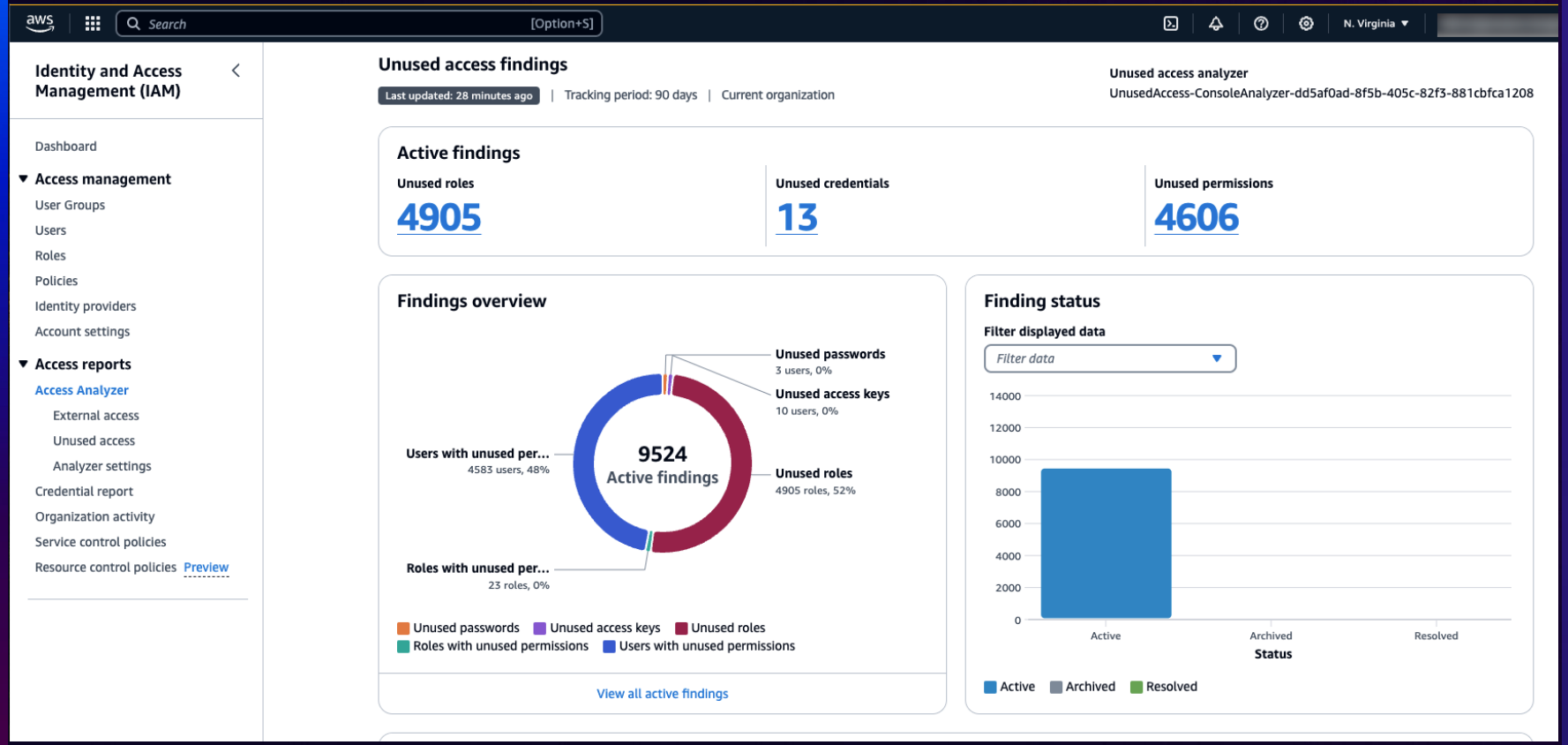


Not good!

