SEC316-R

Access control confidence: Grant the right access to the right things

Brigid Johnson
Senior Manager, AWS Identity
Amazon Web Services

re:Invent
aws

As your organization builds on AWS, granting developers and applications the right access to the right resources at the right time for the right actions is critical to security. In this session, we share an approach to setting permissions in AWS environments. We demonstrate configuring permission guardrails and delegating permission administration to development teams. We show how to set fine-grained permissions that scale with your organization using attribute-based access control (ABAC). Finally, we discuss how to confidently dial in permissions. We guide you through each step and provide examples, helping you gain the confidence to set access controls in your organization.

Access control confidence

. . . it's a journey

Where you start

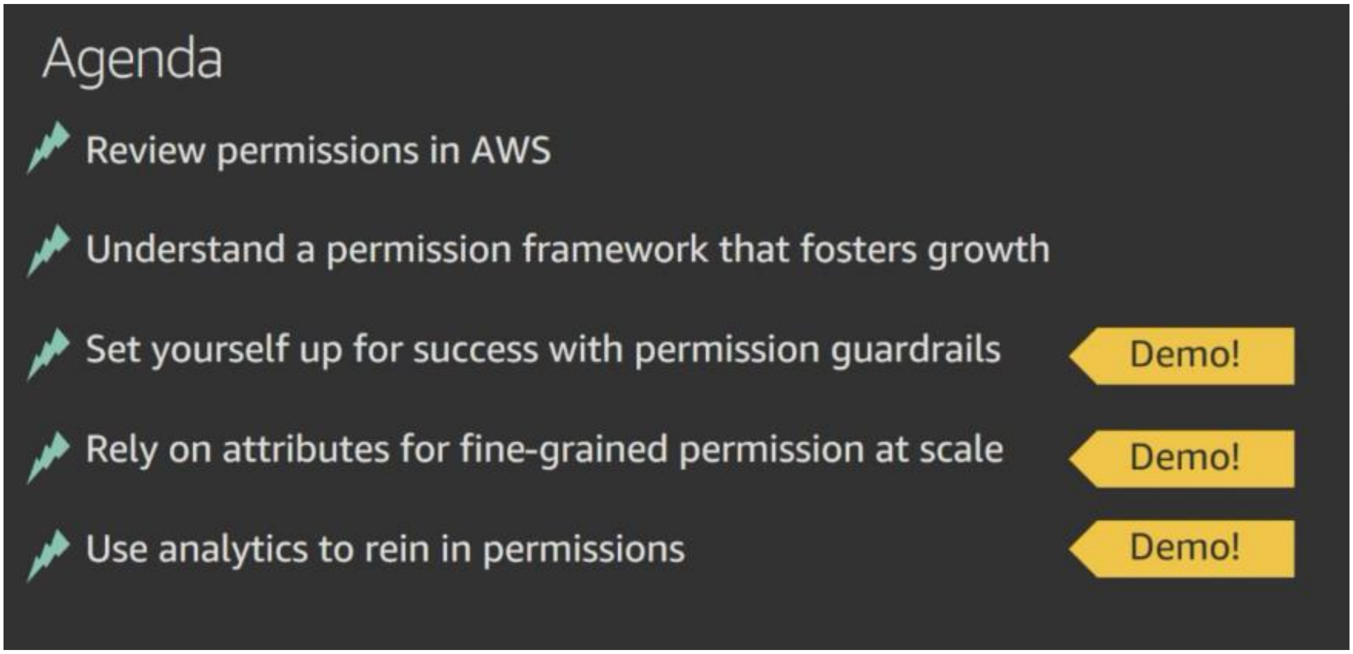
- Business to innovate
- Agility to move fast
- Give builders freedom

Where you are going

- Prevent dangerous actions
- Accountable security posture
- Least privilege

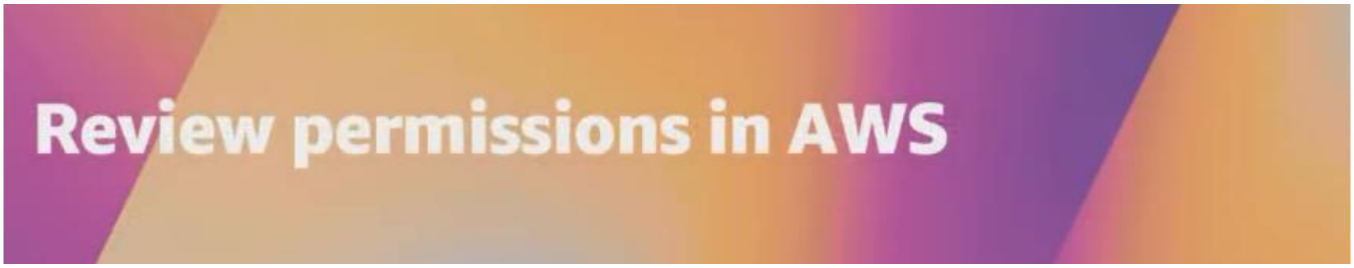
Let's go!

We are on a journey to least privilege access to resources and having the right security posture for our organizations

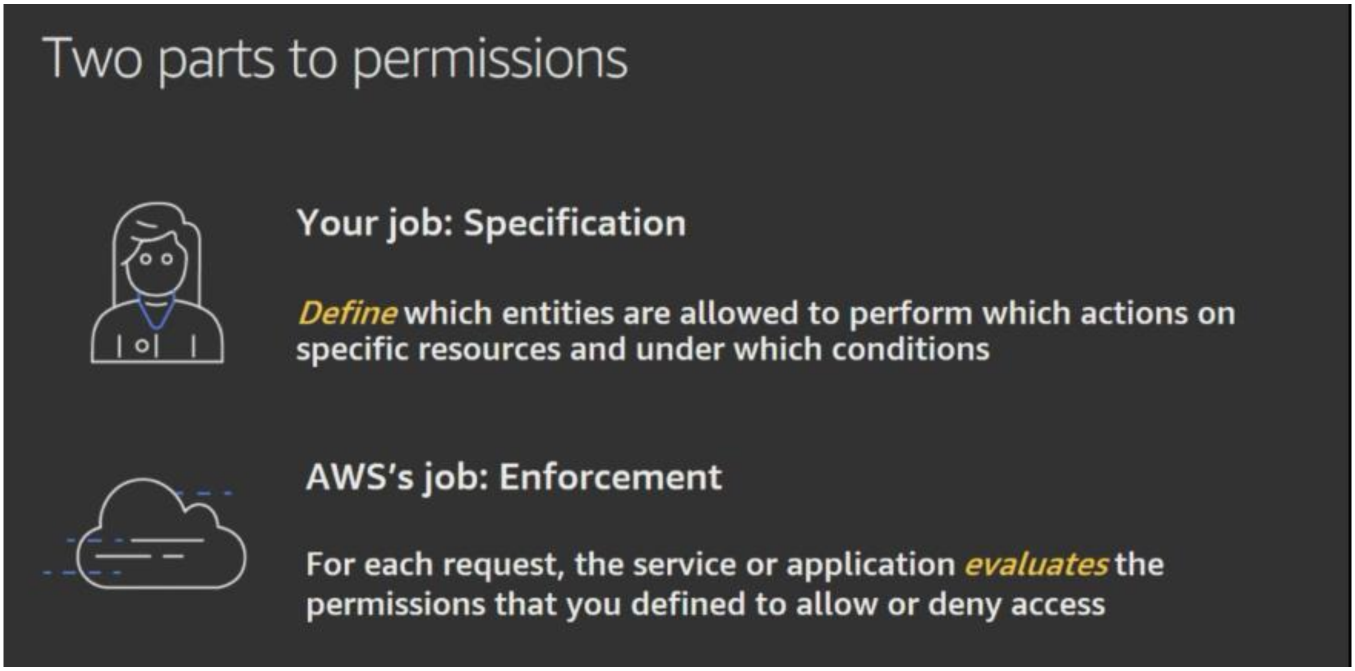


Agenda

- Review permissions in AWS
- Understand a permission framework that fosters growth
- Set yourself up for success with permission guardrails — Demo!
- Rely on attributes for fine-grained permission at scale — Demo!
- Use analytics to rein in permissions — Demo!



Review permissions in AWS



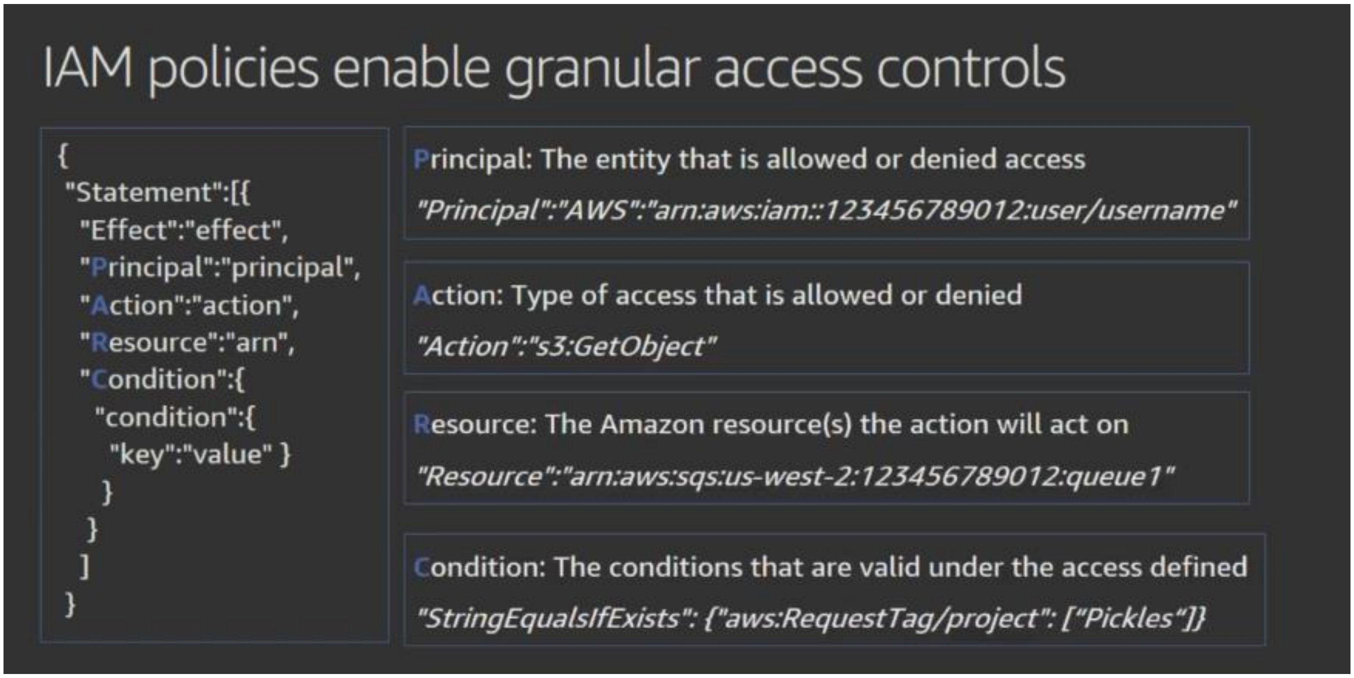
Two parts to permissions

Your job: Specification

Define which entities are allowed to perform which actions on specific resources and under which conditions

AWS's job: Enforcement

For each request, the service or application *evaluates* the permissions that you defined to allow or deny access



IAM policies enable granular access controls

```
{
 "Statement":[{
  "Effect":"effect",
  "Principal":"principal",
  "Action":"action",
  "Resource":"arn",
  "Condition":{
   "condition":{
    "key":"value" }
   }
  }
 ]
}
```

Principal: The entity that is allowed or denied access

"Principal":"AWS":"arn:aws:iam::123456789012:user/username"

Action: Type of access that is allowed or denied

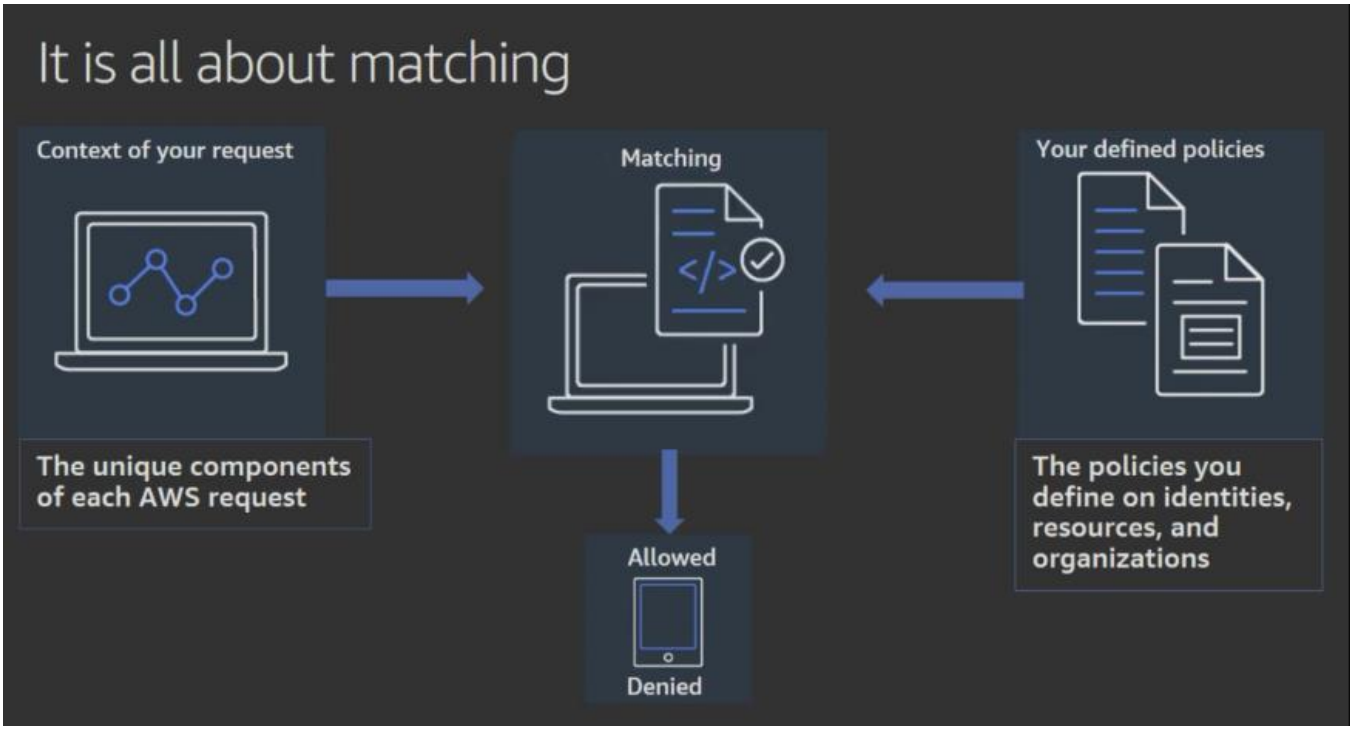
"Action":"s3:GetObject"

Resource: The Amazon resource(s) the action will act on

"Resource":"arn:aws:sqs:us-west-2:123456789012:queue1"

Condition: The conditions that are valid under the access defined

"StringEqualsIfExists": {"aws:RequestTag/project": [“Pickles“]}



It is all about matching

Context of your request

The unique components of each AWS request

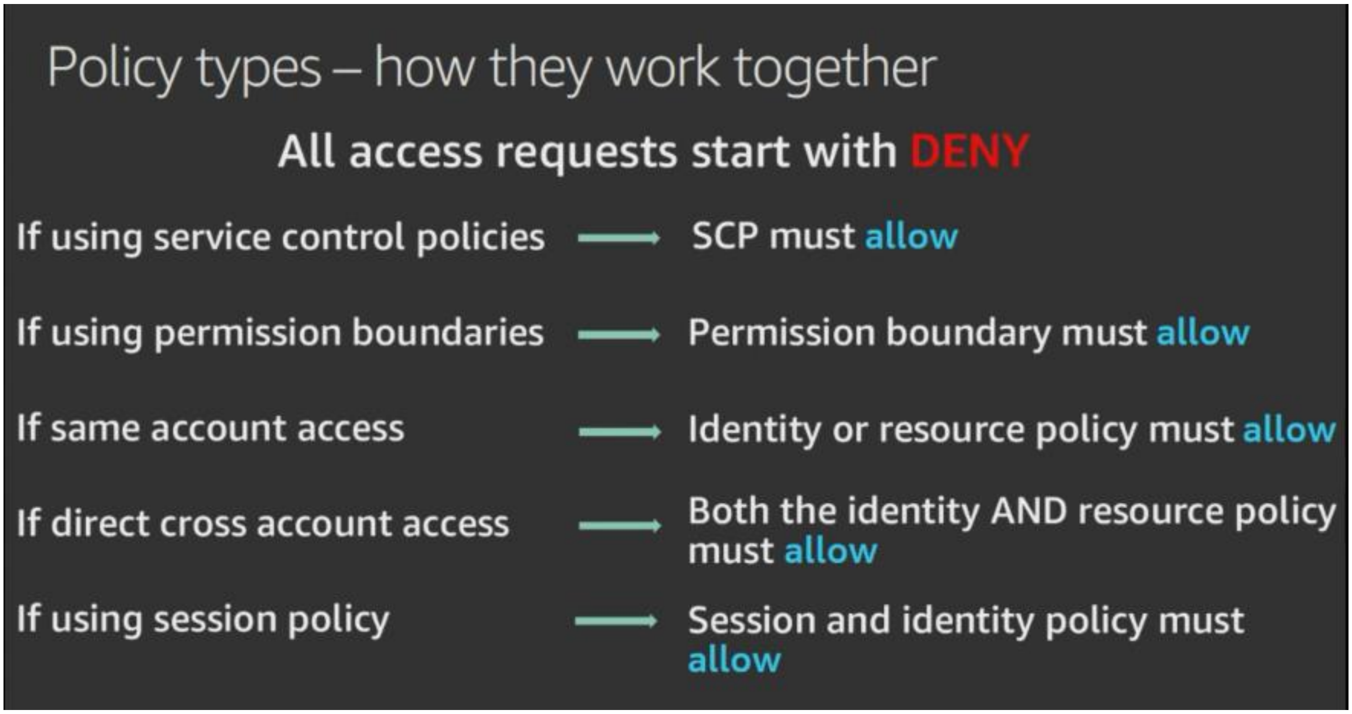
Matching

Allowed

Denied

Your defined policies

The policies you define on identities, resources, and organizations

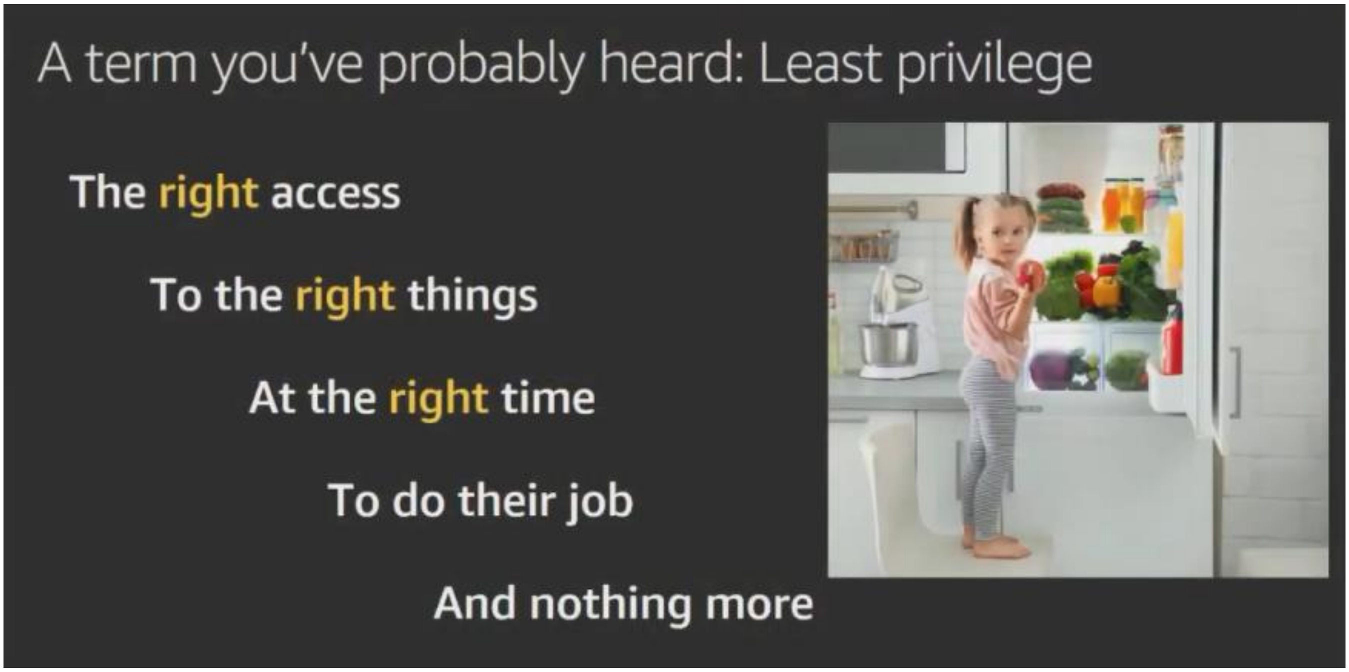


Policy types – how they work together

All access requests start with DENY

If using service control policies →	SCP must allow
If using permission boundaries →	Permission boundary must allow
If same account access →	Identity or resource policy must allow
If direct cross account access →	Both the identity AND resource policy must allow
If using session policy →	Session and identity policy must allow

Understand a permission framework that fosters growth



A term you've probably heard: Least privilege

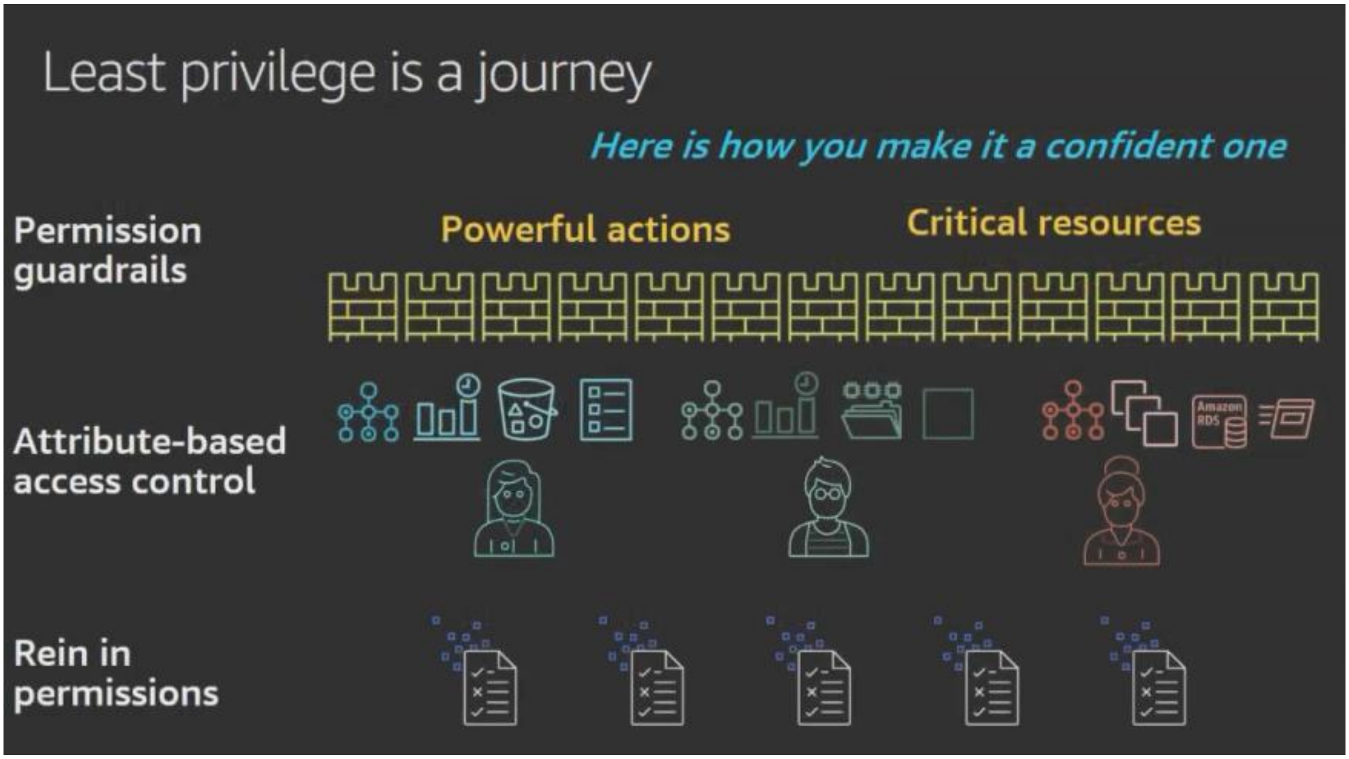
The right access

To the right things

At the right time

To do their job

And nothing more



Least privilege is a journey

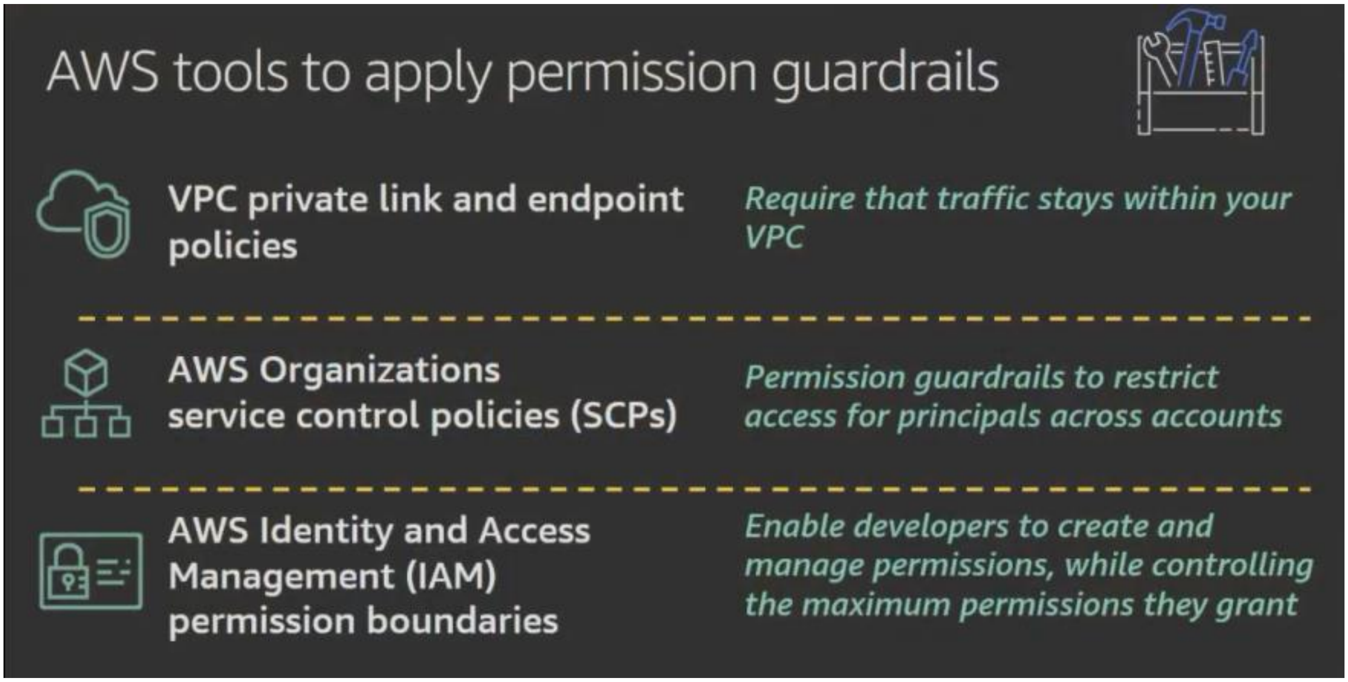
Here is how you make it a confident one

Permission guardrails — Powerful actions · Critical resources

Attribute-based access control

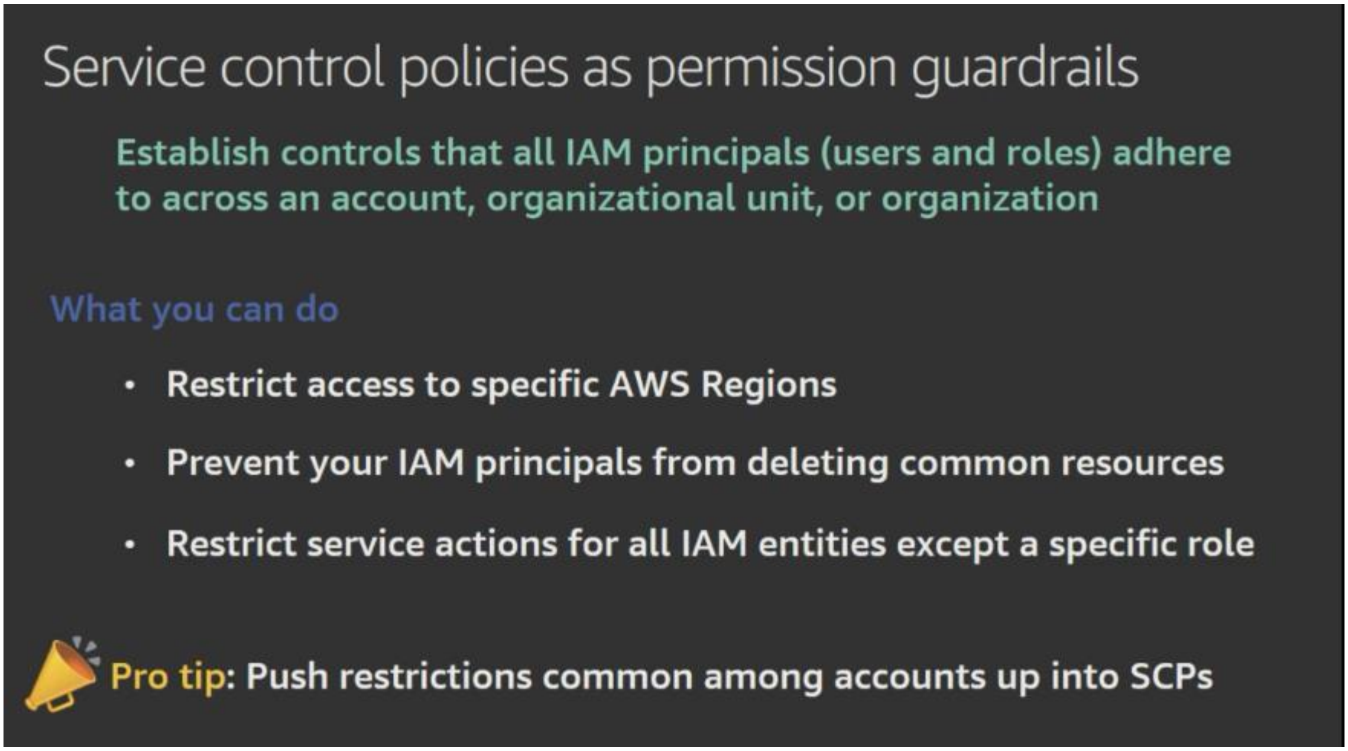
Rein in permissions

Set yourself up for success with permission guardrails



AWS tools to apply permission guardrails

VPC private link and endpoint policies	*Require that traffic stays within your VPC*
AWS Organizations service control policies (SCPs)	*Permission guardrails to restrict access for principals across accounts*
AWS Identity and Access Management (IAM) permission boundaries	*Enable developers to create and manage permissions, while controlling the maximum permissions they grant*



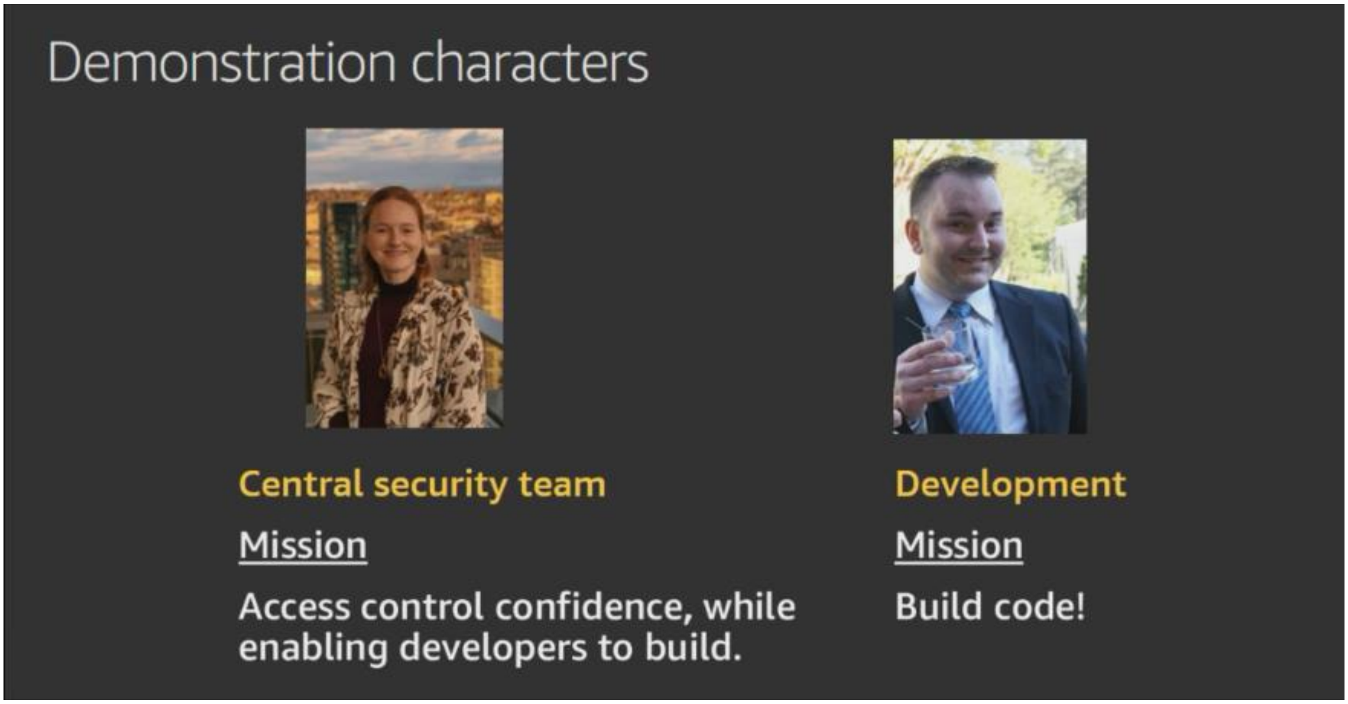
Service control policies as permission guardrails

Establish controls that all IAM principals (users and roles) adhere to across an account, organizational unit, or organization

What you can do

- Restrict access to specific AWS Regions
- Prevent your IAM principals from deleting common resources
- Restrict service actions for all IAM entities except a specific role

Pro tip: Push restrictions common among accounts up into SCPs



Demonstration characters

Central security team

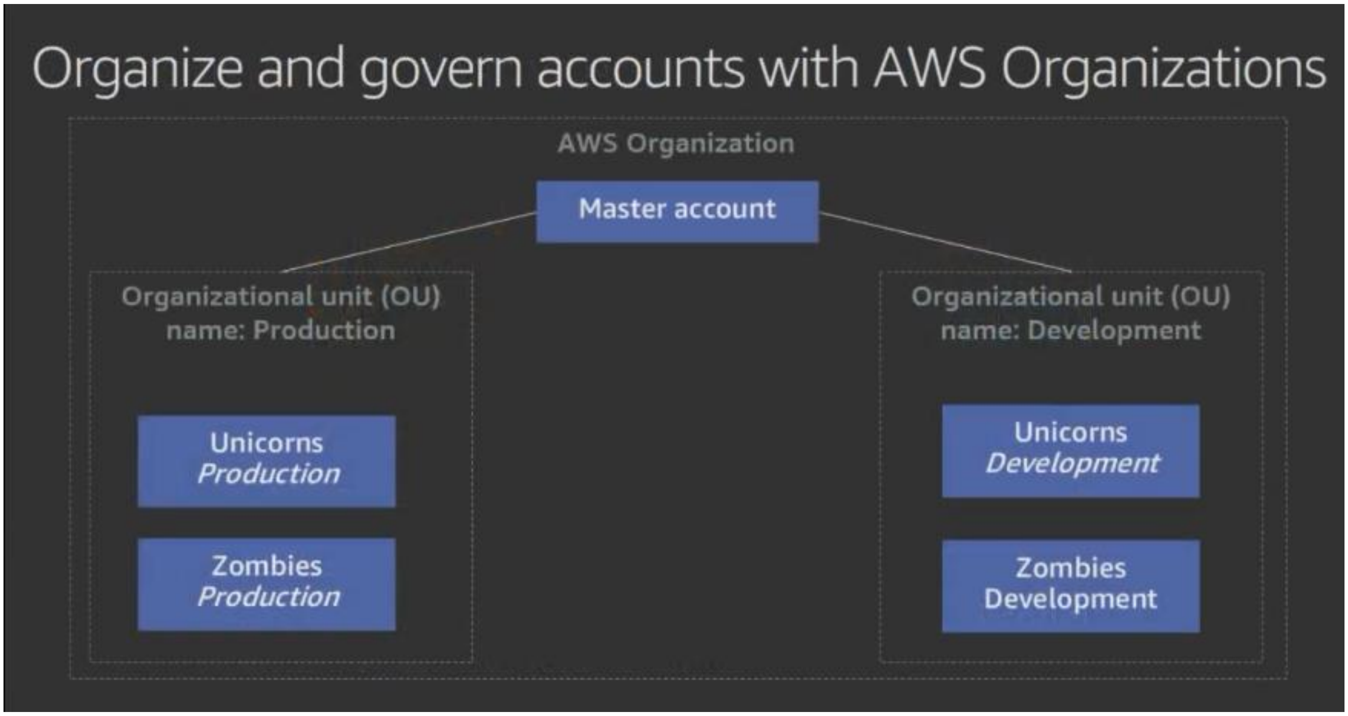
Mission

Access control confidence, while enabling developers to build.

Development

Mission

Build code!



Organize and govern accounts with AWS Organizations

AWS Organization

Master account

Organizational unit (OU) name: Production

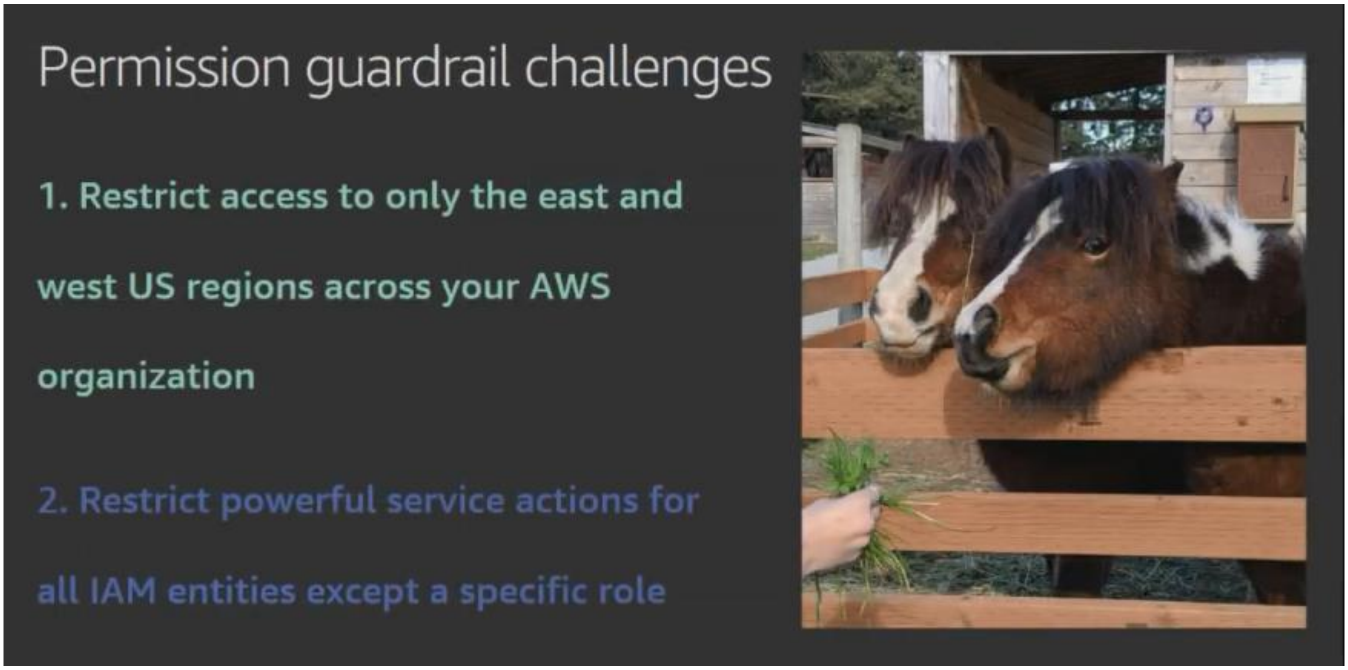
Unicorns *Production*

Zombies *Production*

Organizational unit (OU) name: Development

Unicorns *Development*

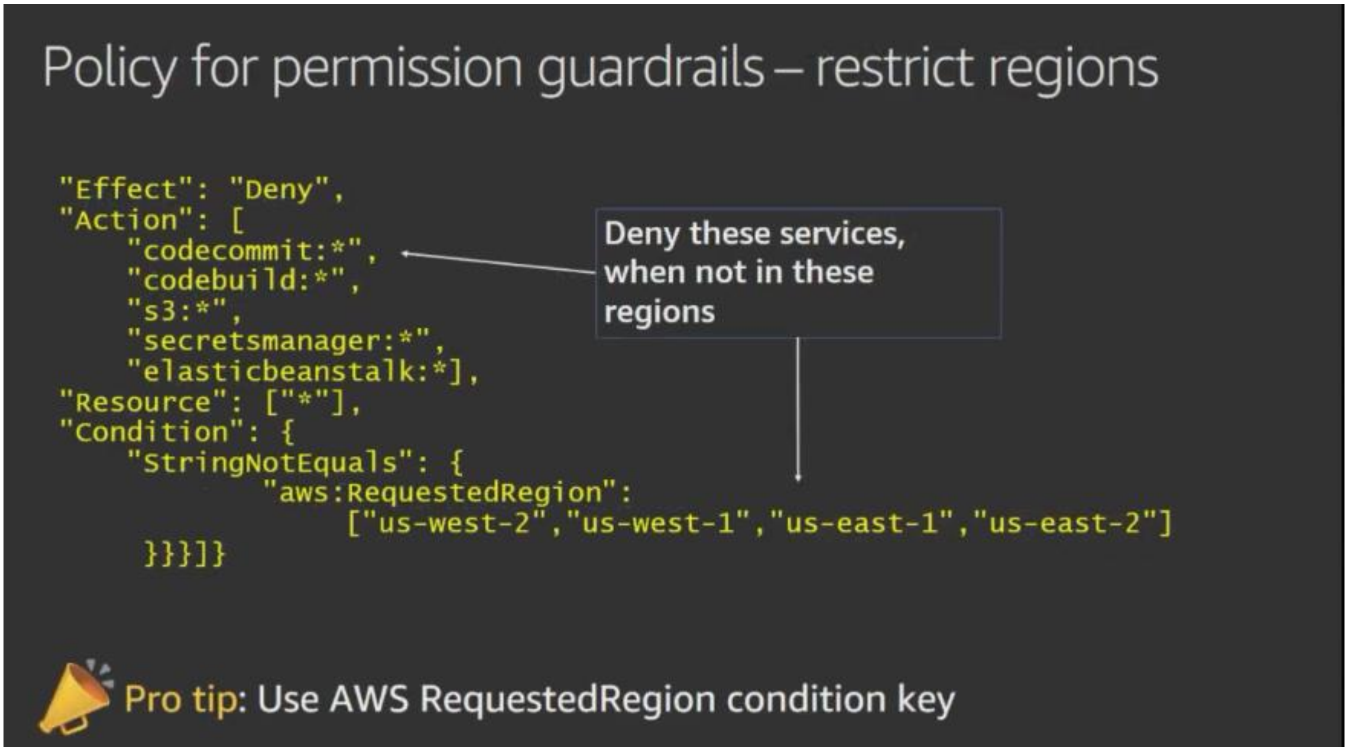
Zombies Development



Permission guardrail challenges

1. Restrict access to only the east and west US regions across your AWS organization

2. Restrict powerful service actions for all IAM entities except a specific role

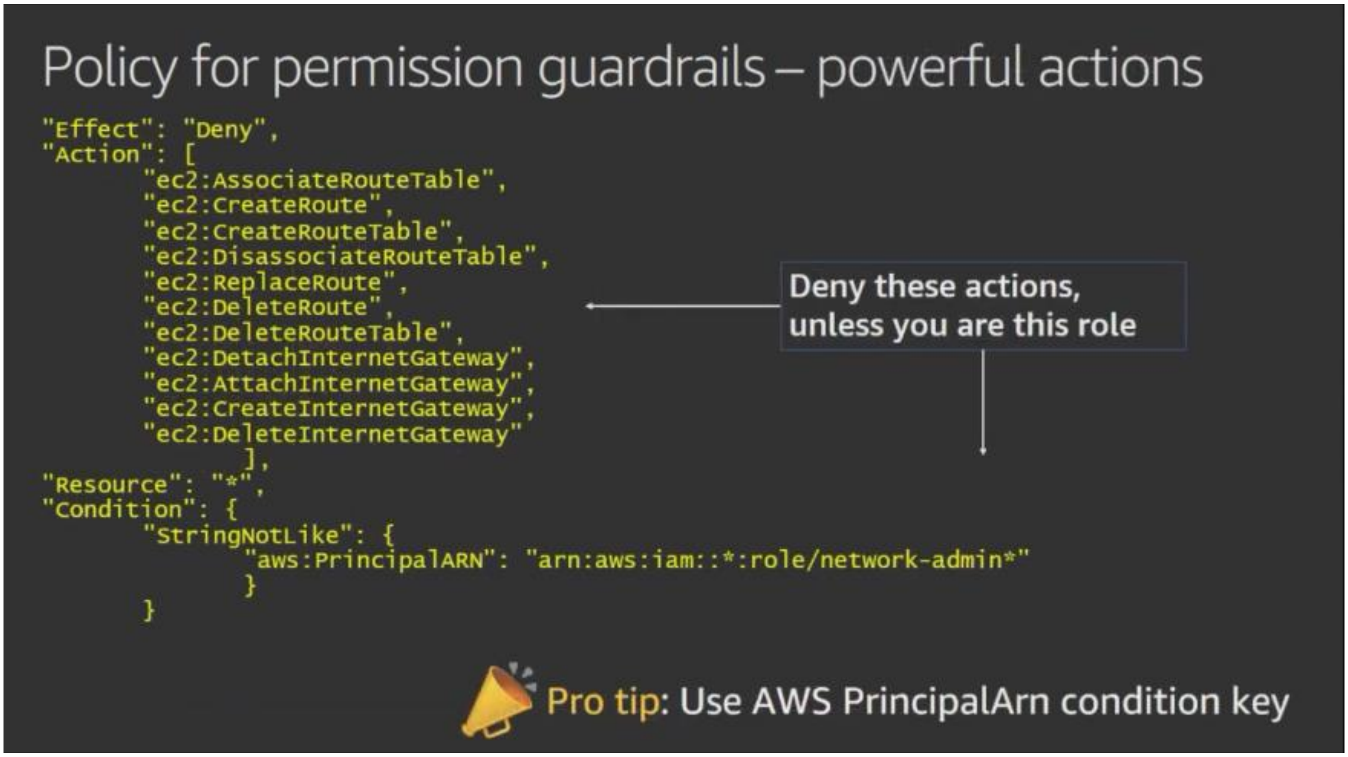


Policy for permission guardrails – restrict regions

```
"Effect": "Deny",
"Action": [
    "codecommit:*",
    "codebuild:*",
    "s3:*",
    "secretsmanager:*",
    "elasticbeanstalk:*],
"Resource": ["*"],
"Condition": {
    "StringNotEquals": {
            "aws:RequestedRegion":
                 ["us-west-2","us-west-1","us-east-1","us-east-2"]
     }}}]}
```