A lot of IAM users

Long-lived credentials such as access keys

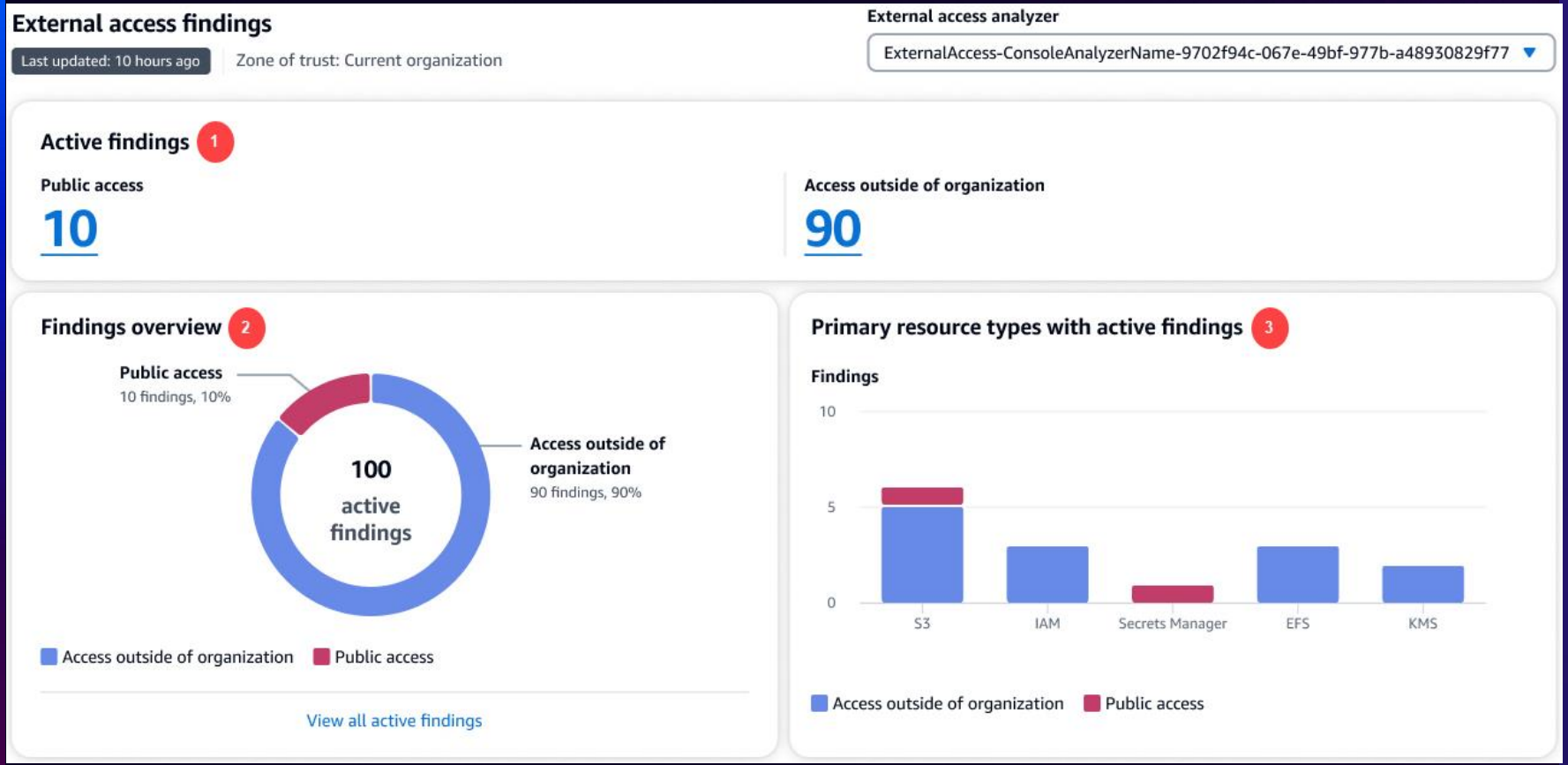
Assess your AWS IAM posture



Not good!