Deny these services, when not in these regions

Pro tip: Use AWS RequestedRegion condition key

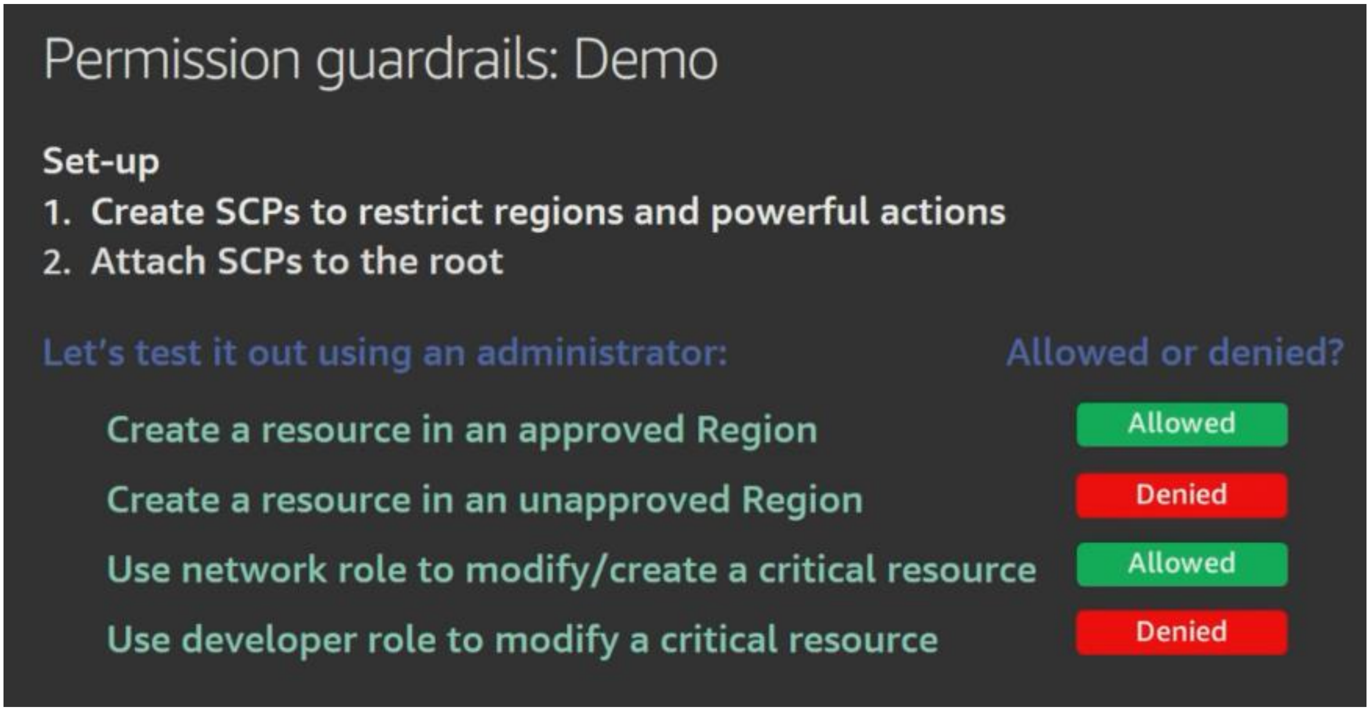


Policy for permission guardrails – powerful actions

```
"Effect": "Deny",
"Action": [
       "ec2:AssociateRouteTable",
       "ec2:CreateRoute",
       "ec2:CreateRouteTable",
       "ec2:DisassociateRouteTable",
       "ec2:ReplaceRoute",
       "ec2:DeleteRoute",
       "ec2:DeleteRouteTable",
       "ec2:DetachInternetGateway",
       "ec2:AttachInternetGateway",
       "ec2:CreateInternetGateway",
       "ec2:DeleteInternetGateway"
              ],
"Resource": "*",
"Condition": {
       "StringNotLike": {
              "aws:PrincipalARN": "arn:aws:iam::*:role/network-admin*"
              }
       }
```

Deny these actions, unless you are this role

Pro tip: Use AWS PrincipalArn condition key

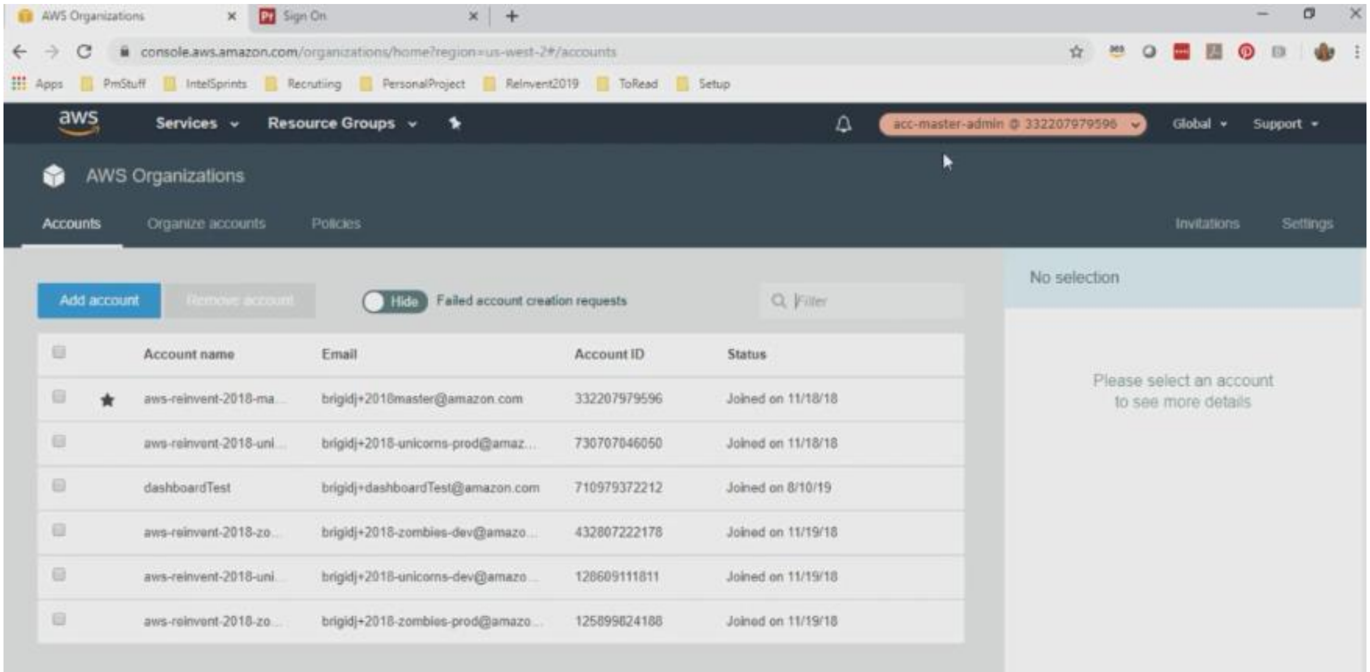


Permission guardrails: Demo

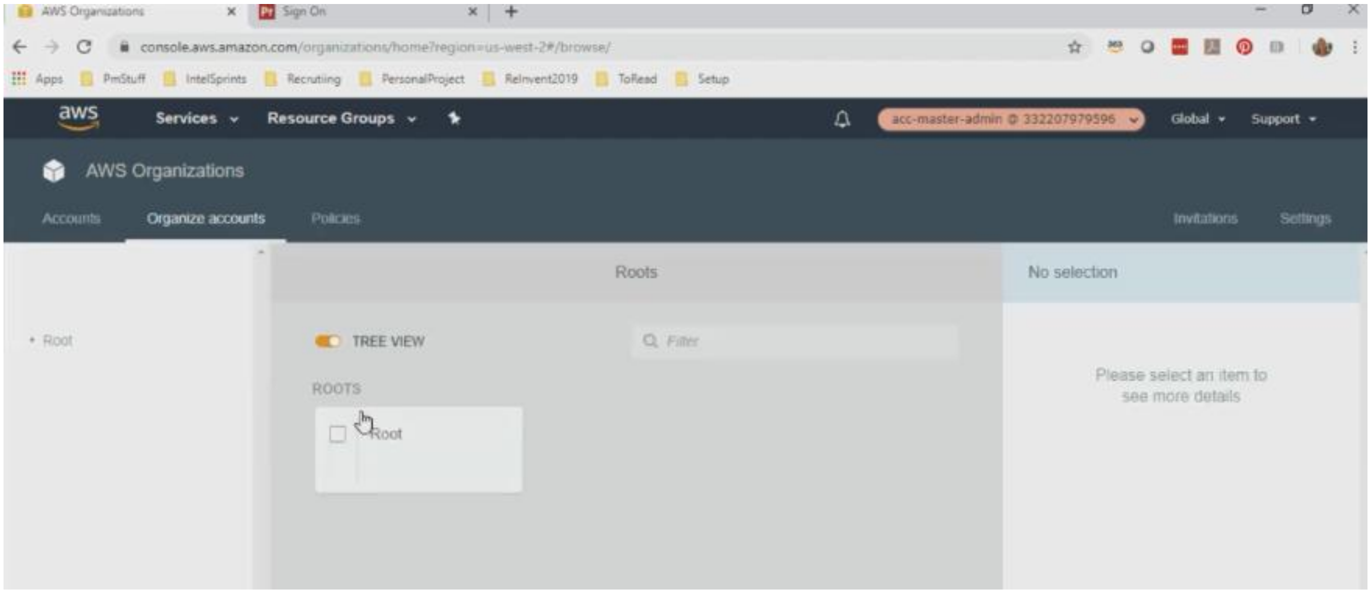
Set-up

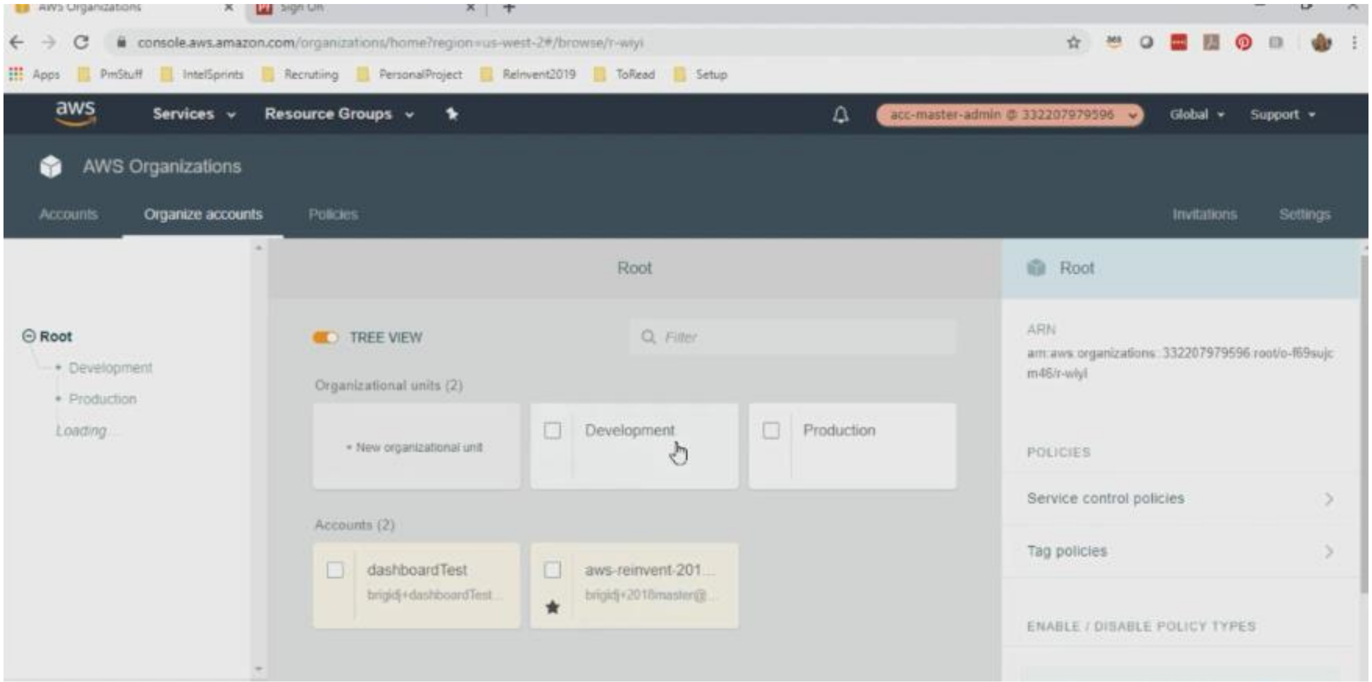
1. **Create SCPs to restrict regions and powerful actions**
2. **Attach SCPs to the root**

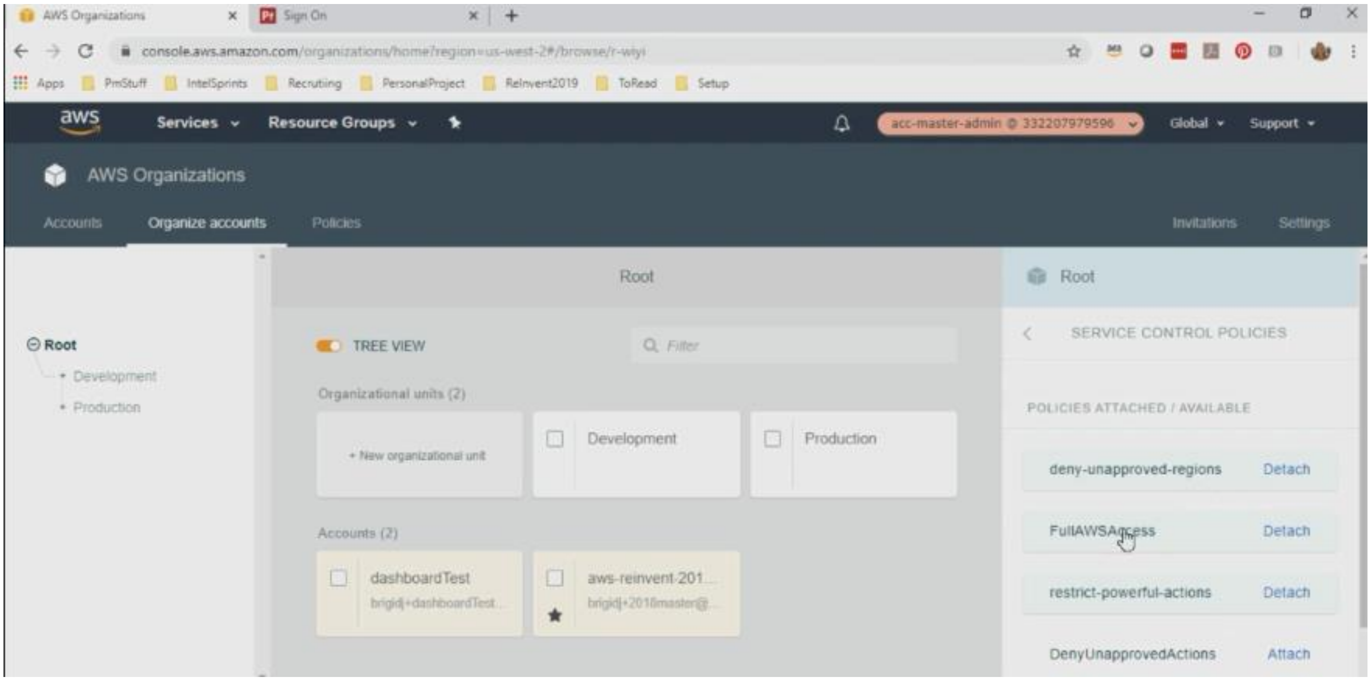
Let's test it out using an administrator:	Allowed or denied?
Create a resource in an approved Region	Allowed
Create a resource in an unapproved Region	Denied
Use network role to modify/create a critical resource	Allowed
Use developer role to modify a critical resource	Denied



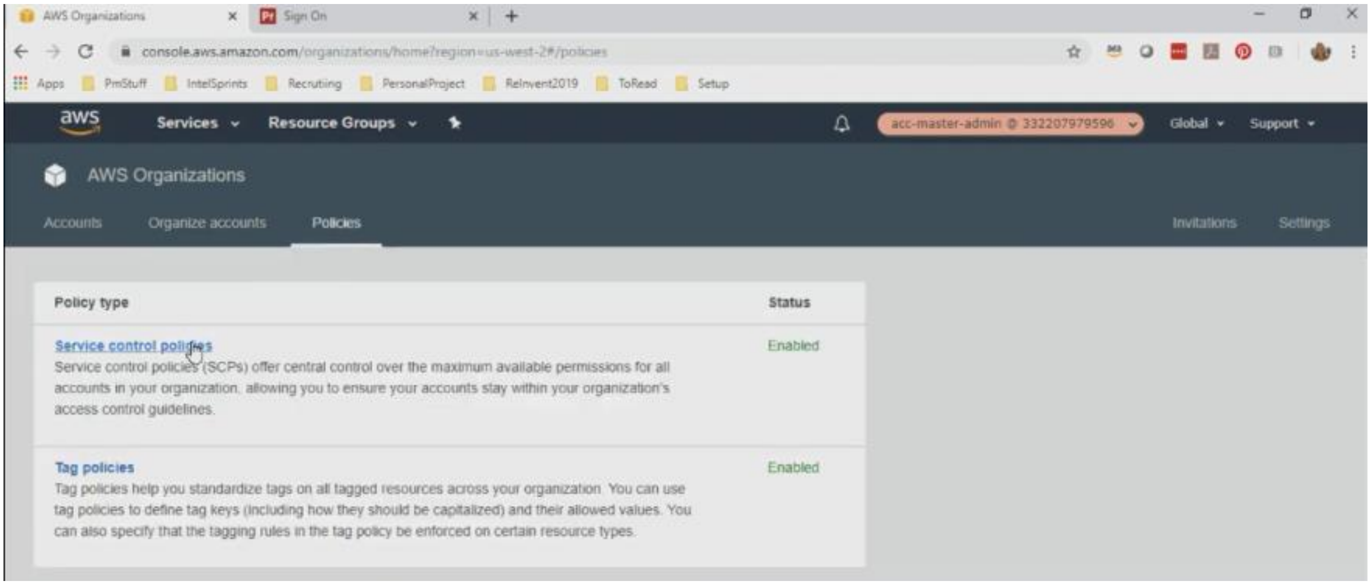
□		Account name	Email	Account ID	Status
□	★	aws-reinvent-2018-ma...	brigidj+2018master@amazon.com	332207979596	Joined on 11/18/18
□		aws-reinvent-2018-uni...	brigidj+2018-unicorns-prod@amaz...	730707046050	Joined on 11/18/18
□		dashboardTest	brigidj+dashboardTest@amazon.com	710979372212	Joined on 8/10/19
□		aws-reinvent-2018-zo...	brigidj+2018-zombies-dev@amazo...	432807222178	Joined on 11/19/18
□		aws-reinvent-2018-uni...	brigidj+2018-unicorns-dev@amazo...	128609111811	Joined on 11/19/18
□		aws-reinvent-2018-zo...	brigidj+2018-zombies-prod@amazo...	125899824188	Joined on 11/19/18


AWS Organizations
Sign On
console.aws.amazon.com/organizations/home?region=us-west-2#/browse/
Apps
PmStuff
IntelSprints
Recrutiing
PersonalProject
ReInvent2019
ToRead
Setup
aws
Services
Resource Groups
acc-master-admin @ 332207979596
Global
Support
AWS Organizations
Accounts
Organize accounts
Policies
Invitations
Settings
Root
Roots
TREE VIEW
Filter
ROOTS
□ Root
No selection
Please select an item to see more details

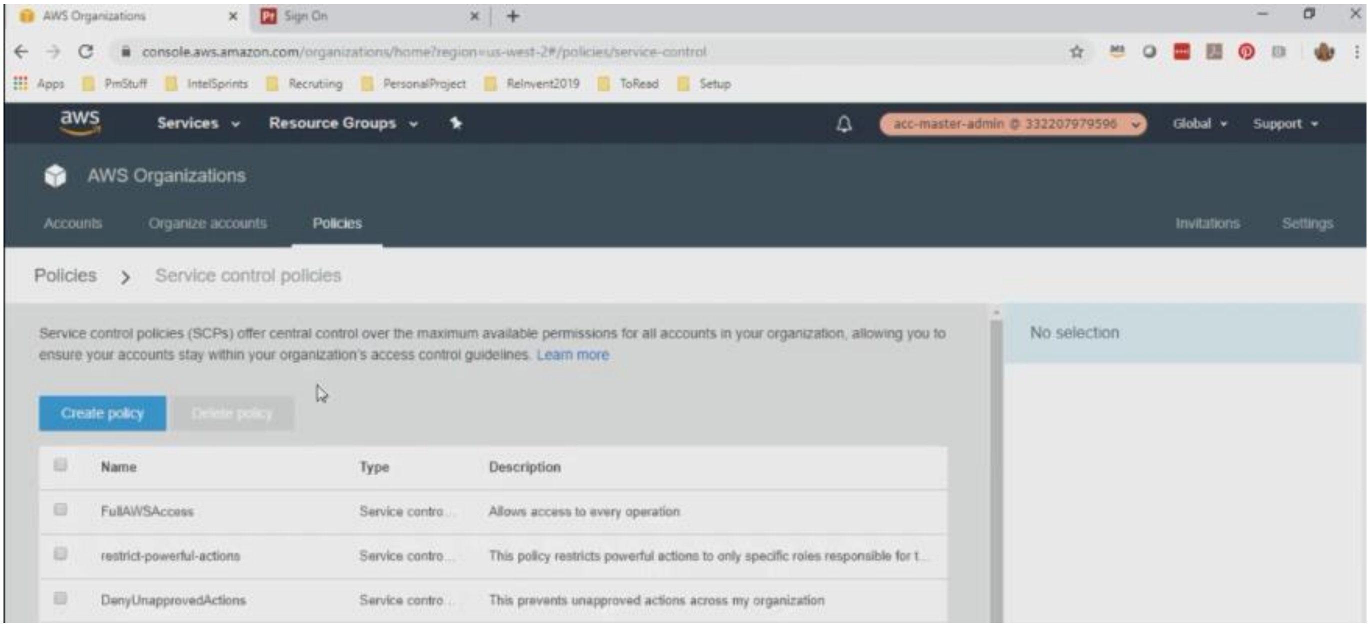

AWS Organizations
Sign On
console.aws.amazon.com/organizations/home?region=us-west-2#/browse/r-wlyl
Apps
PmStuff
IntelSprints
Recrutiing
PersonalProject
ReInvent2019
ToRead
Setup
aws
Services
Resource Groups
acc-master-admin @ 332207979596
Global
Support
AWS Organizations
Accounts
Organize accounts
Policies
Invitations
Settings
Root
Development
Production
Loading ...
Root
TREE VIEW
Filter
Organizational units (2)
+ New organizational unit
□ Development
□ Production
Accounts (2)
□ dashboardTest
brigidj+dashboardTest...
□ aws-reinvent-201...
brigidj+2018master@...
Root
ARN
arn:aws:organizations::332207979596:root/o-f69sujcm46/r-wlyl
POLICIES
Service control policies
Tag policies
ENABLE / DISABLE POLICY TYPES


AWS Organizations
Sign On
console.aws.amazon.com/organizations/home?region=us-west-2#/browse/r-wiyi
Apps PmStuff IntelSprints Recruting PersonalProject ReInvent2019 ToRead Setup
aws Services Resource Groups
acc-master-admin @ 332207979596
Global Support
AWS Organizations
Accounts Organize accounts Policies
Invitations Settings
Root
TREE VIEW
Filter
Root
Development
Production
Organizational units (2)
+ New organizational unit
Development
Production
Accounts (2)
dashboardTest
brigid+dashboardTest...
aws-reinvent-201...
brigid+2018master@...
SERVICE CONTROL POLICIES
POLICIES ATTACHED / AVAILABLE

deny-unapproved-regions	Detach
FullAWSAccess	Detach
restrict-powerful-actions	Detach
DenyUnapprovedActions	Attach

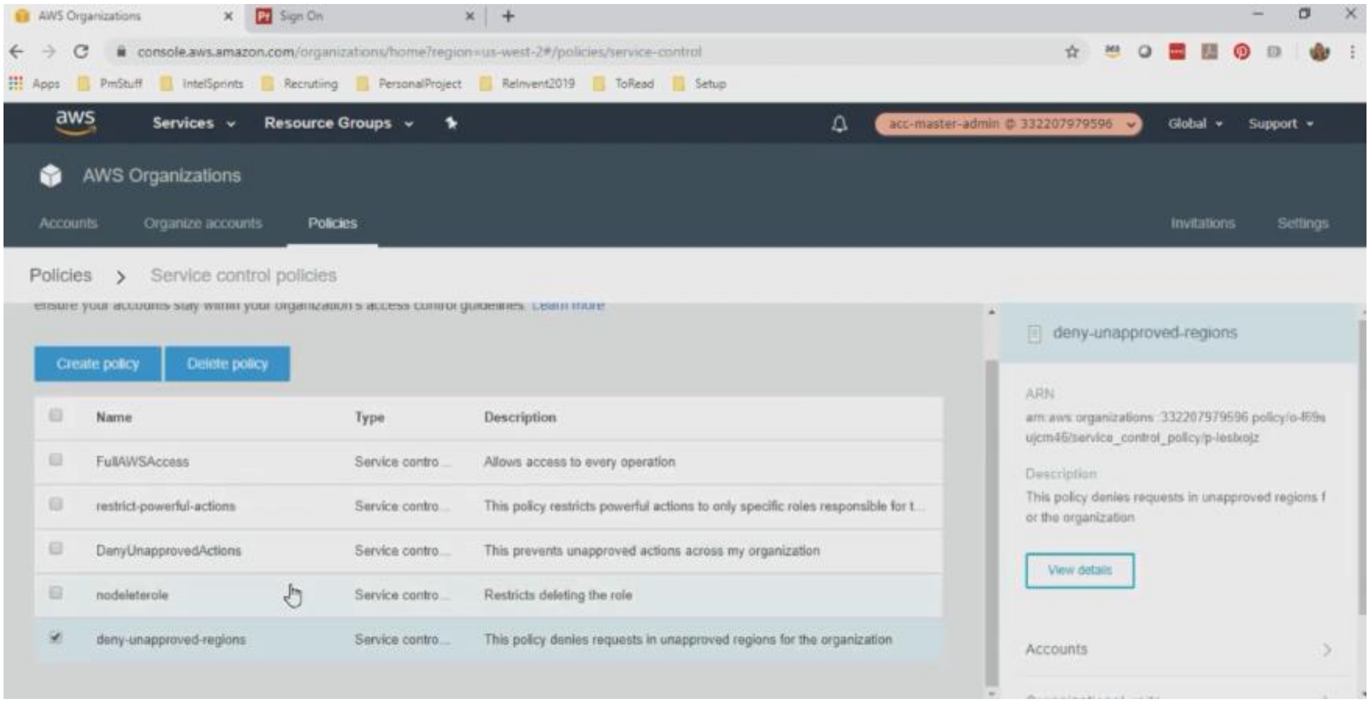

AWS Organizations
Sign On
console.aws.amazon.com/organizations/home?region=us-west-2#/policies
Apps PmStuff IntelSprints Recruting PersonalProject ReInvent2019 ToRead Setup
aws Services Resource Groups
acc-master-admin @ 332207979596
Global Support
AWS Organizations
Accounts Organize accounts Policies
Invitations Settings

Policy type	Status
Service control policies Service control policies (SCPs) offer central control over the maximum available permissions for all accounts in your organization, allowing you to ensure your accounts stay within your organization's access control guidelines.	Enabled
Tag policies Tag policies help you standardize tags on all tagged resources across your organization. You can use tag policies to define tag keys (including how they should be capitalized) and their allowed values. You can also specify that the tagging rules in the tag policy be enforced on certain resource types.	Enabled


AWS Organizations
Sign On
console.aws.amazon.com/organizations/home?region=us-west-2#/policies/service-control
Apps PmStuff IntelSprints Recruting PersonalProject ReInvent2019 ToRead Setup
aws Services Resource Groups
acc-master-admin @ 332207979596
Global Support
AWS Organizations
Accounts Organize accounts Policies
Invitations Settings
Policies > Service control policies
No selection
Create policy
Delete policy

Service control policies (SCPs) offer central control over the maximum available permissions for all accounts in your organization, allowing you to ensure your accounts stay within your organization's access control guidelines. Learn more

	Name	Type	Description
□	FullAWSAccess	Service contro...	Allows access to every operation
□	restrict-powerful-actions	Service contro...	This policy restricts powerful actions to only specific roles responsible for t...
□	DenyUnapprovedActions	Service contro...	This prevents unapproved actions across my organization



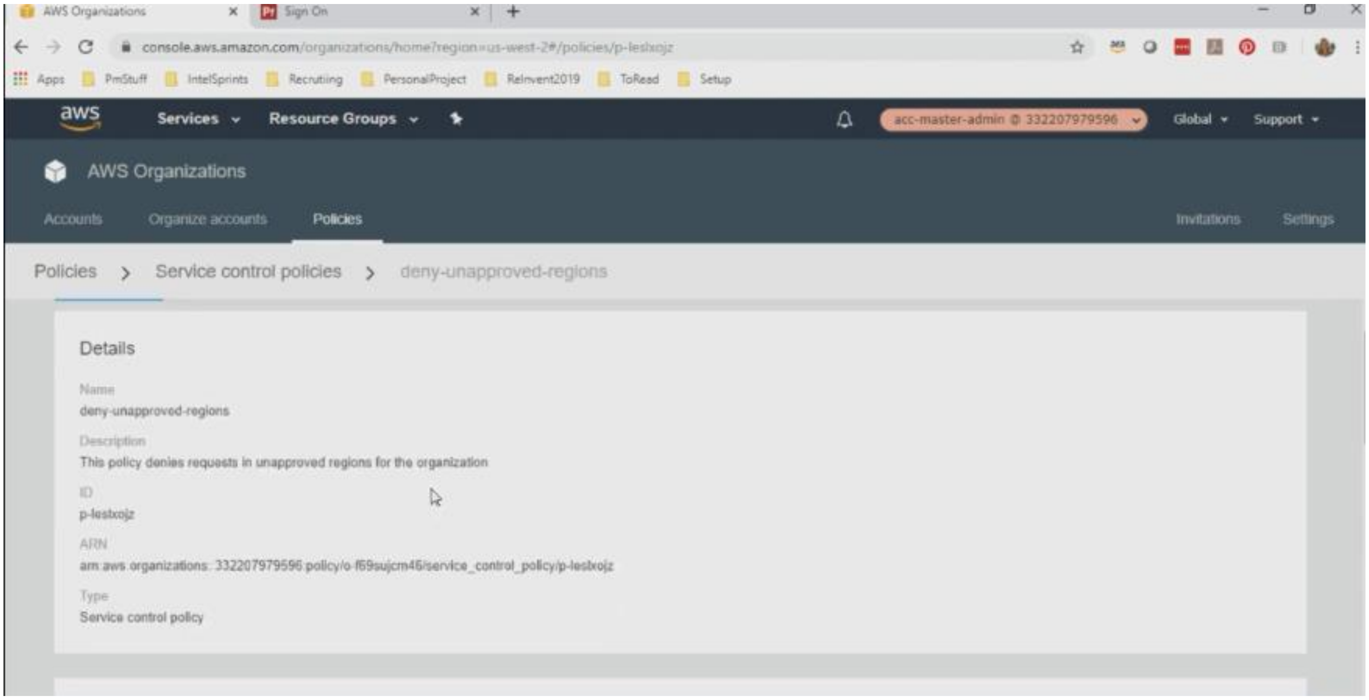
ensure your accounts stay within your organization's access control guidelines. Learn more

□	Name	Type	Description
□	FullAWSAccess	Service contro...	Allows access to every operation
□	restrict-powerful-actions	Service contro...	This policy restricts powerful actions to only specific roles responsible for t...
□	DenyUnapprovedActions	Service contro...	This prevents unapproved actions across my organization
□	nodeleterole	Service contro...	Restricts deleting the role
☑	deny-unapproved-regions	Service contro...	This policy denies requests in unapproved regions for the organization

deny-unapproved-regions

ARN
arn:aws:organizations::332207979596:policy/o-f69sujcm46/service_control_policy/p-lesbxojz

Description
This policy denies requests in unapproved regions for the organization



Details

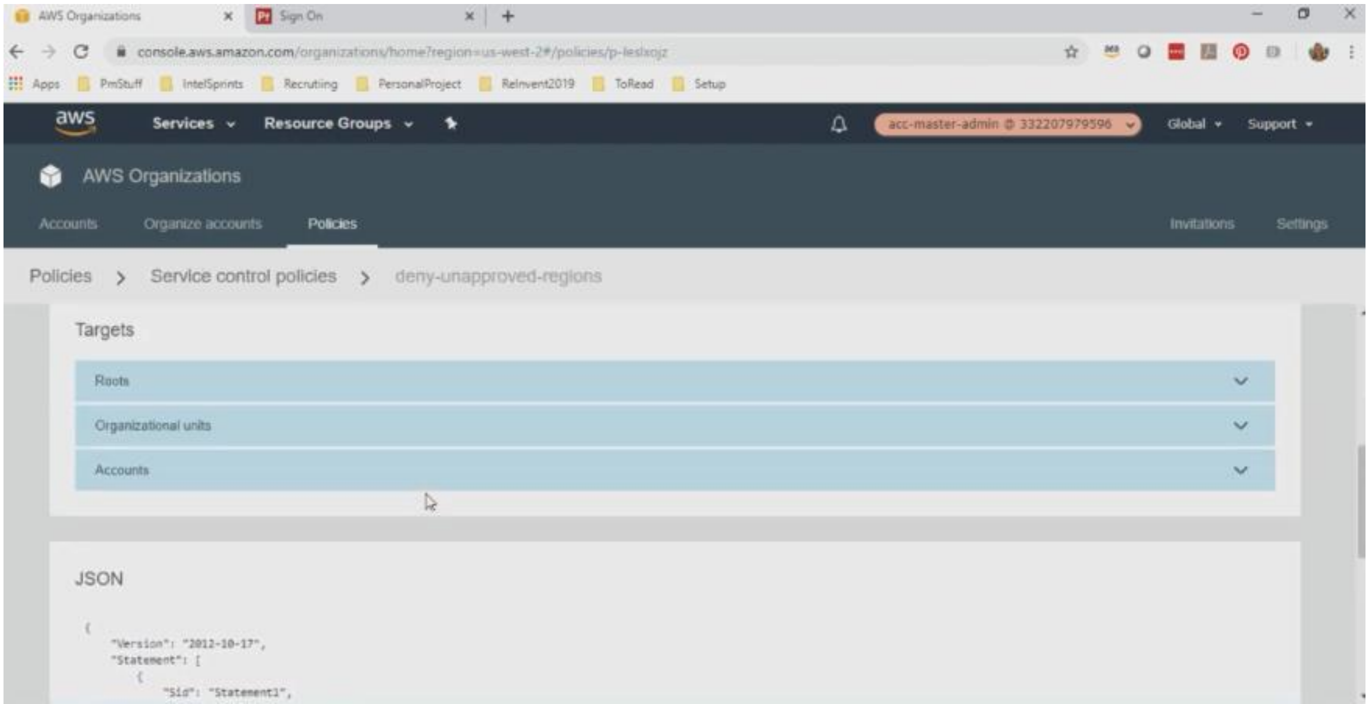
Name
deny-unapproved-regions

Description
This policy denies requests in unapproved regions for the organization

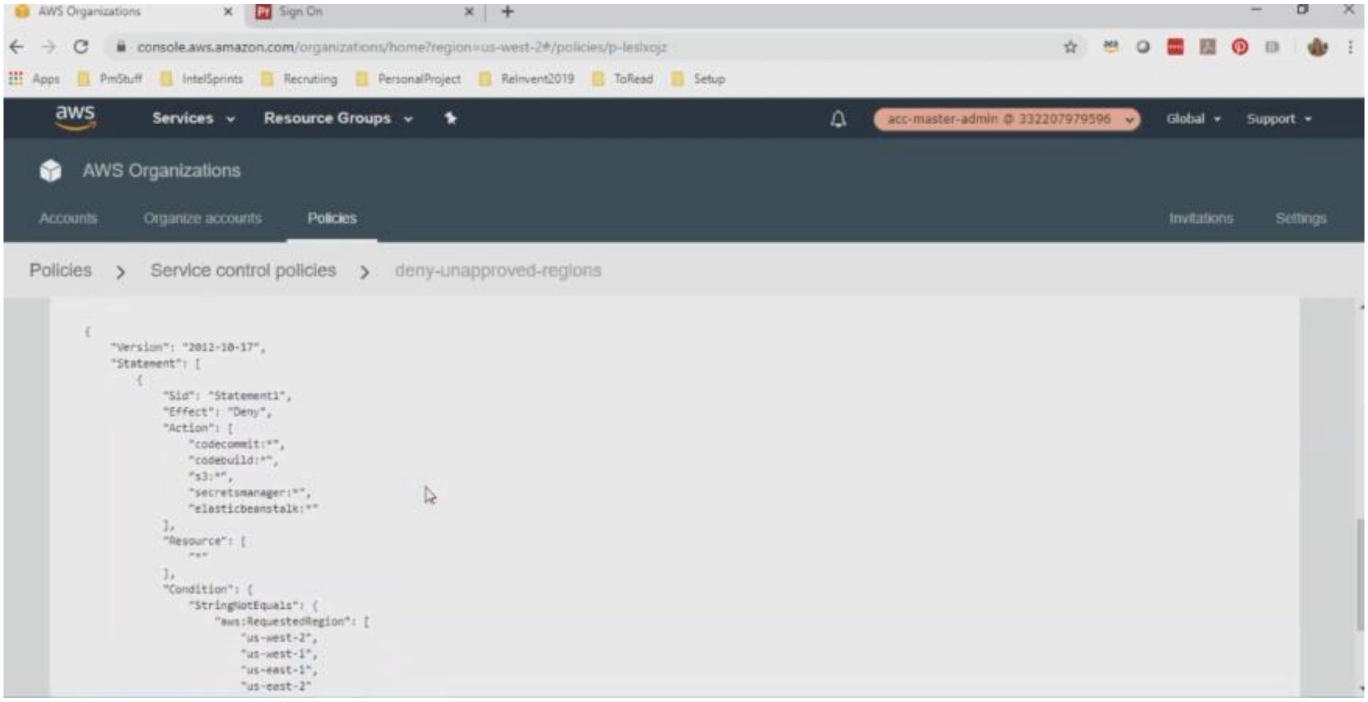
ID
p-lesbxojz

ARN
arn:aws:organizations::332207979596:policy/o-f69sujcm46/service_control_policy/p-lesbxojz

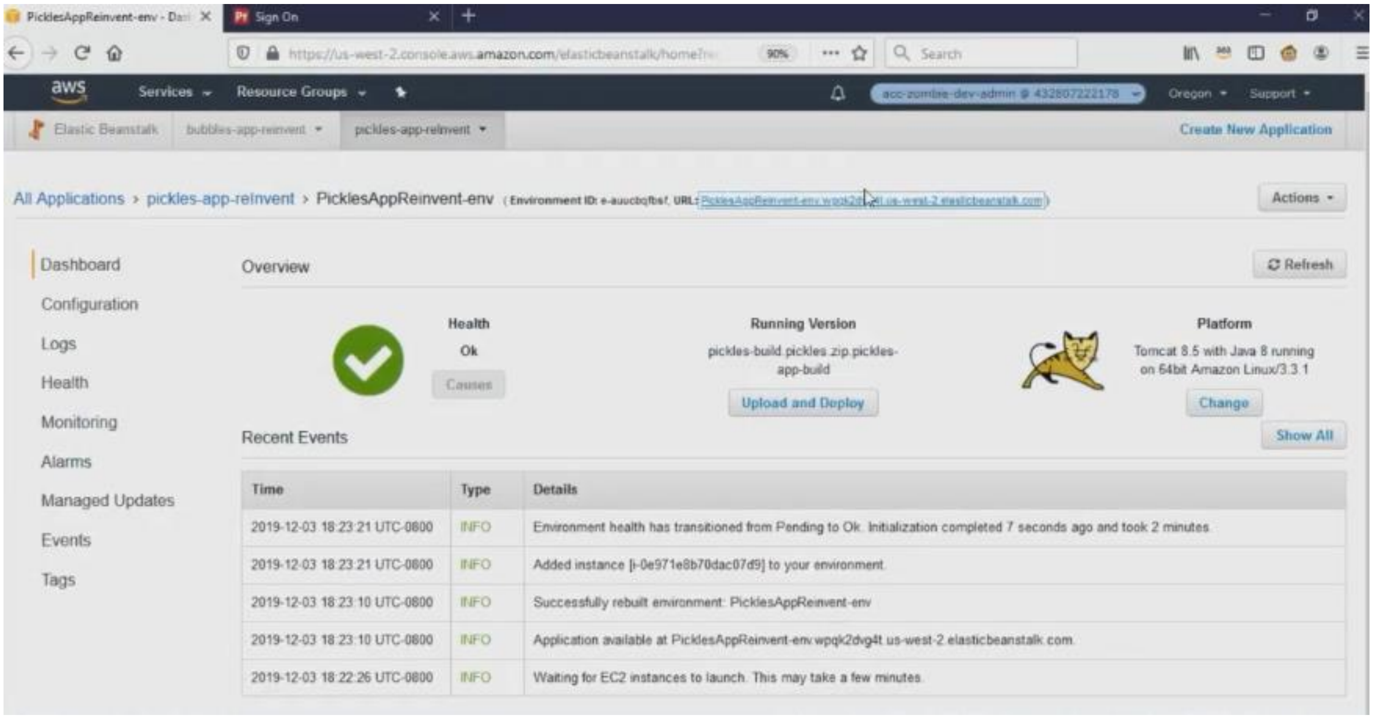
Type
Service control policy



```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Statement1",
```



```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Statement1",
            "Effect": "Deny",
            "Action": [
                "codecommit:*",
                "codebuild:*",
                "s3:*",
                "secretsmanager:*",
                "elasticbeanstalk:*"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:RequestedRegion": [
                        "us-west-2",
                        "us-west-1",
                        "us-east-1",
                        "us-east-2"
```



All Applications › pickles-app-reInvent › PicklesAppReinvent-env (Environment ID: e-auucbqfbsf, URL: PicklesAppReinvent-env.wpqk2dvg4t.us-west-2.elasticbeanstalk.com)

Overview

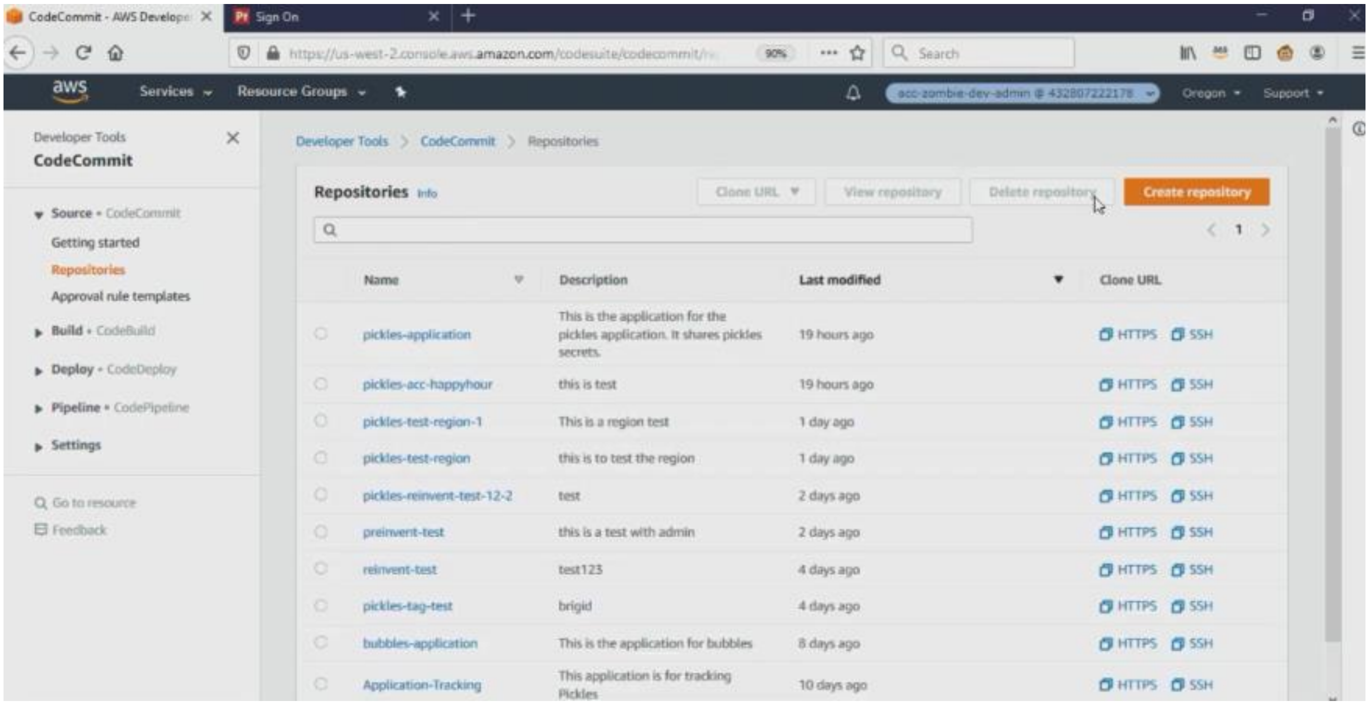
Health
Ok

Running Version
pickles-build.pickles.zip.pickles-app-build

Platform
Tomcat 8.5 with Java 8 running on 64bit Amazon Linux/3.3.1

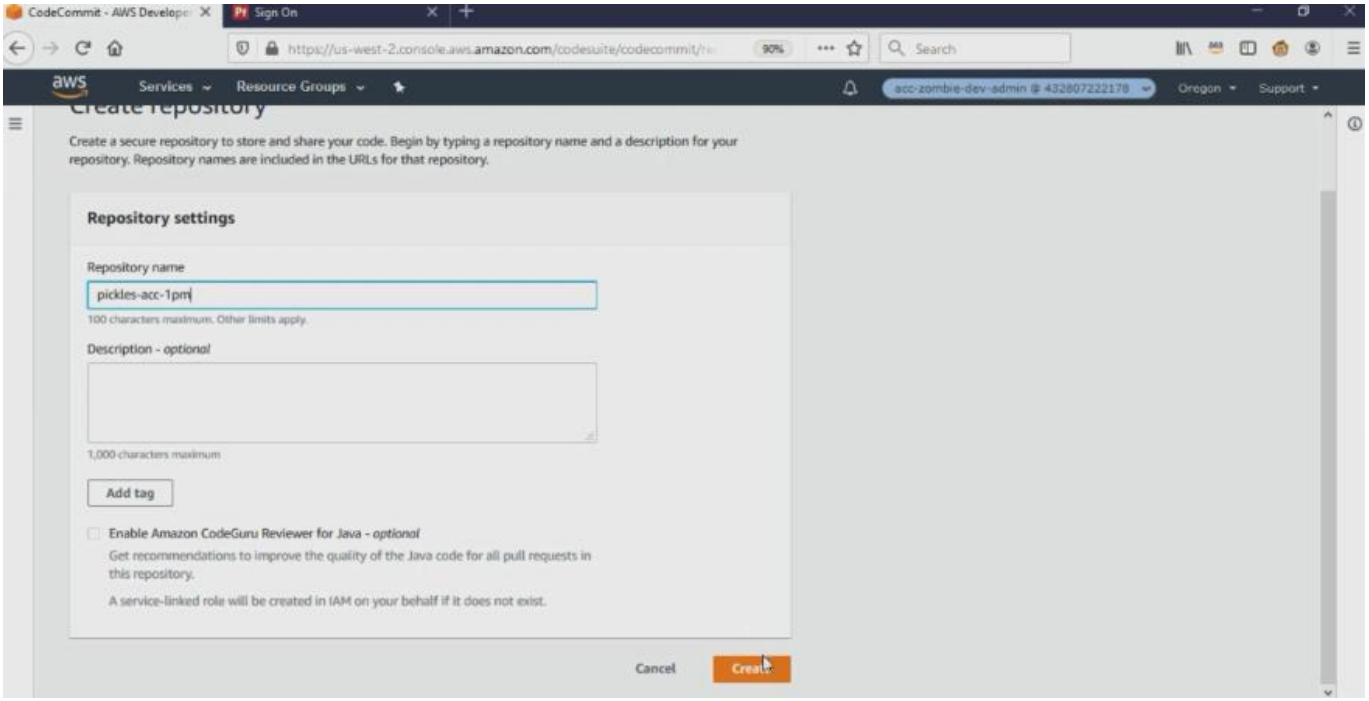
Recent Events

Time	Type	Details
2019-12-03 18:23:21 UTC-0800	INFO	Environment health has transitioned from Pending to Ok. Initialization completed 7 seconds ago and took 2 minutes.
2019-12-03 18:23:21 UTC-0800	INFO	Added instance [i-0e971e8b70dac07d9] to your environment.
2019-12-03 18:23:10 UTC-0800	INFO	Successfully rebuilt environment: PicklesAppReinvent-env
2019-12-03 18:23:10 UTC-0800	INFO	Application available at PicklesAppReinvent-env.wpqk2dvg4t.us-west-2.elasticbeanstalk.com.
2019-12-03 18:22:26 UTC-0800	INFO	Waiting for EC2 instances to launch. This may take a few minutes.



Repositories

	Name	Description	Last modified	Clone URL
○	pickles-application	This is the application for the pickles application. It shares pickles secrets.	19 hours ago	HTTPS SSH
○	pickles-acc-happyhour	this is test	19 hours ago	HTTPS SSH
○	pickles-test-region-1	This is a region test	1 day ago	HTTPS SSH
○	pickles-test-region	this is to test the region	1 day ago	HTTPS SSH
○	pickles-reinvent-test-12-2	test	2 days ago	HTTPS SSH
○	preinvent-test	this is a test with admin	2 days ago	HTTPS SSH
○	reinvent-test	test123	4 days ago	HTTPS SSH
○	pickles-tag-test	brigid	4 days ago	HTTPS SSH
○	bubbles-application	This is the application for bubbles	8 days ago	HTTPS SSH
○	Application-Tracking	This application is for tracking Pickles	10 days ago	HTTPS SSH


CodeCommit - AWS Develope
Sign On
https://us-west-2.console.aws.amazon.com/codesuite/codecommit/re
90%
Search
aws
Services
Resource Groups
acc-zombie-dev-admin @ 432807222178
Oregon
Support

Create repository

Create a secure repository to store and share your code. Begin by typing a repository name and a description for your repository. Repository names are included in the URLs for that repository.

Repository settings

Repository name

pickles-acc-1pm

100 characters maximum. Other limits apply.