Overly-permissive permissions

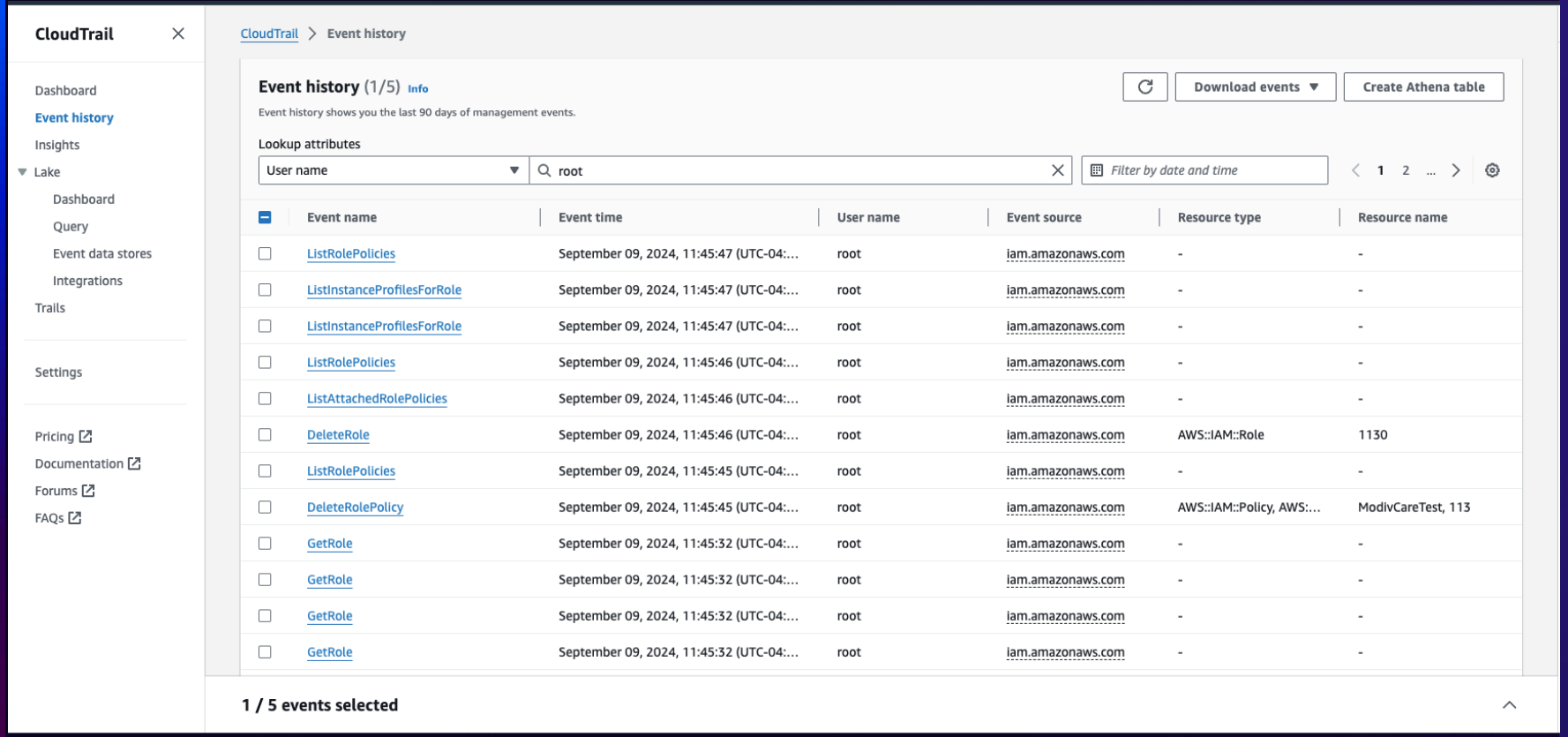
Assess your AWS IAM posture



Not good!