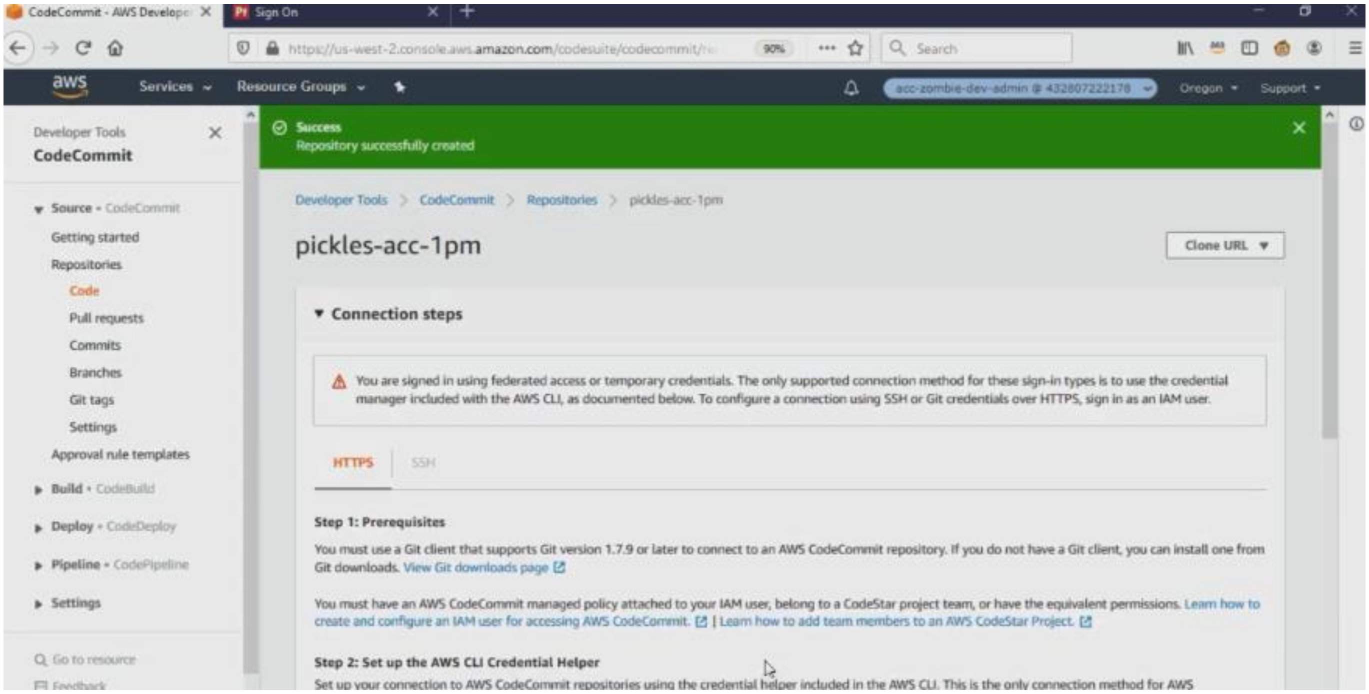
Description - *optional*

1,000 characters maximum

☐ Enable Amazon CodeGuru Reviewer for Java - *optional*

Get recommendations to improve the quality of the Java code for all pull requests in this repository.

A service-linked role will be created in IAM on your behalf if it does not exist.



Success
Repository successfully created

Developer Tools > CodeCommit > Repositories > pickles-acc-1pm

pickles-acc-1pm

▼ **Connection steps**

⚠ You are signed in using federated access or temporary credentials. The only supported connection method for these sign-in types is to use the credential manager included with the AWS CLI, as documented below. To configure a connection using SSH or Git credentials over HTTPS, sign in as an IAM user.

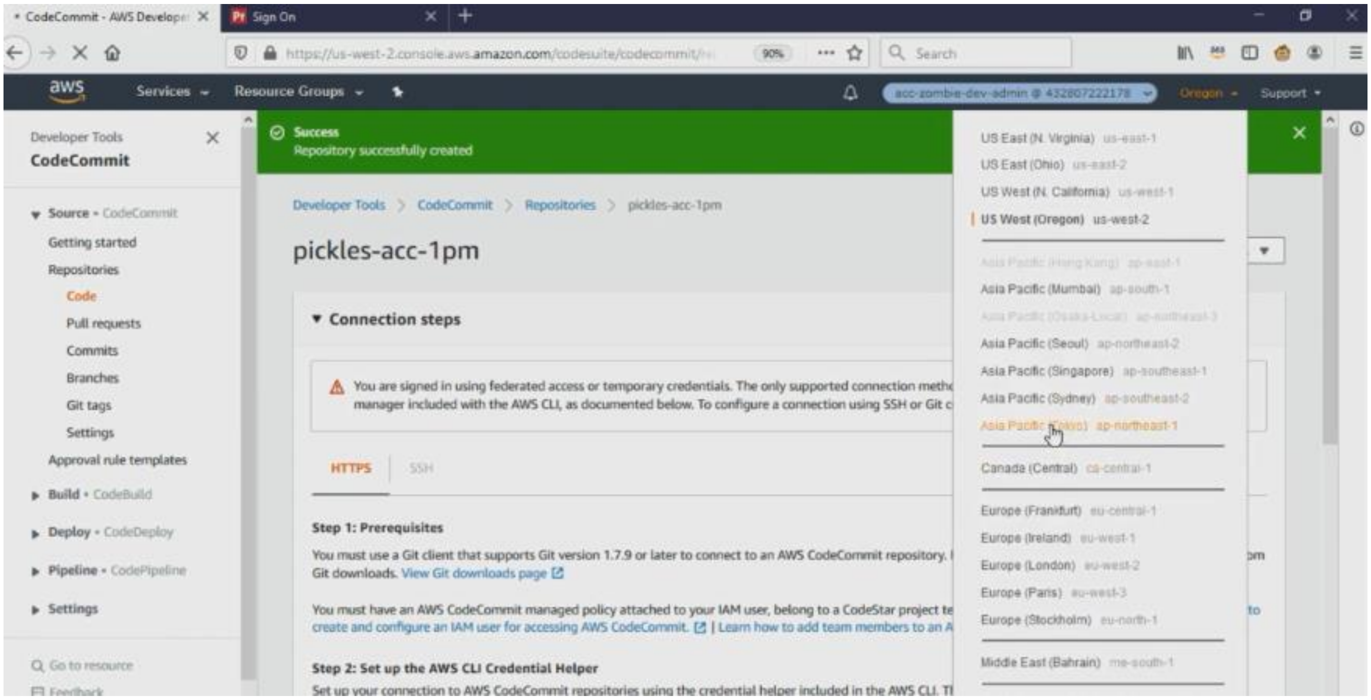
Step 1: Prerequisites

You must use a Git client that supports Git version 1.7.9 or later to connect to an AWS CodeCommit repository. If you do not have a Git client, you can install one from Git downloads. View Git downloads page

You must have an AWS CodeCommit managed policy attached to your IAM user, belong to a CodeStar project team, or have the equivalent permissions. Learn how to create and configure an IAM user for accessing AWS CodeCommit. | Learn how to add team members to an AWS CodeStar Project.

Step 2: Set up the AWS CLI Credential Helper

Set up your connection to AWS CodeCommit repositories using the credential helper included in the AWS CLI. This is the only connection method for AWS



CodeCommit - AWS Develope
Sign On
https://us-west-2.console.aws.amazon.com/codesuite/codecommit/
90%
Search
Services
Resource Groups
acc-zombie-dev-admin @ 432807222178
Oregon
Support
Developer Tools
CodeCommit
Source • CodeCommit
Getting started
Repositories
Code
Pull requests
Commits
Branches
Git tags
Settings
Approval rule templates
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Settings
Go to resource
Feedback
Developer Tools > CodeCommit > Repositories > pickles-acc-1pm
US East (N. Virginia) us-east-1
US East (Ohio) us-east-2
US West (N. California) us-west-1
US West (Oregon) us-west-2
Asia Pacific (Hong Kong) ap-east-1
Asia Pacific (Mumbai) ap-south-1
Asia Pacific (Osaka-Local) ap-northeast-3
Asia Pacific (Seoul) ap-northeast-2
Asia Pacific (Singapore) ap-southeast-1
Asia Pacific (Sydney) ap-southeast-2
Asia Pacific (Tokyo) ap-northeast-1
Canada (Central) ca-central-1
Europe (Frankfurt) eu-central-1
Europe (Ireland) eu-west-1
Europe (London) eu-west-2
Europe (Paris) eu-west-3
Europe (Stockholm) eu-north-1
Middle East (Bahrain) me-south-1
HTTPS
SSH

Success
Repository successfully created

pickles-acc-1pm

Connection steps

You are signed in using federated access or temporary credentials. The only supported connection meth
manager included with the AWS CLI, as documented below. To configure a connection using SSH or Git c

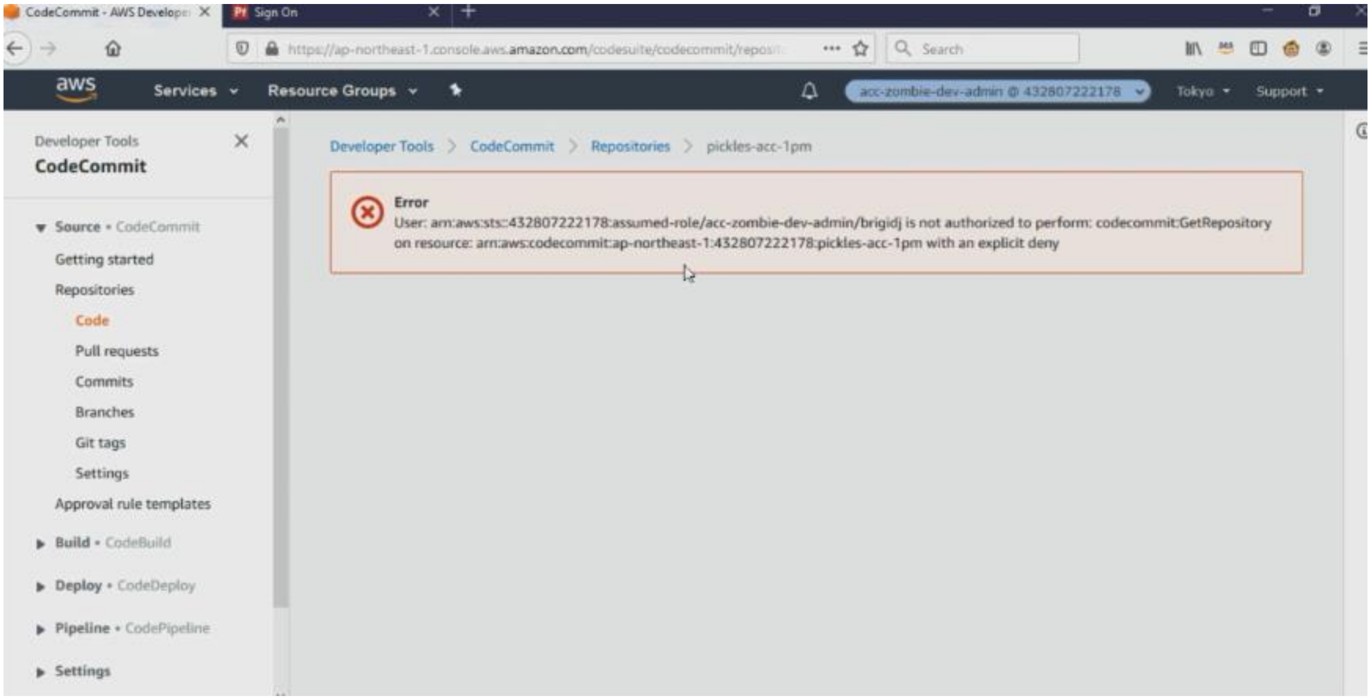
Step 1: Prerequisites

You must use a Git client that supports Git version 1.7.9 or later to connect to an AWS CodeCommit repository. I
Git downloads. View Git downloads page

You must have an AWS CodeCommit managed policy attached to your IAM user, belong to a CodeStar project te
create and configure an IAM user for accessing AWS CodeCommit. | Learn how to add team members to an A

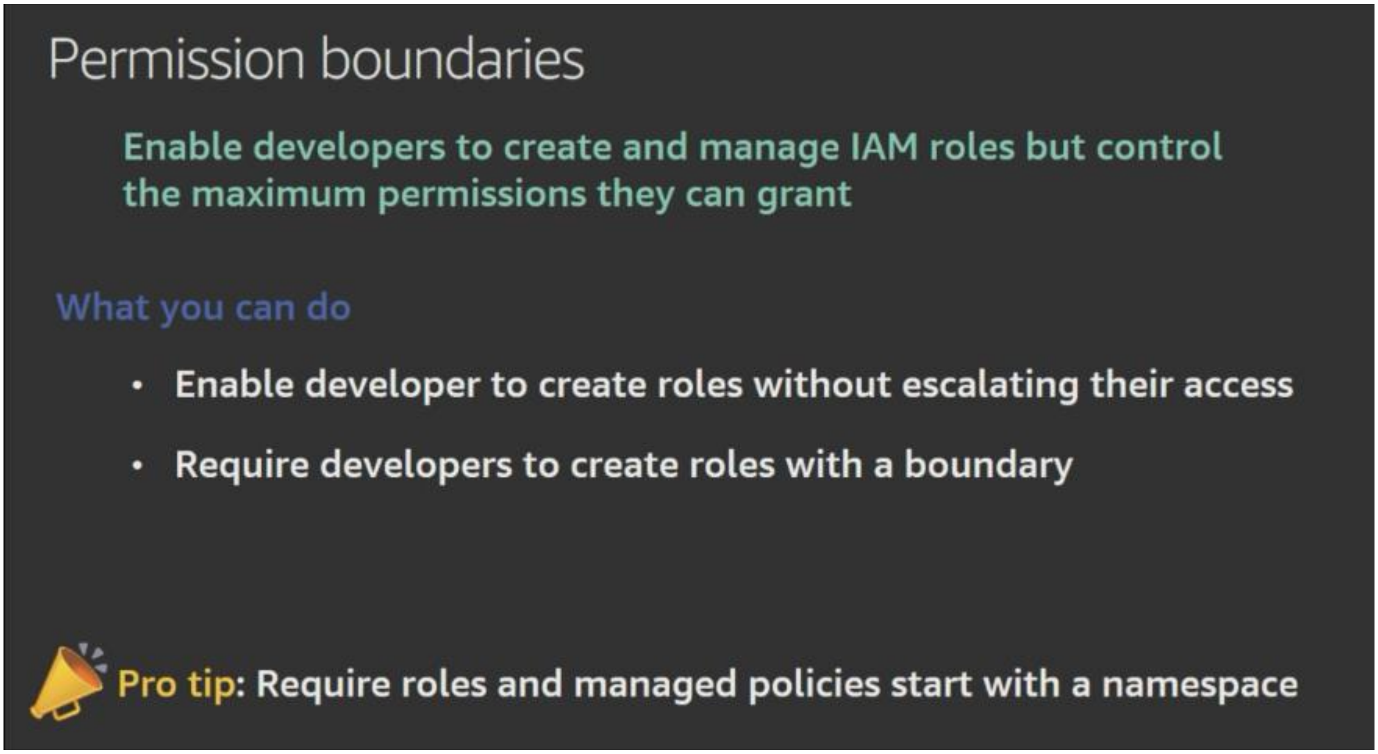
Step 2: Set up the AWS CLI Credential Helper

Set up your connection to AWS CodeCommit repositories using the credential helper included in the AWS CLI. Tl



Error
User: arn:aws:sts::432807222178:assumed-role/acc-zombie-dev-admin/brigidj is not authorized to perform: codecommit:GetRepository on resource: arn:aws:codecommit:ap-northeast-1:432807222178:pickles-acc-1pm with an explicit deny

The admin will not be able to do anything in the Tokyo region due to the SCP we created earlier



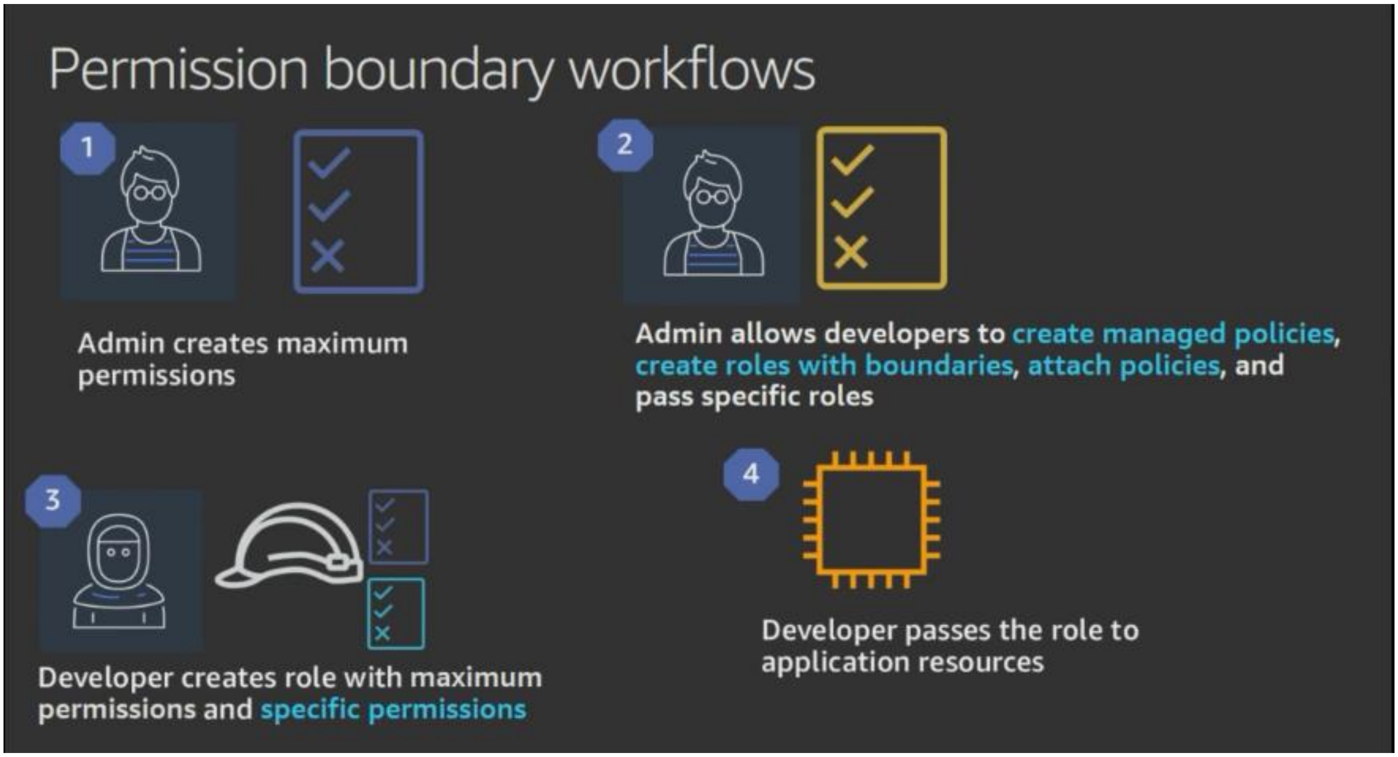
Permission boundaries

Enable developers to create and manage IAM roles but control the maximum permissions they can grant

What you can do

- Enable developer to create roles without escalating their access
- Require developers to create roles with a boundary

Pro tip: Require roles and managed policies start with a namespace



Permission boundary workflows

1 Admin creates maximum permissions

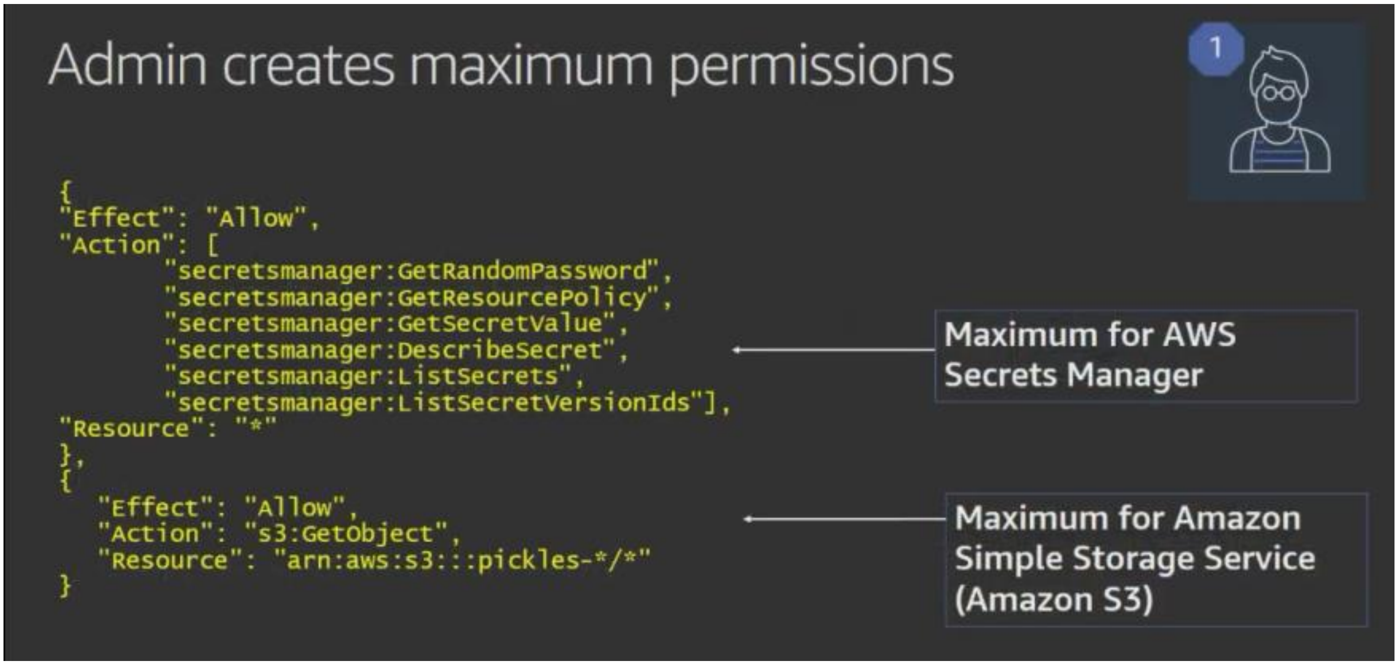
2 Admin allows developers to create managed policies, create roles with boundaries, attach policies, and pass specific roles

3 Developer creates role with maximum permissions and specific permissions

4 Developer passes the role to application resources

Permission boundaries challenge

Enable your developers to create IAM roles to pass to Amazon Elastic Compute Cloud (Amazon EC2) and AWS Lambda, but ensure they cannot exceed the maximum permissions

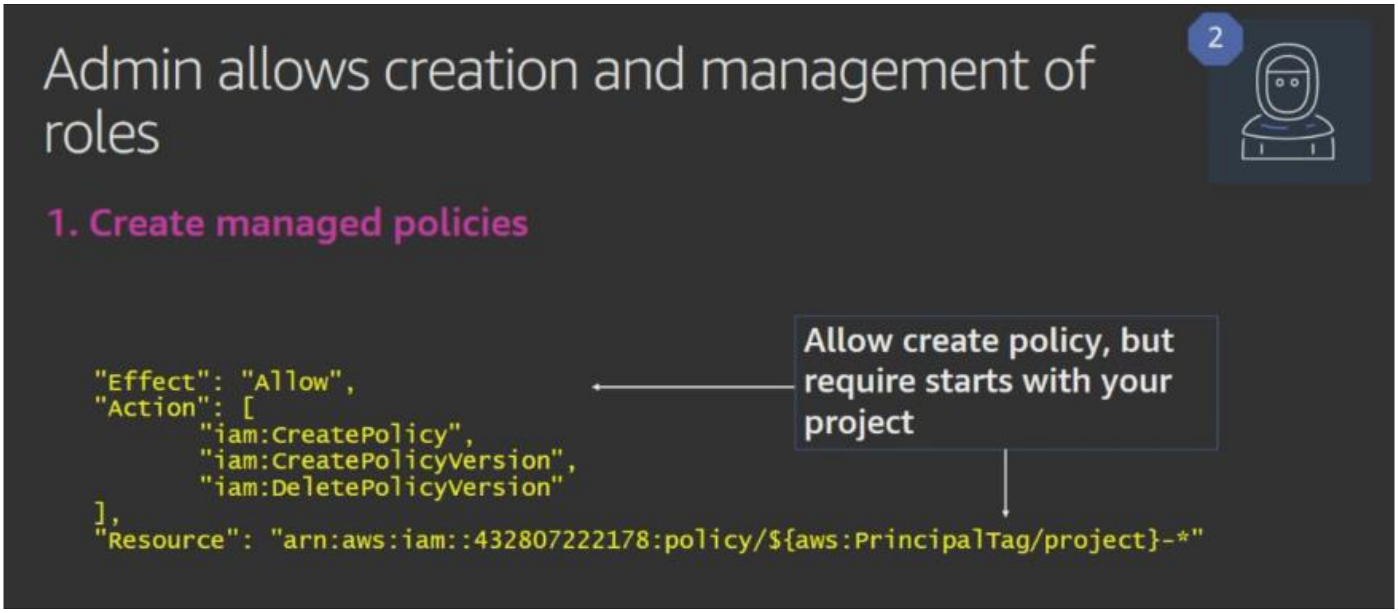


Admin creates maximum permissions

```
{
"Effect": "Allow",
"Action": [
        "secretsmanager:GetRandomPassword",
        "secretsmanager:GetResourcePolicy",
        "secretsmanager:GetSecretValue",
        "secretsmanager:DescribeSecret",
        "secretsmanager:ListSecrets",
        "secretsmanager:ListSecretVersionIds"],
"Resource": "*"
},
{
   "Effect": "Allow",
   "Action": "s3:GetObject",
   "Resource": "arn:aws:s3:::pickles-*/*"
}
```

Maximum for AWS Secrets Manager

Maximum for Amazon Simple Storage Service (Amazon S3)

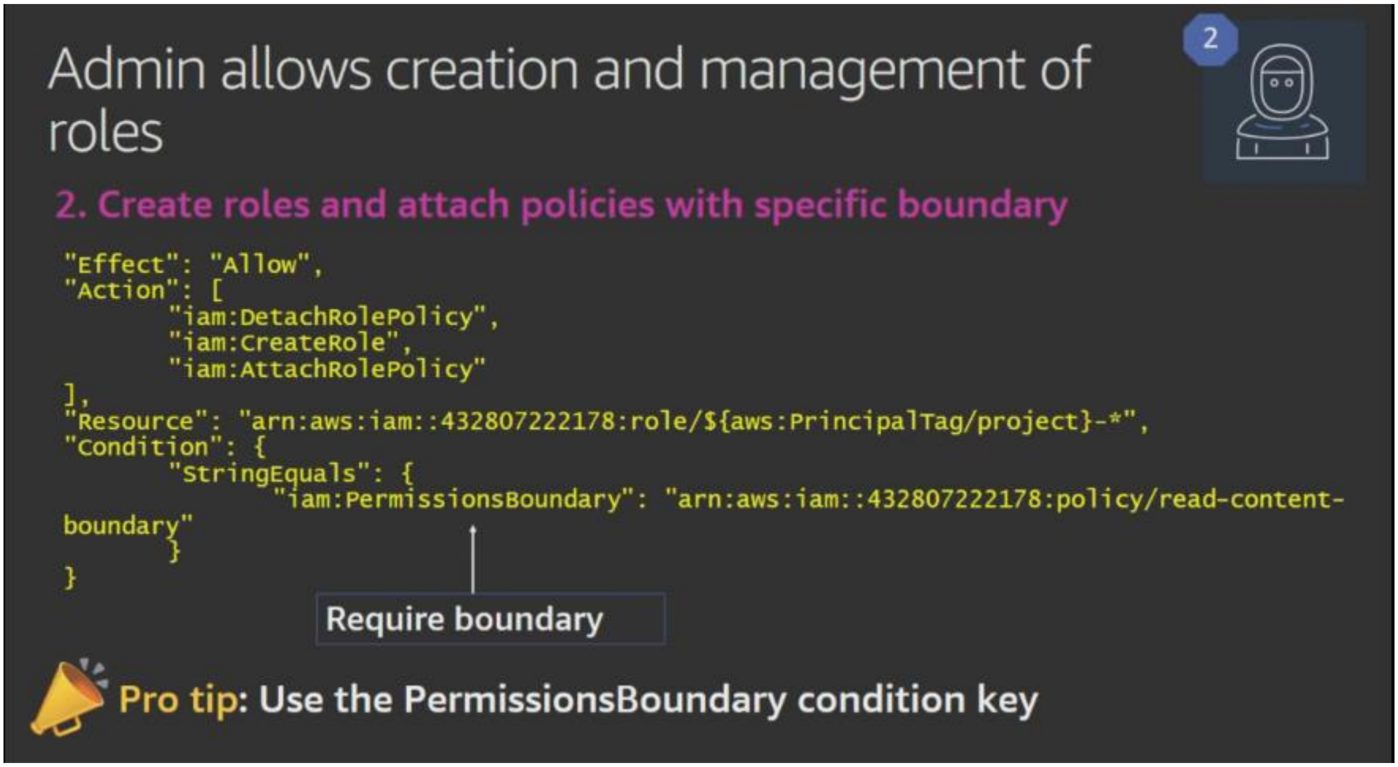


Admin allows creation and management of roles

1. Create managed policies

```
"Effect": "Allow",
"Action": [
        "iam:CreatePolicy",
        "iam:CreatePolicyVersion",
        "iam:DeletePolicyVersion"
],
"Resource": "arn:aws:iam::432807222178:policy/${aws:PrincipalTag/project}-*"
```

Allow create policy, but require starts with your project



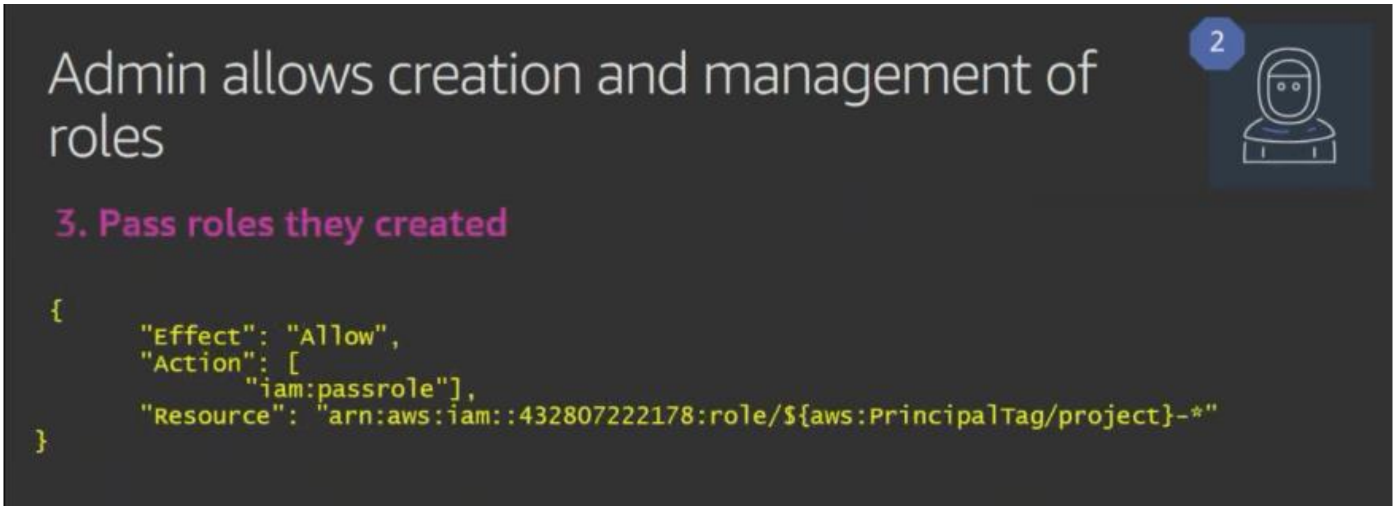
Admin allows creation and management of roles

2. Create roles and attach policies with specific boundary

```
"Effect": "Allow",
"Action": [
       "iam:DetachRolePolicy",
       "iam:CreateRole",
       "iam:AttachRolePolicy"
],
"Resource": "arn:aws:iam::432807222178:role/${aws:PrincipalTag/project}-*",
"Condition": {
       "StringEquals": {
              "iam:PermissionsBoundary": "arn:aws:iam::432807222178:policy/read-content-
boundary"
       }
}
```

Require boundary

Pro tip: Use the PermissionsBoundary condition key

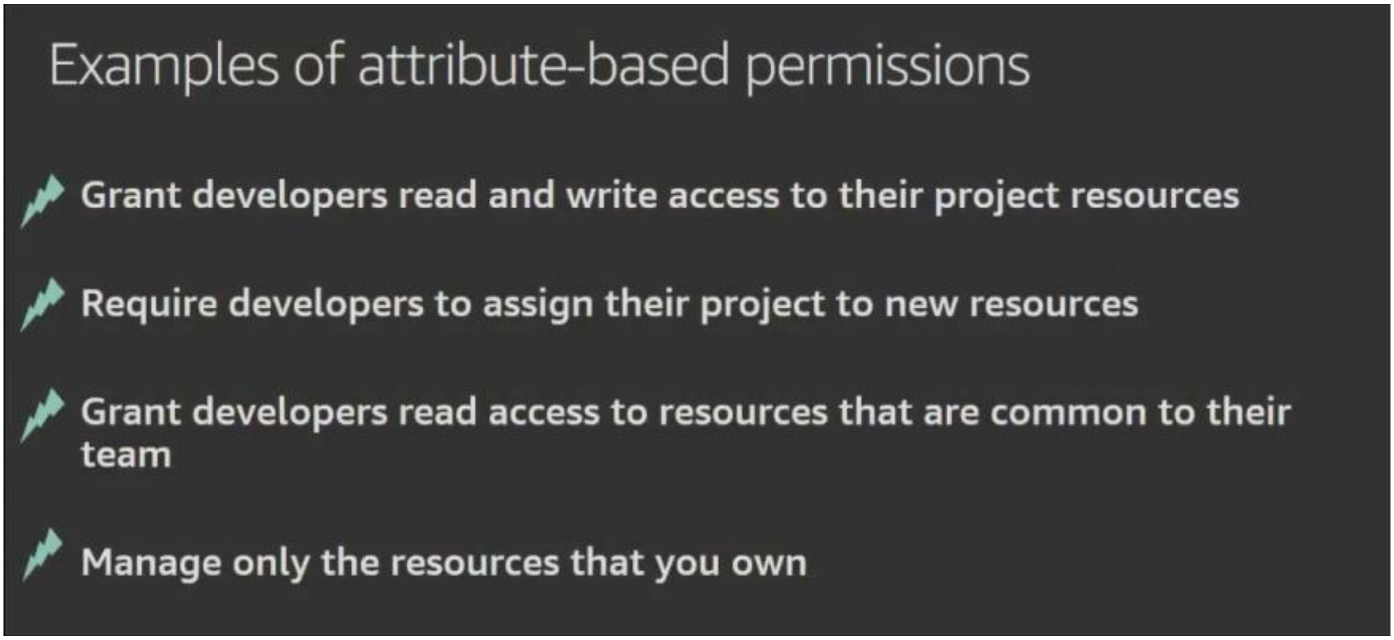


Admin allows creation and management of roles

3. Pass roles they created

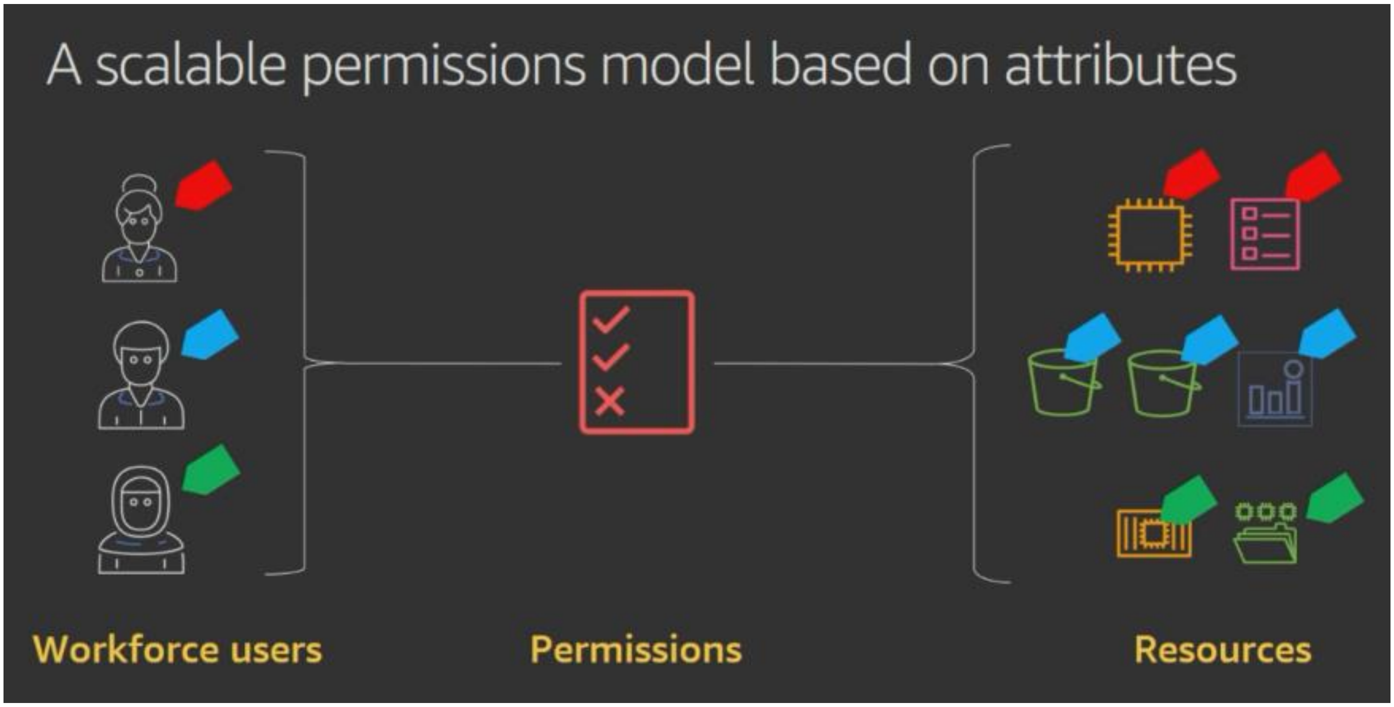
```
 {
       "Effect": "Allow",
       "Action": [
              "iam:passrole"],
       "Resource": "arn:aws:iam::432807222178:role/${aws:PrincipalTag/project}-*"
}
```

Rely on attributes for fine-grained permission at scale



Examples of attribute-based permissions

- Grant developers read and write access to their project resources
- Require developers to assign their project to new resources
- Grant developers read access to resources that are common to their team
- Manage only the resources that you own



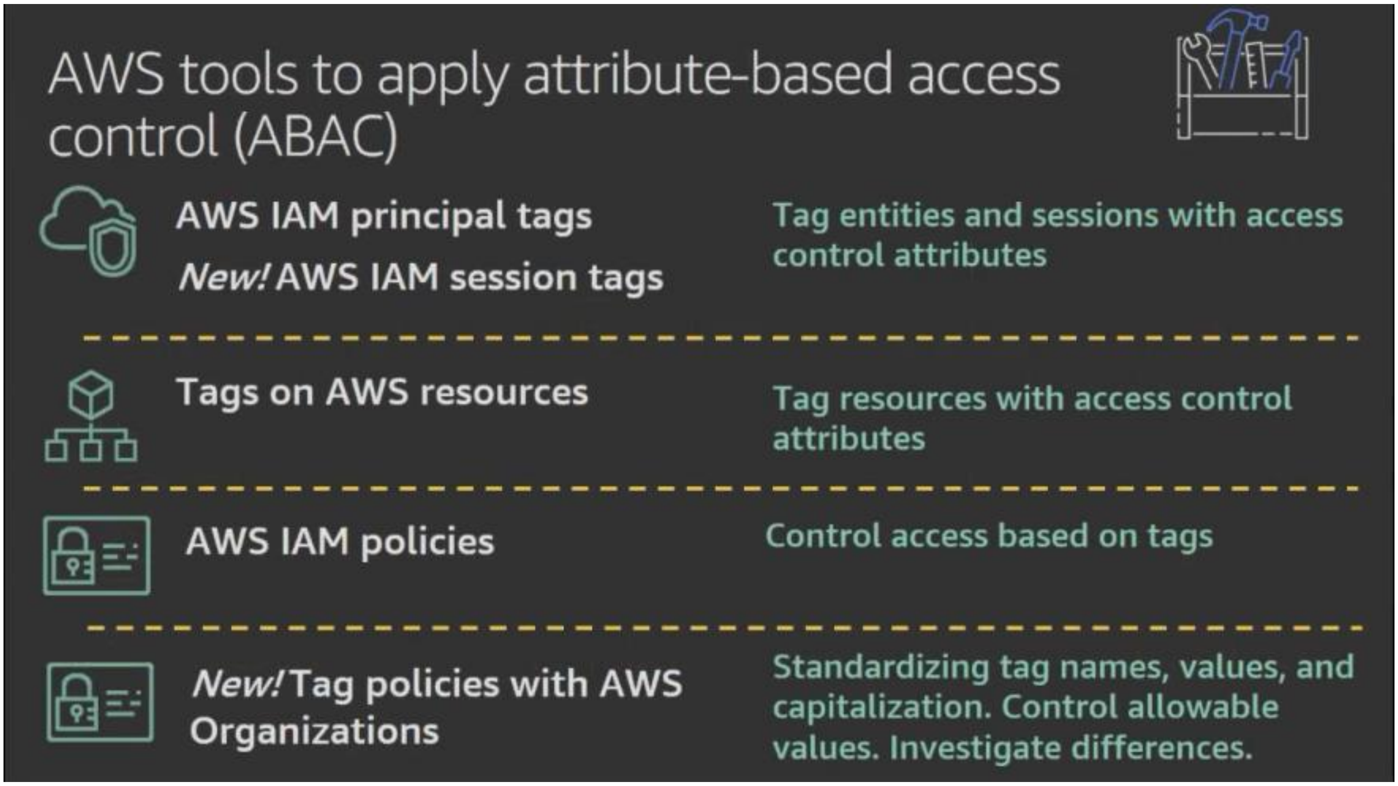
A scalable permissions model based on attributes

Workforce users

Permissions

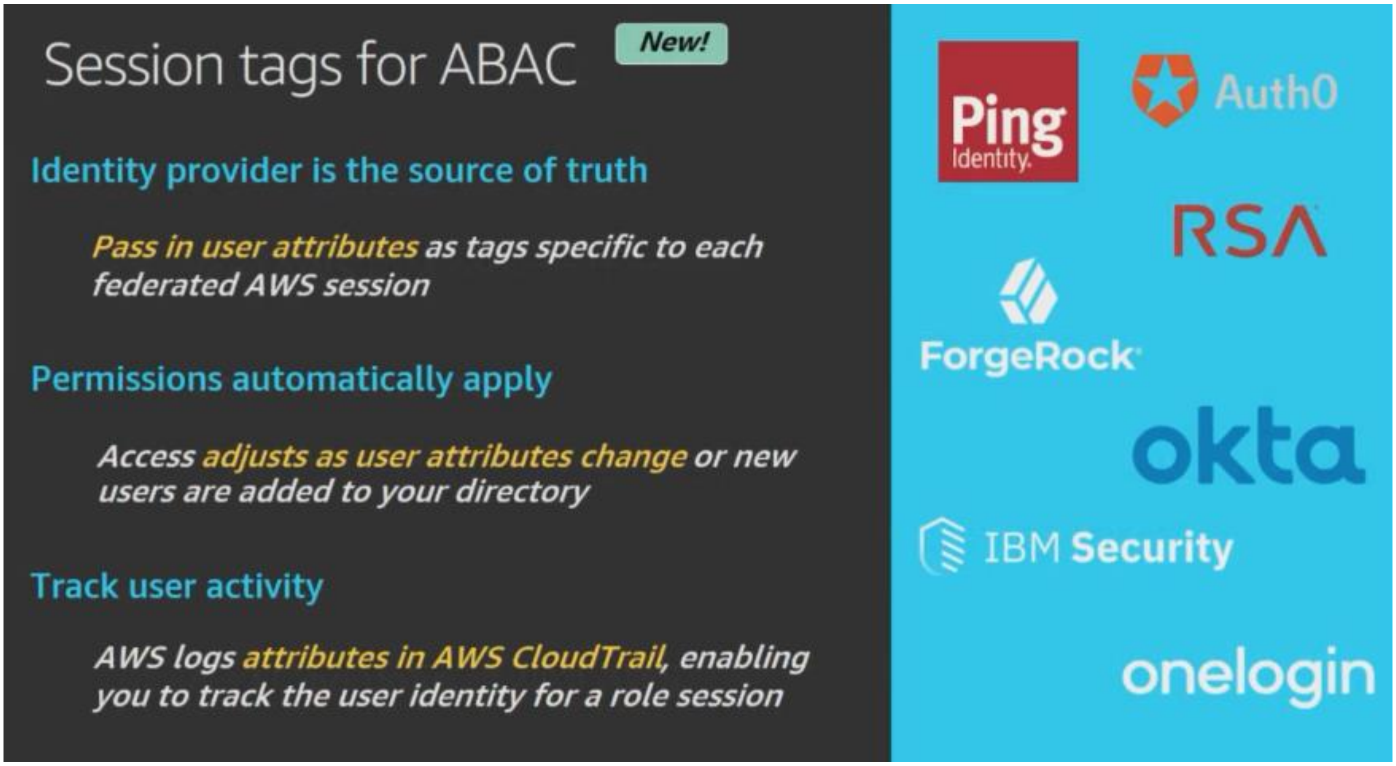
Resources

The policy says allow if the attributes match, permissions will automatically apply based on the attributes on the users and the resources



AWS tools to apply attribute-based access control (ABAC)

AWS IAM principal tags ***New!*** **AWS IAM session tags**	Tag entities and sessions with access control attributes
Tags on AWS resources	Tag resources with access control attributes
AWS IAM policies	Control access based on tags
New! **Tag policies with AWS Organizations**	Standardizing tag names, values, and capitalization. Control allowable values. Investigate differences.



Session tags for ABAC

New!