External access on resources

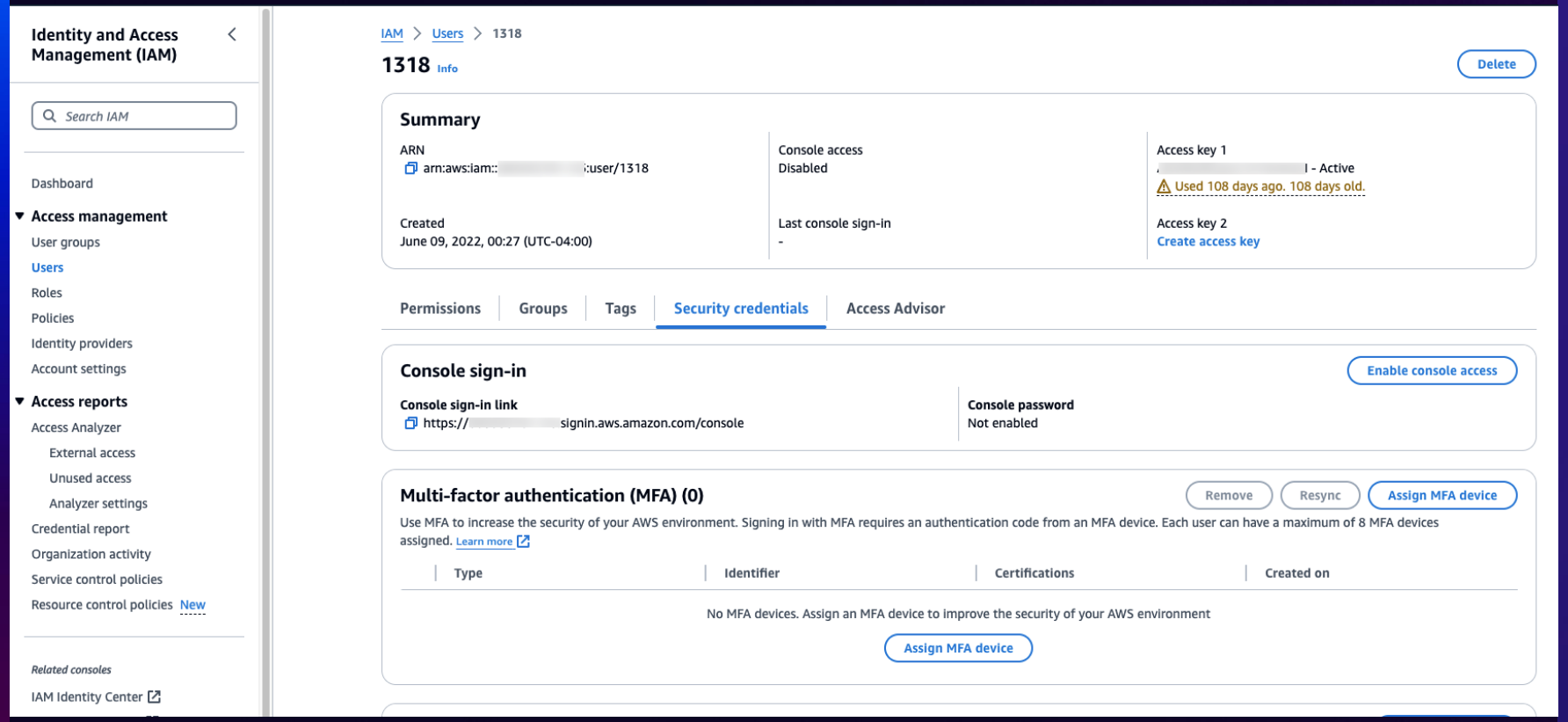
Assess your AWS IAM posture



Not good!