Identity provider is the source of truth

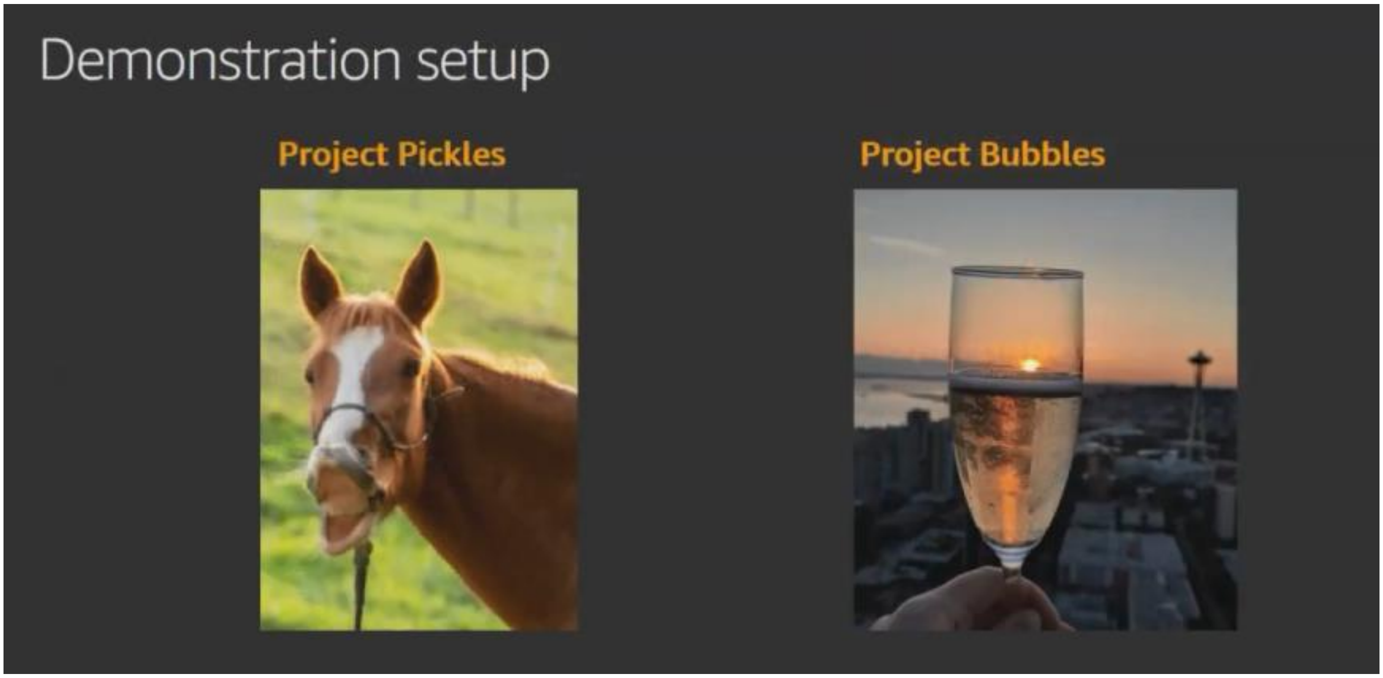
Pass in user attributes as tags specific to each federated AWS session

Permissions automatically apply

Access adjusts as user attributes change or new users are added to your directory

Track user activity

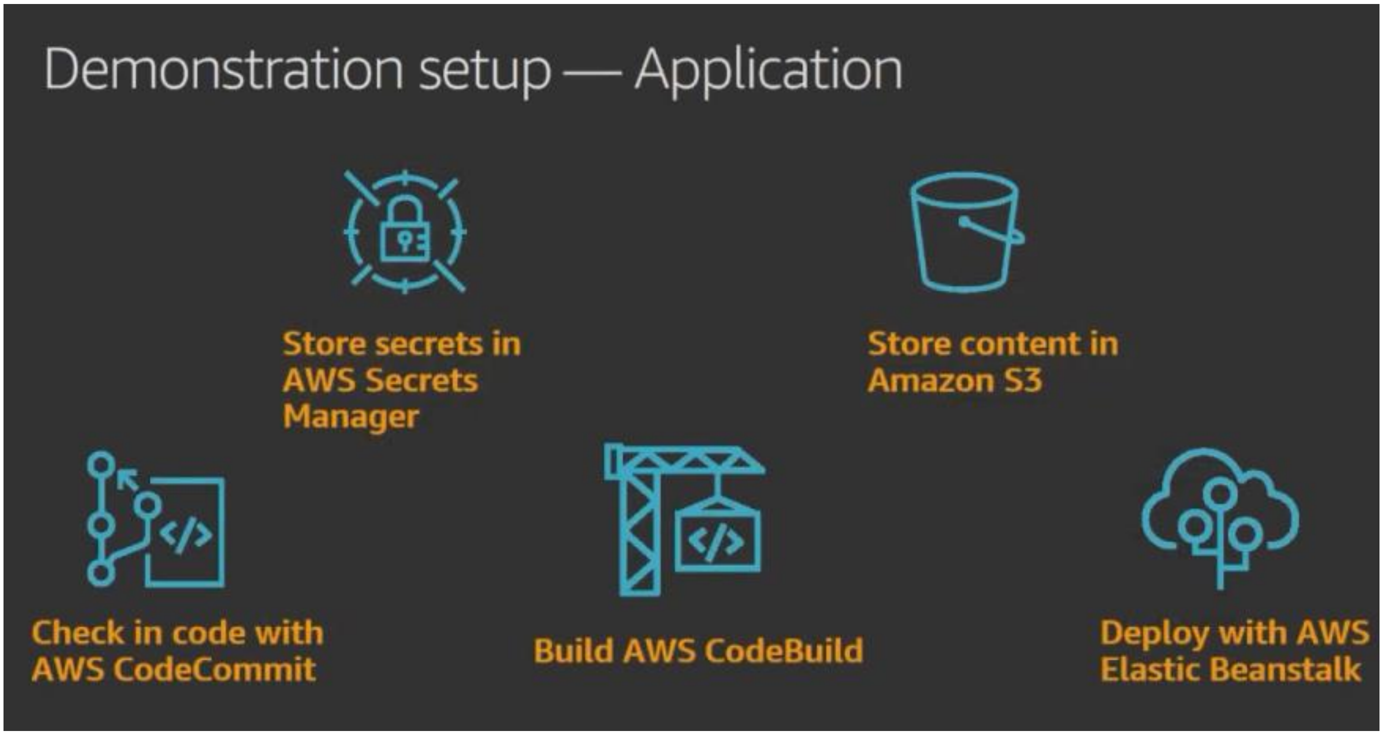
AWS logs attributes in AWS CloudTrail, enabling you to track the user identity for a role session



Demonstration setup

Project Pickles

Project Bubbles



Demonstration setup — Application

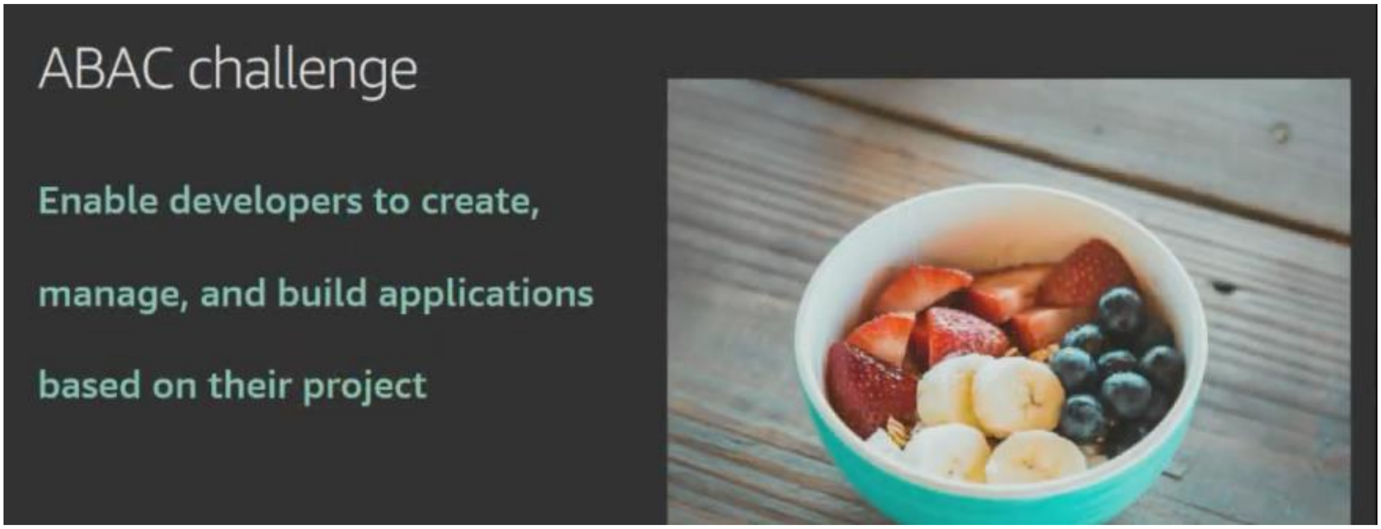
Store secrets in AWS Secrets Manager

Store content in Amazon S3

Check in code with AWS CodeCommit

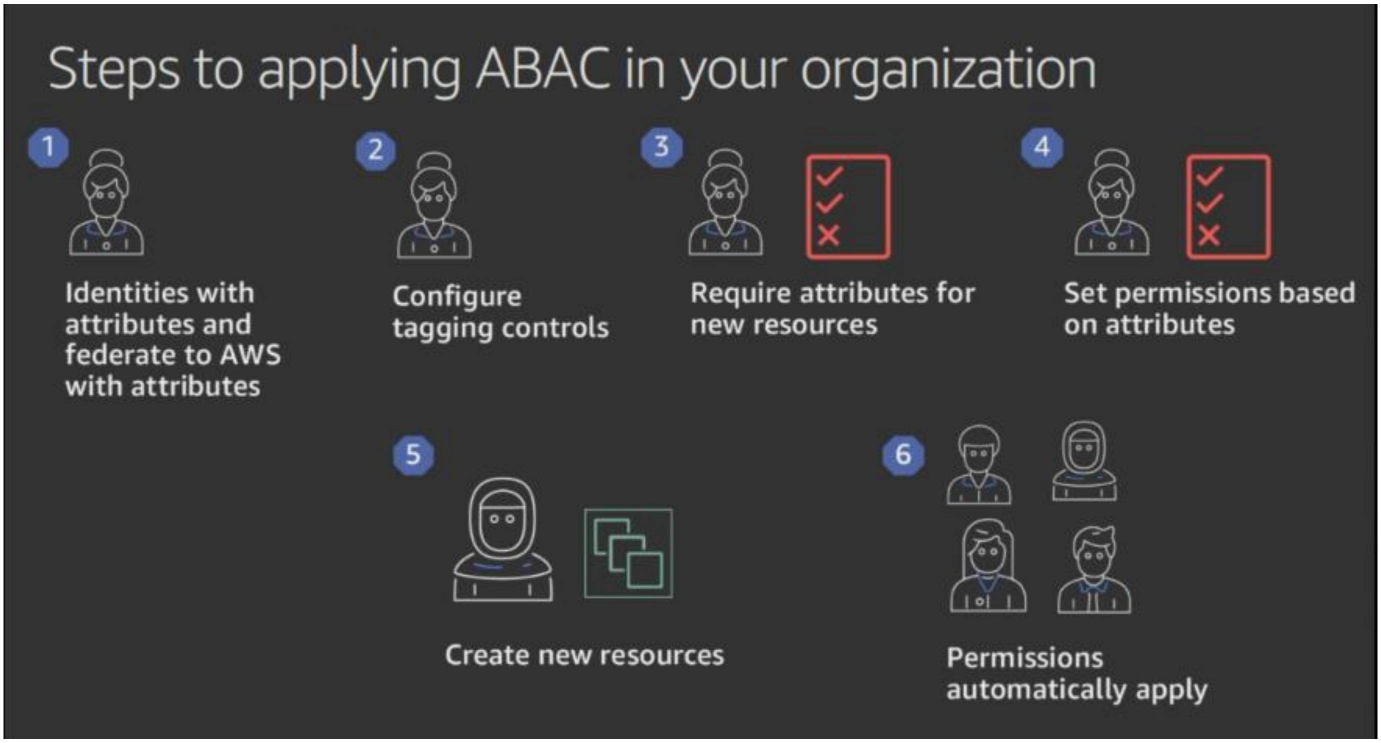
Build AWS CodeBuild

Deploy with AWS Elastic Beanstalk



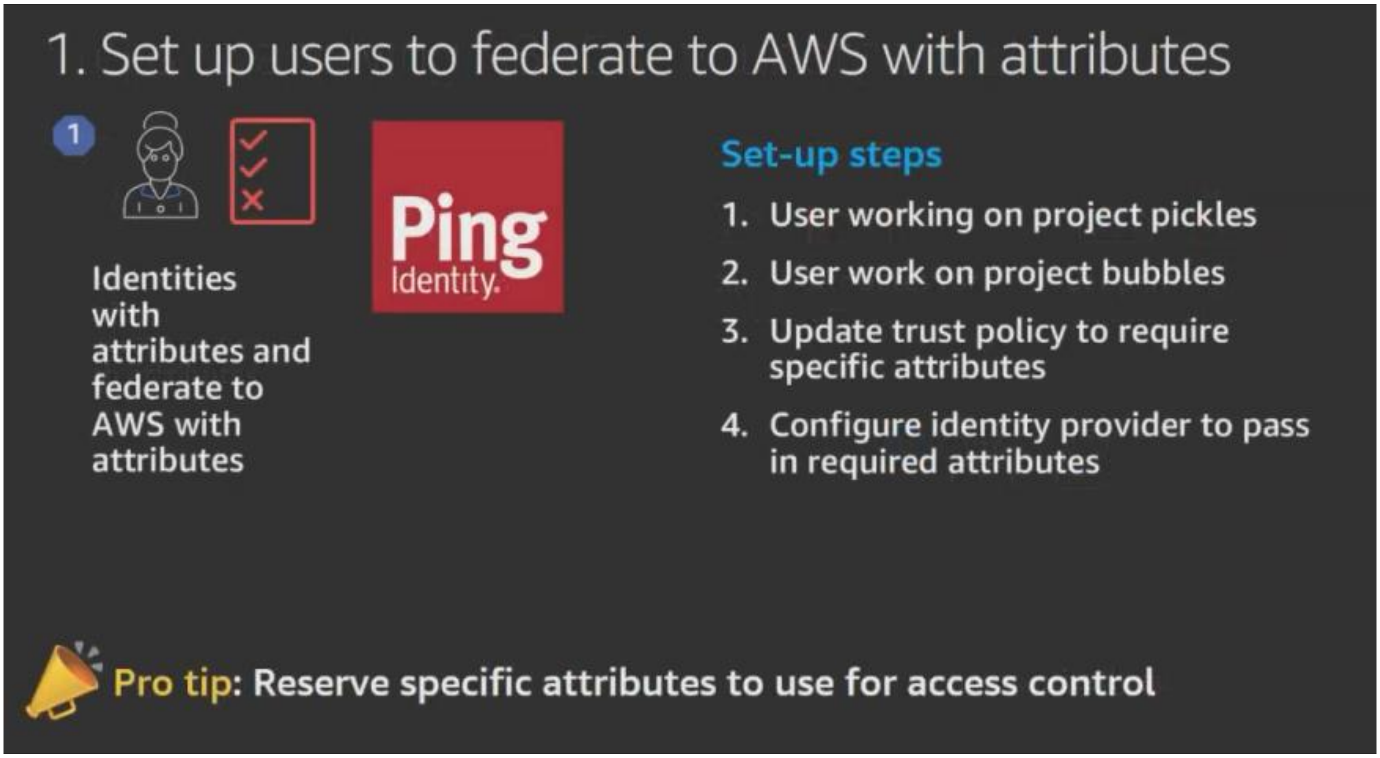
ABAC challenge

Enable developers to create, manage, and build applications based on their project



Steps to applying ABAC in your organization

1. Identities with attributes and federate to AWS with attributes
2. Configure tagging controls
3. Require attributes for new resources
4. Set permissions based on attributes
5. Create new resources
6. Permissions automatically apply



1. Set up users to federate to AWS with attributes

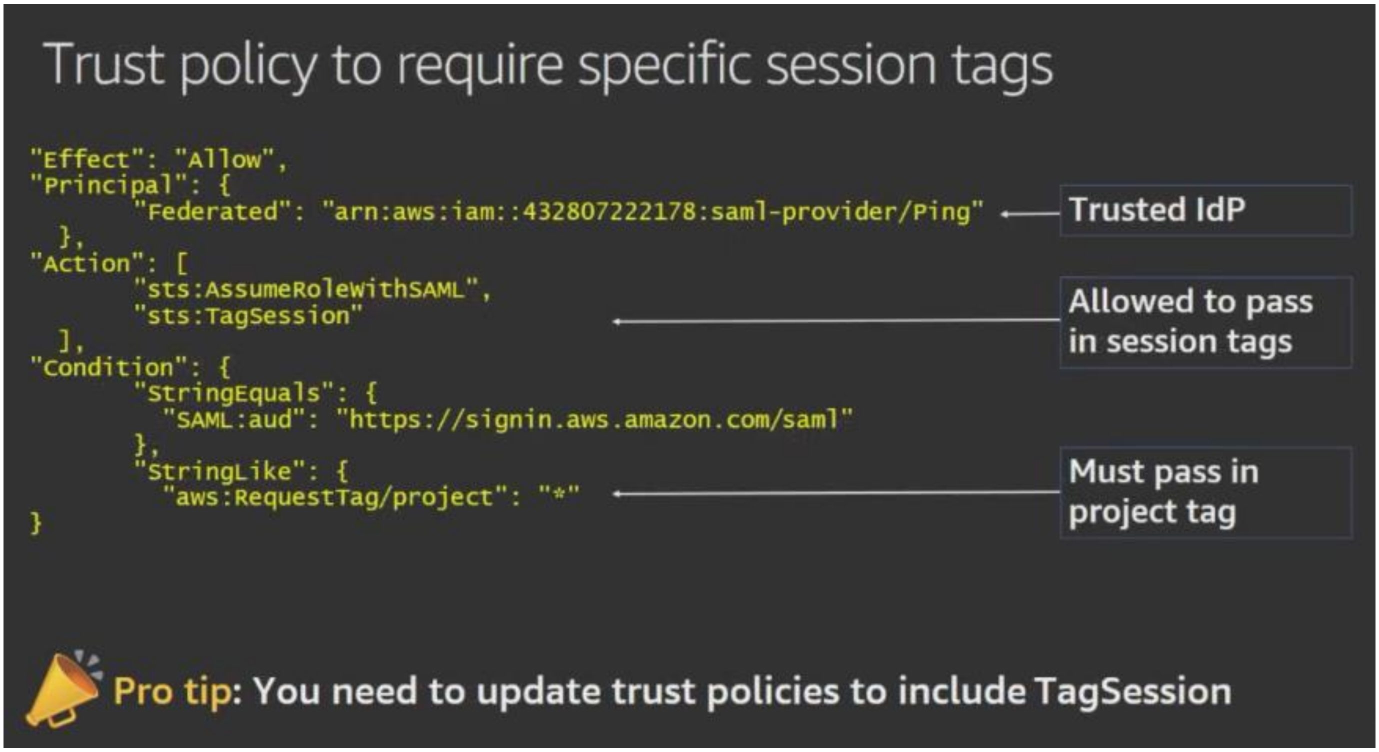
1
Ping Identity.

Identities with attributes and federate to AWS with attributes

Set-up steps

1. User working on project pickles
2. User work on project bubbles
3. Update trust policy to require specific attributes
4. Configure identity provider to pass in required attributes

Pro tip: Reserve specific attributes to use for access control



Trust policy to require specific session tags

```
"Effect": "Allow",
"Principal": {
       "Federated": "arn:aws:iam::432807222178:saml-provider/Ping"
  },
"Action": [
       "sts:AssumeRoleWithSAML",
       "sts:TagSession"
  ],
"Condition": {
       "StringEquals": {
         "SAML:aud": "https://signin.aws.amazon.com/saml"
       },
       "StringLike": {
         "aws:RequestTag/project": "*"
}
```

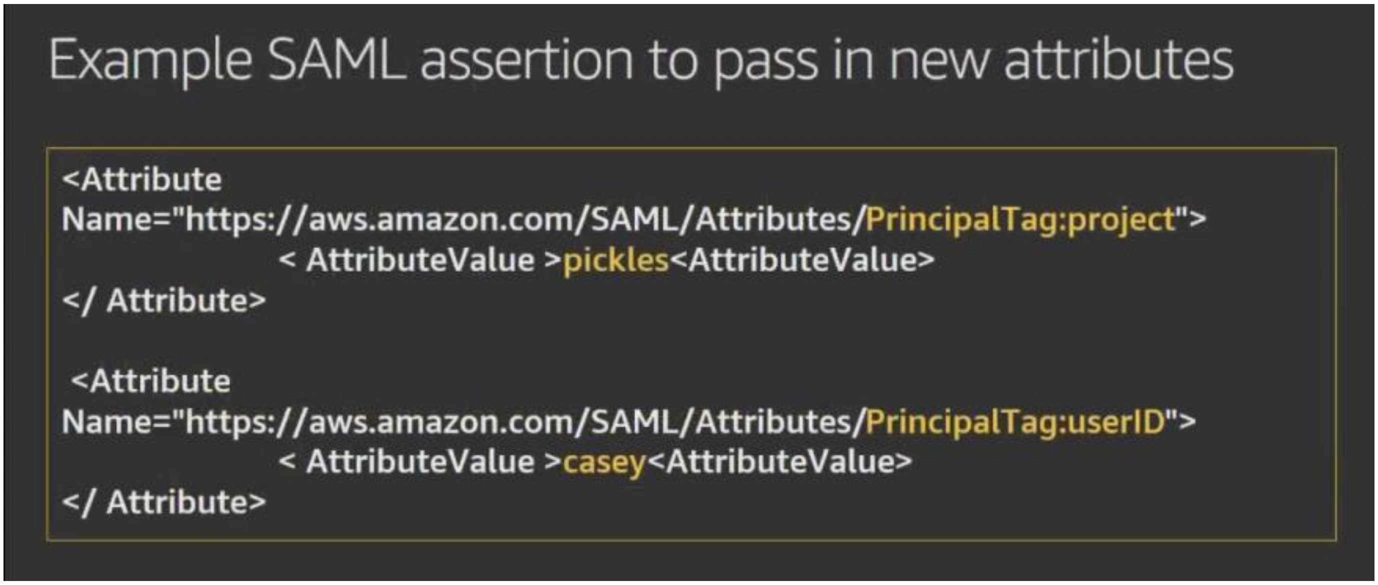
Trusted IdP

Allowed to pass in session tags

Must pass in project tag

Pro tip: You need to update trust policies to include TagSession

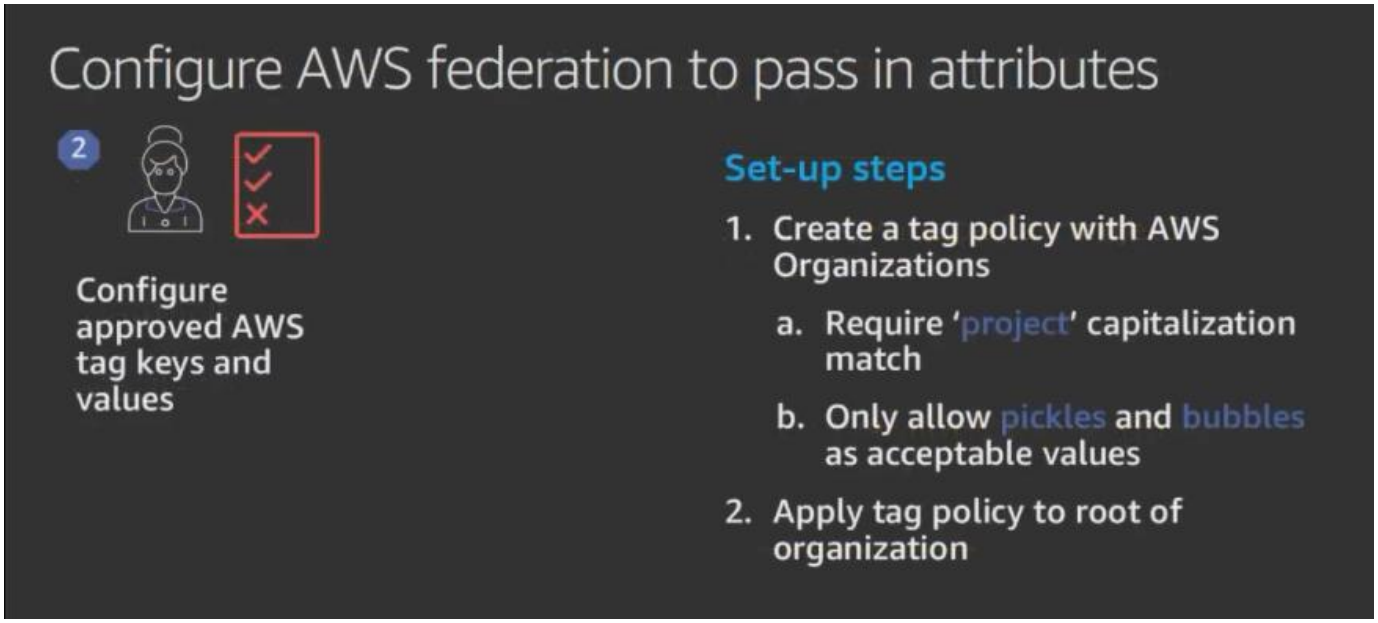
A trust policy is a resource policy attached to a role, here the principal is Ping that we trust to assume the SAML roles



Example SAML assertion to pass in new attributes

```
<Attribute
Name="https://aws.amazon.com/SAML/Attributes/PrincipalTag:project">
              < AttributeValue >pickles<AttributeValue>
</ Attribute>

 <Attribute
Name="https://aws.amazon.com/SAML/Attributes/PrincipalTag:userID">
              < AttributeValue >casey<AttributeValue>
</ Attribute>
```



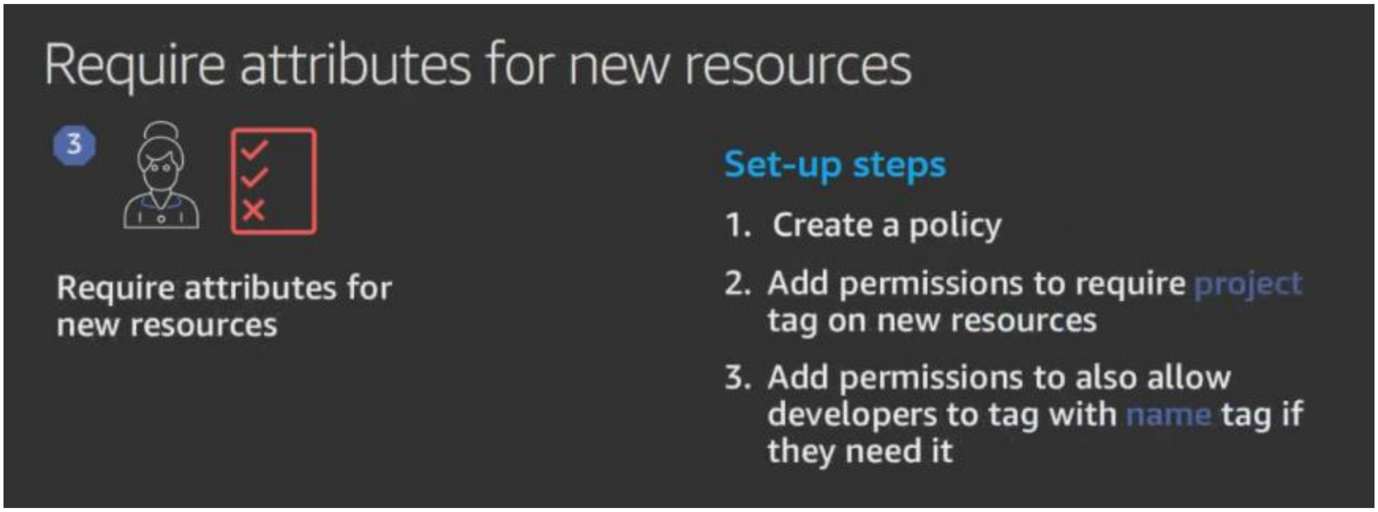
Configure AWS federation to pass in attributes

2

Configure approved AWS tag keys and values

Set-up steps

1. Create a tag policy with AWS Organizations
   a. Require 'project' capitalization match
   b. Only allow pickles and bubbles as acceptable values
2. Apply tag policy to root of organization