Root user usage

Assess your AWS IAM posture

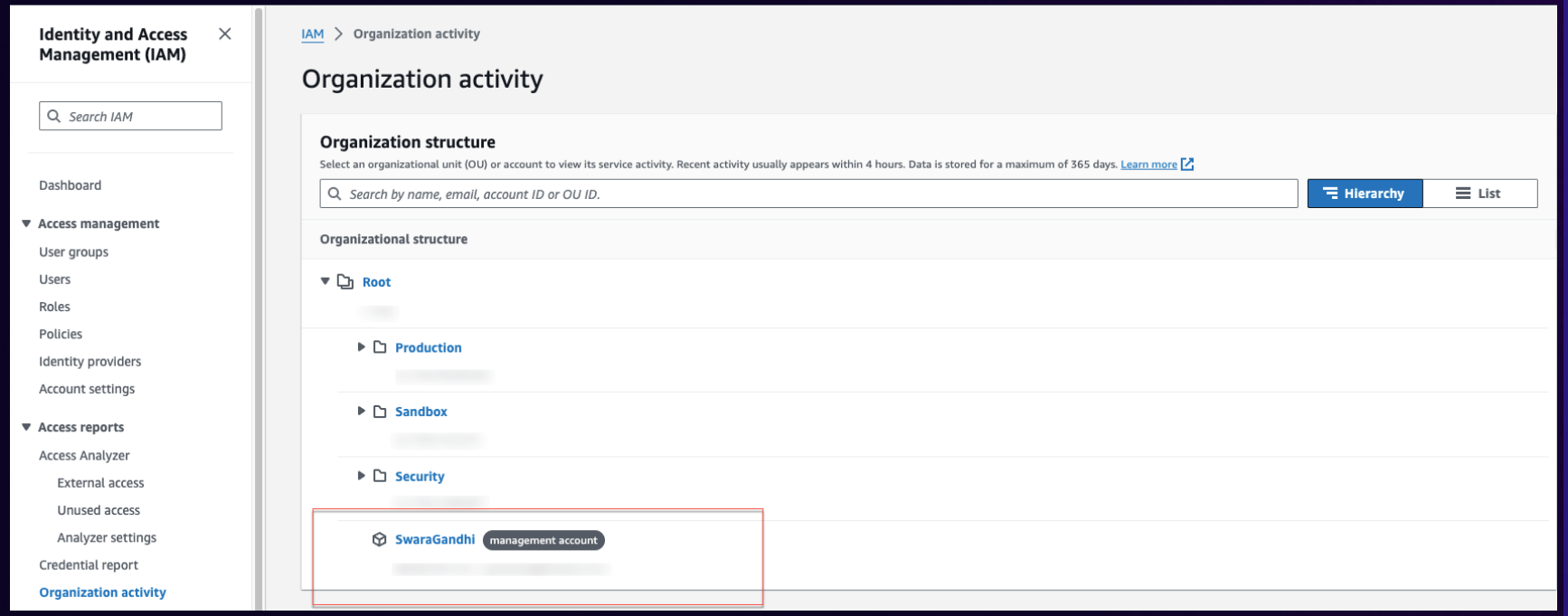


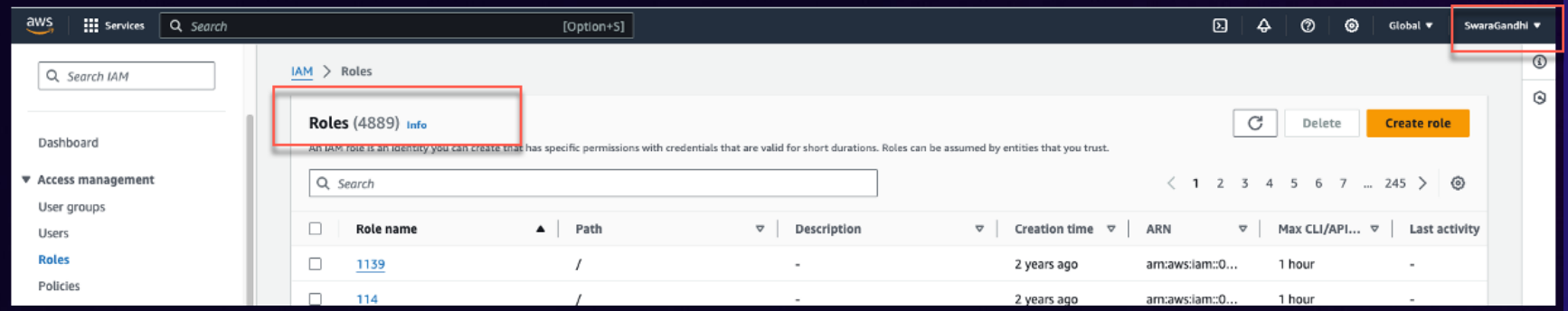
Not good!

Access keys are not rotated

No MFA

Assess your AWS IAM posture

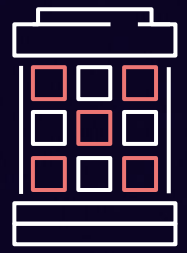




Not good!

Multiple IAM resources in AWS Management account

Elements of scaling



Visibility

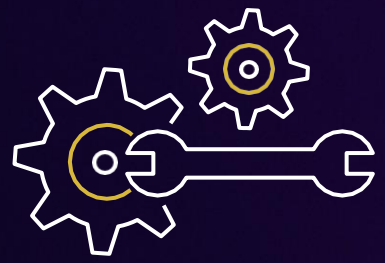
IAM Access Analyzer

AWS Security Hub

Scaling

Automation

Delegate



Centralization

AWS IAM Identity center

Policy deployment pipeline

Security

Data perimeter controls

Proactive controls embedded in CI/CD

Defined Processes

Ownership

Feedback Loop

Just in time access

Resources

Validate IAM policies in CloudFormation templates using IAM Access Analyzer

AWS CloudFormation policy validator

AWS IAM Access Analyzer custom policy check samples

Data Perimeters landing page

Thank you!

Please complete the session survey in the mobile app

Swara Gandhi

linkedin.com/in/swaragandhi

ganswara@amazon.com

Jeremy Ware

linkedin.com/in/jeremyware104/

jmware@amazon.com