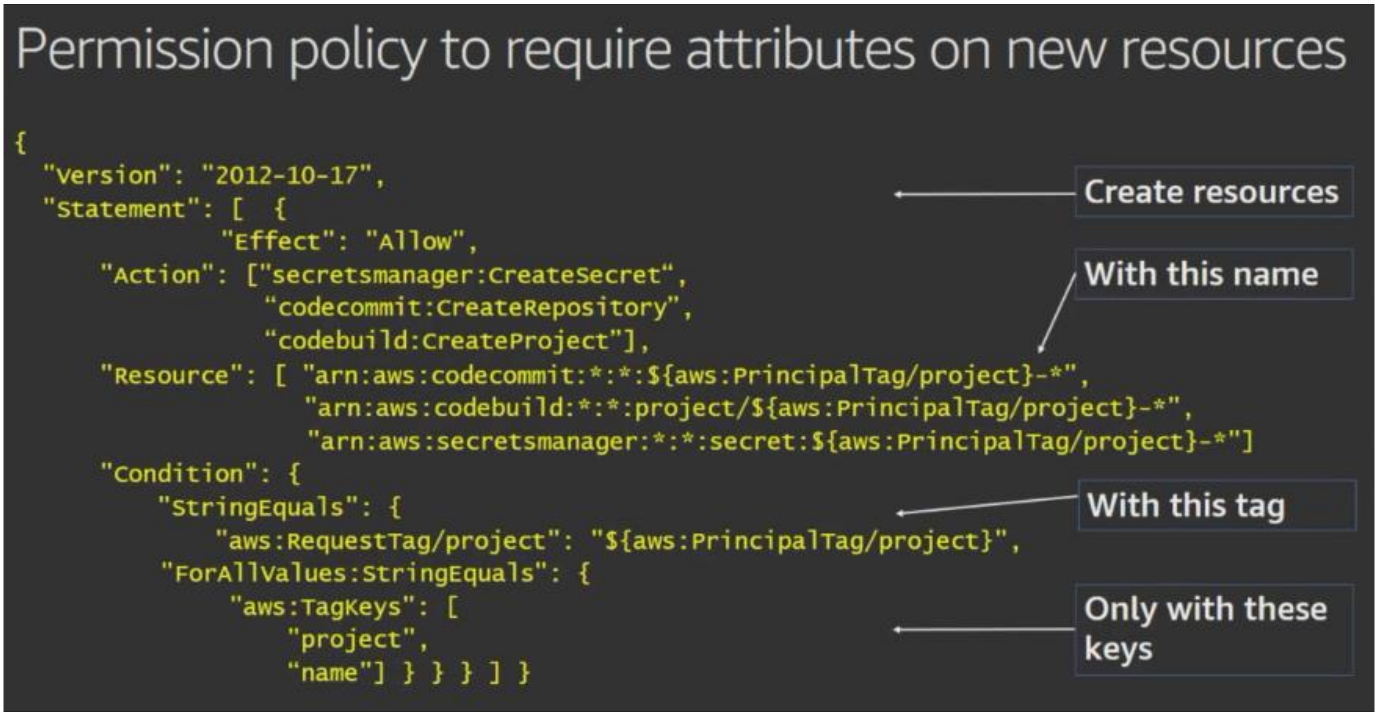
Require attributes for new resources

3

Require attributes for new resources

Set-up steps

1. Create a policy
2. Add permissions to require project tag on new resources
3. Add permissions to also allow developers to tag with name tag if they need it



Permission policy to require attributes on new resources

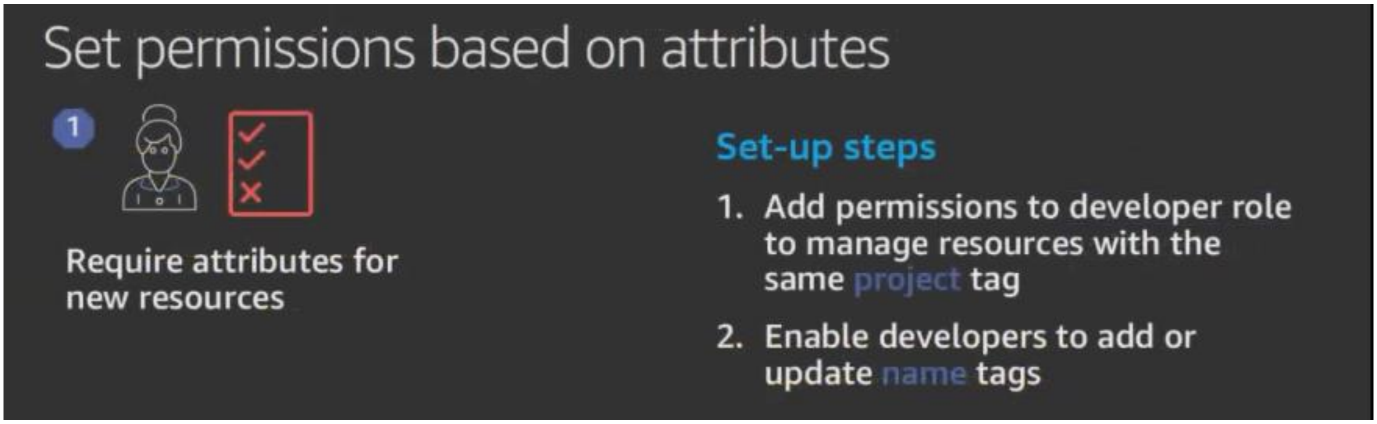
```
{
  "Version": "2012-10-17",
  "Statement": [  {
              "Effect": "Allow",
      "Action": ["secretsmanager:CreateSecret“,
                 “codecommit:CreateRepository”,
                 “codebuild:CreateProject”],
      "Resource": [ "arn:aws:codecommit:*:*:${aws:PrincipalTag/project}-*",
                    "arn:aws:codebuild:*:*:project/${aws:PrincipalTag/project}-*",
                    "arn:aws:secretsmanager:*:*:secret:${aws:PrincipalTag/project}-*"]
      "Condition": {
          "StringEquals": {
              "aws:RequestTag/project": "${aws:PrincipalTag/project}",
          "ForAllValues:StringEquals": {
              "aws:TagKeys": [
                  "project",
                  “name"] } } } ] }
```

Create resources

With this name

With this tag

Only with these keys



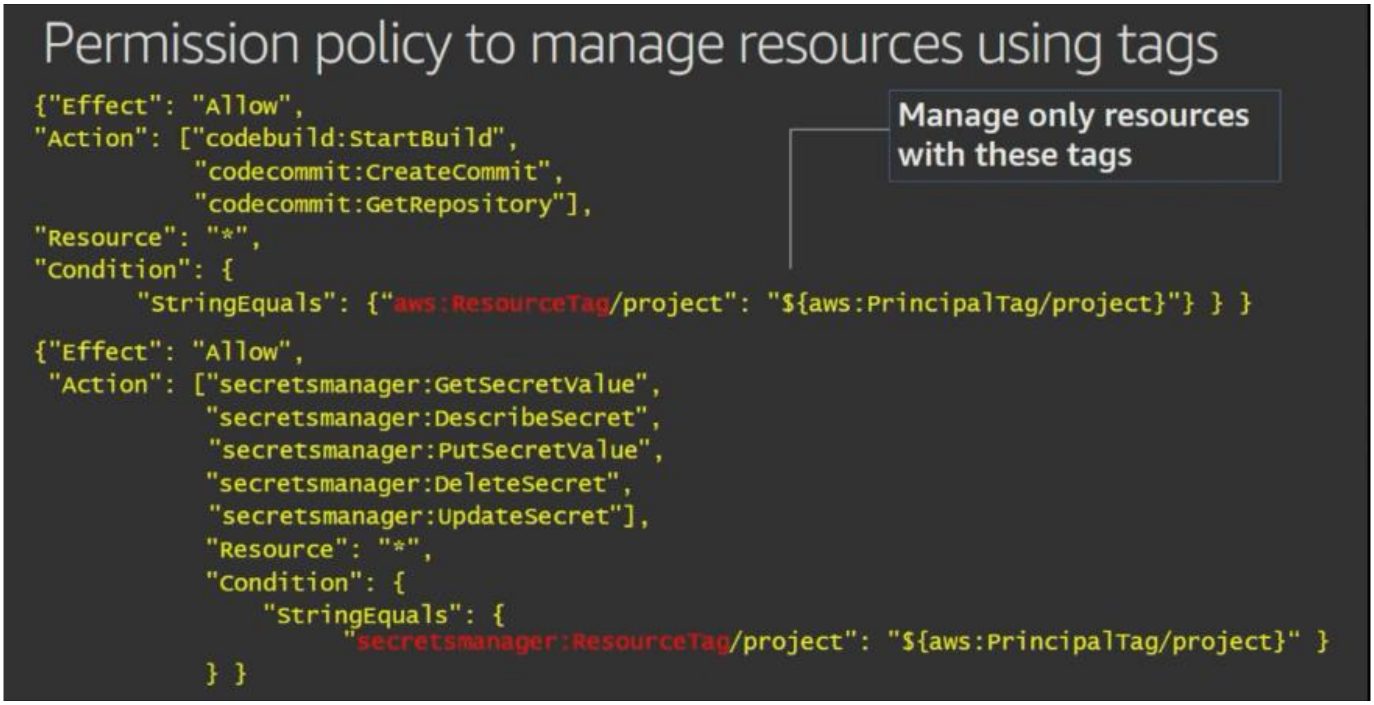
Set permissions based on attributes

1

Require attributes for new resources

Set-up steps

1. Add permissions to developer role to manage resources with the same project tag
2. Enable developers to add or update name tags

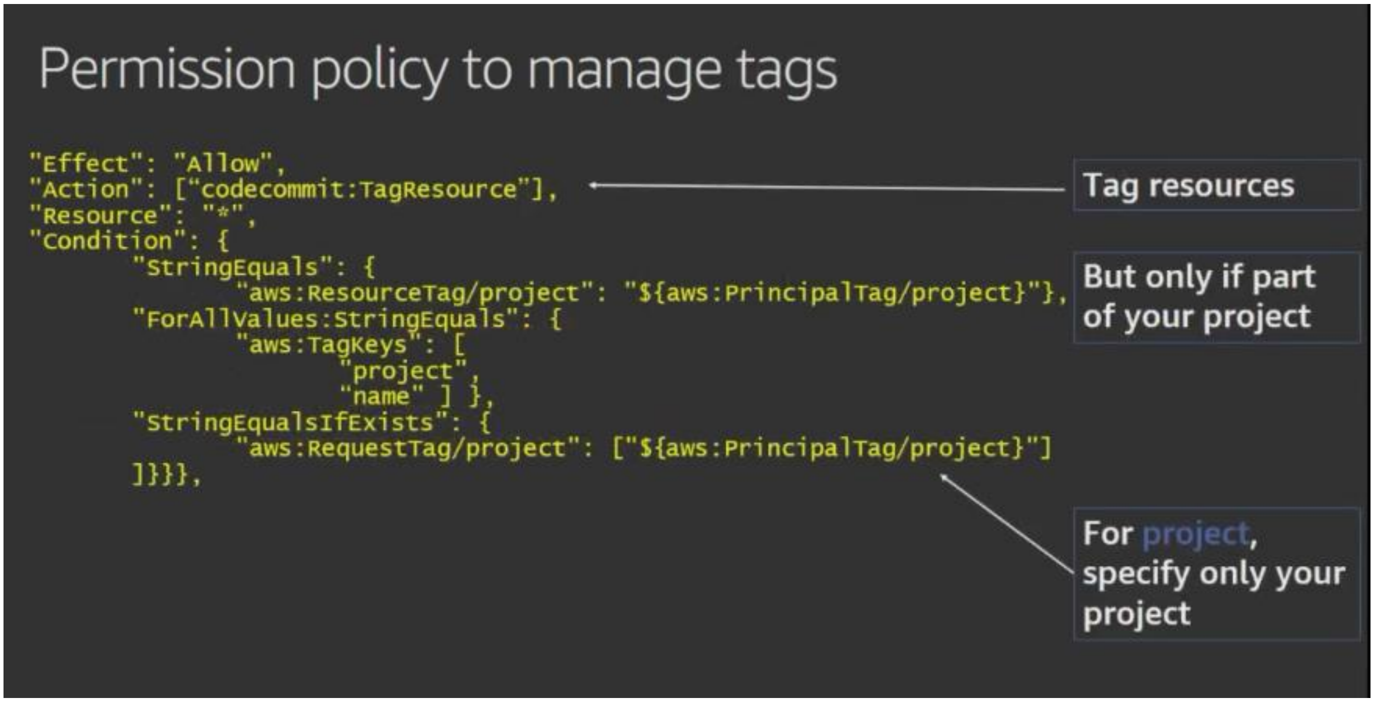


Permission policy to manage resources using tags

```
{"Effect": "Allow",
"Action": ["codebuild:StartBuild",
           "codecommit:CreateCommit",
           "codecommit:GetRepository"],
"Resource": "*",
"Condition": {
       "StringEquals": {“aws:ResourceTag/project": "${aws:PrincipalTag/project}"} } }

{"Effect": "Allow",
 "Action": ["secretsmanager:GetSecretValue",
            "secretsmanager:DescribeSecret",
            "secretsmanager:PutSecretValue",
            "secretsmanager:DeleteSecret",
            "secretsmanager:UpdateSecret"],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                      "secretsmanager:ResourceTag/project": "${aws:PrincipalTag/project}" }
            } }
```

Manage only resources with these tags



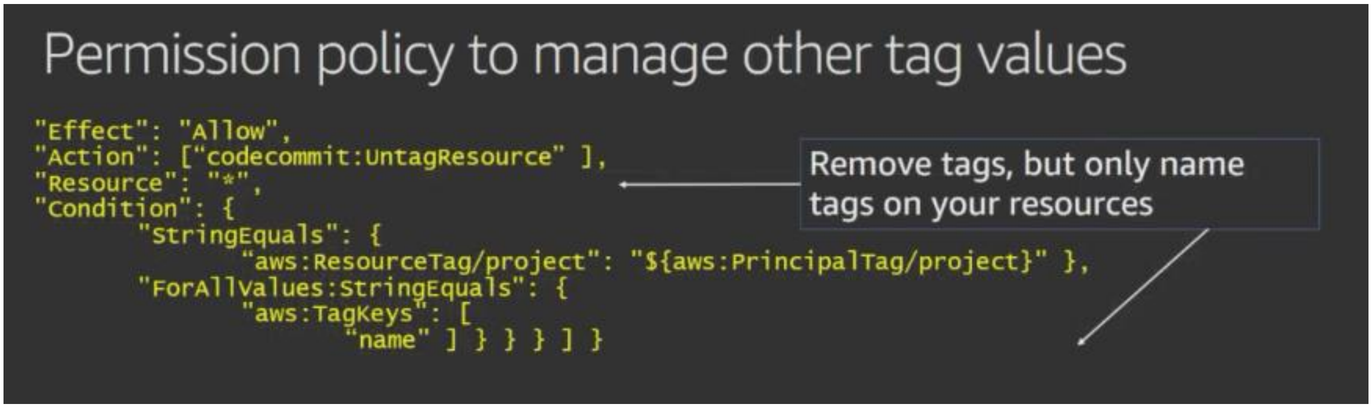
Permission policy to manage tags

```
"Effect": "Allow",
"Action": ["codecommit:TagResource"],
"Resource": "*",
"Condition": {
        "StringEquals": {
                "aws:ResourceTag/project": "${aws:PrincipalTag/project}"},
        "ForAllValues:StringEquals": {
                "aws:TagKeys": [
                        "project",
                        "name" ] },
        "StringEqualsIfExists": {
                "aws:RequestTag/project": ["${aws:PrincipalTag/project}"]
        ]}}},
```

Tag resources

But only if part of your project

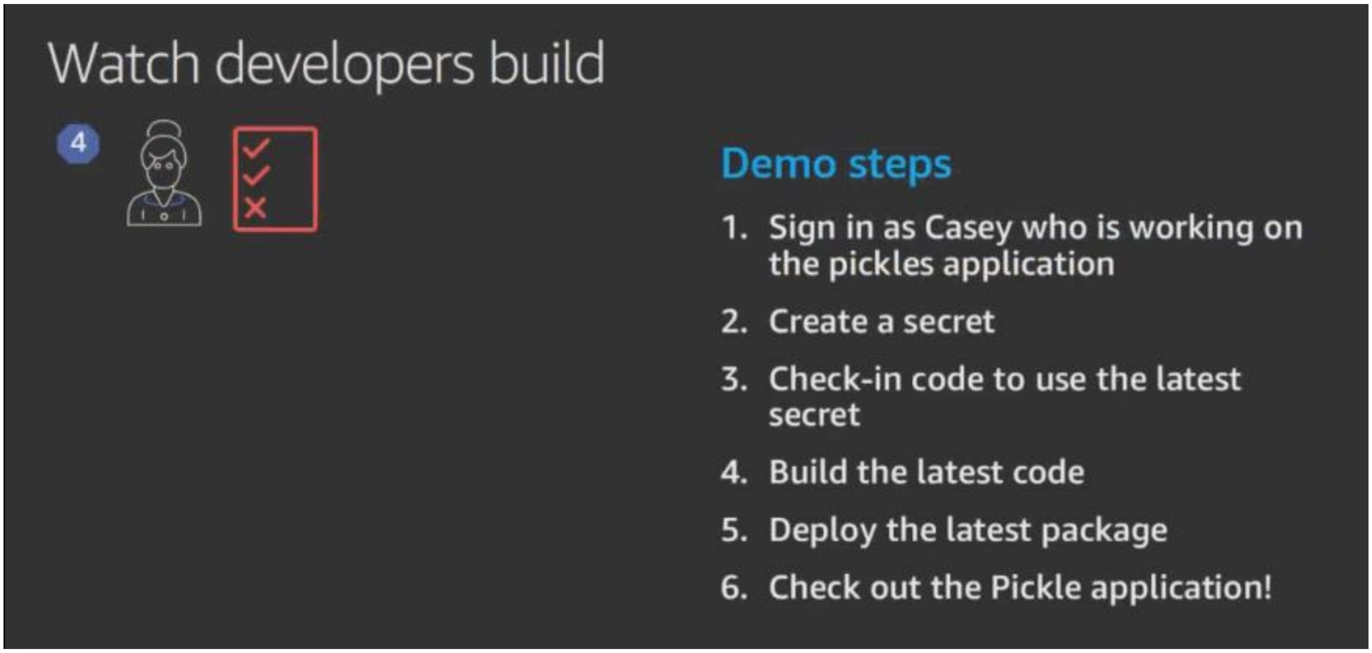
For project, specify only your project



Permission policy to manage other tag values

```
"Effect": "Allow",
"Action": ["codecommit:UntagResource" ],
"Resource": "*",
"Condition": {
        "StringEquals": {
                "aws:ResourceTag/project": "${aws:PrincipalTag/project}" },
        "ForAllValues:StringEquals": {
                "aws:TagKeys": [
                        "name" ] } } } ] }
```

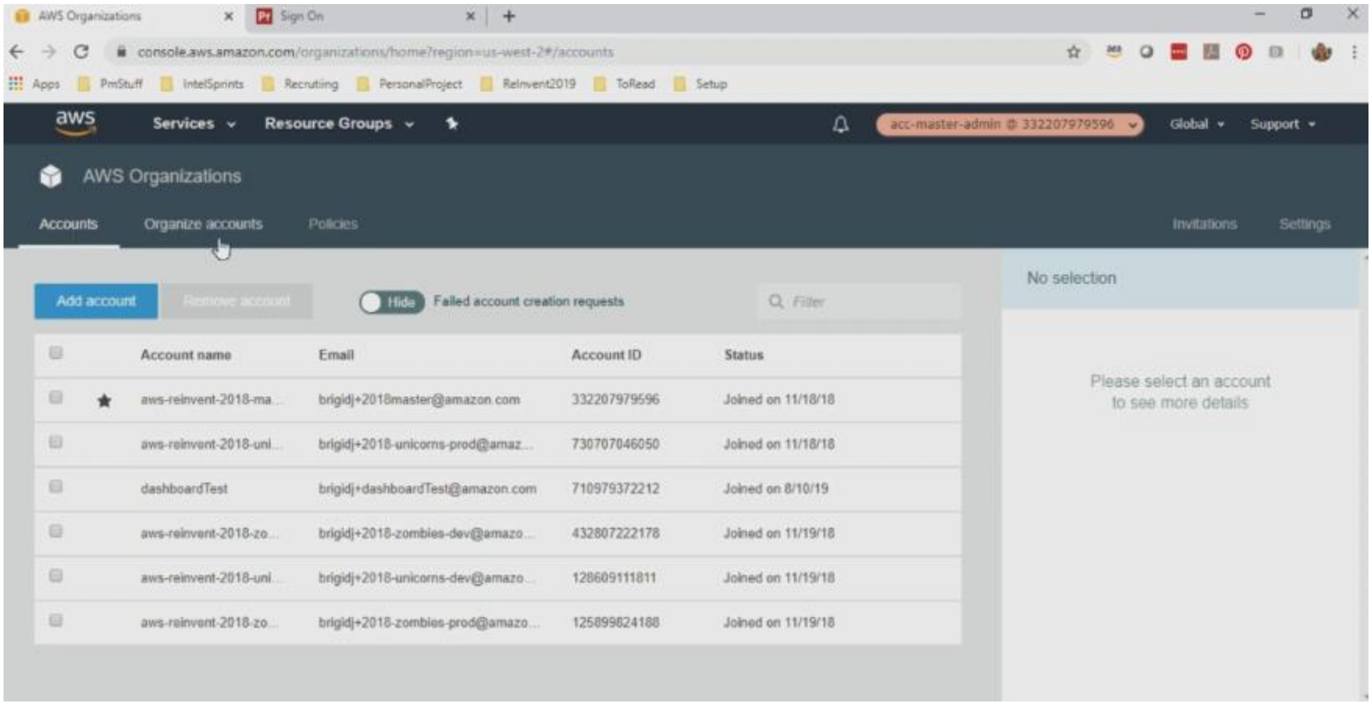
Remove tags, but only name tags on your resources



Watch developers build

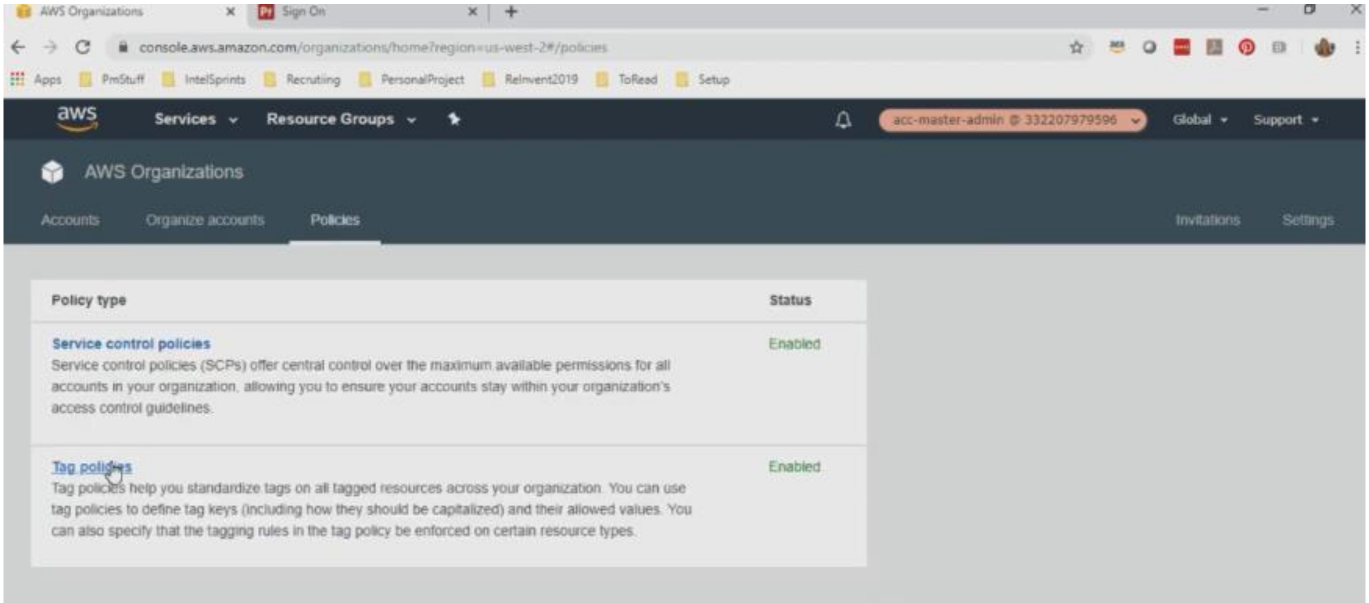
Demo steps

1. Sign in as Casey who is working on the pickles application
2. Create a secret
3. Check-in code to use the latest secret
4. Build the latest code
5. Deploy the latest package
6. Check out the Pickle application!

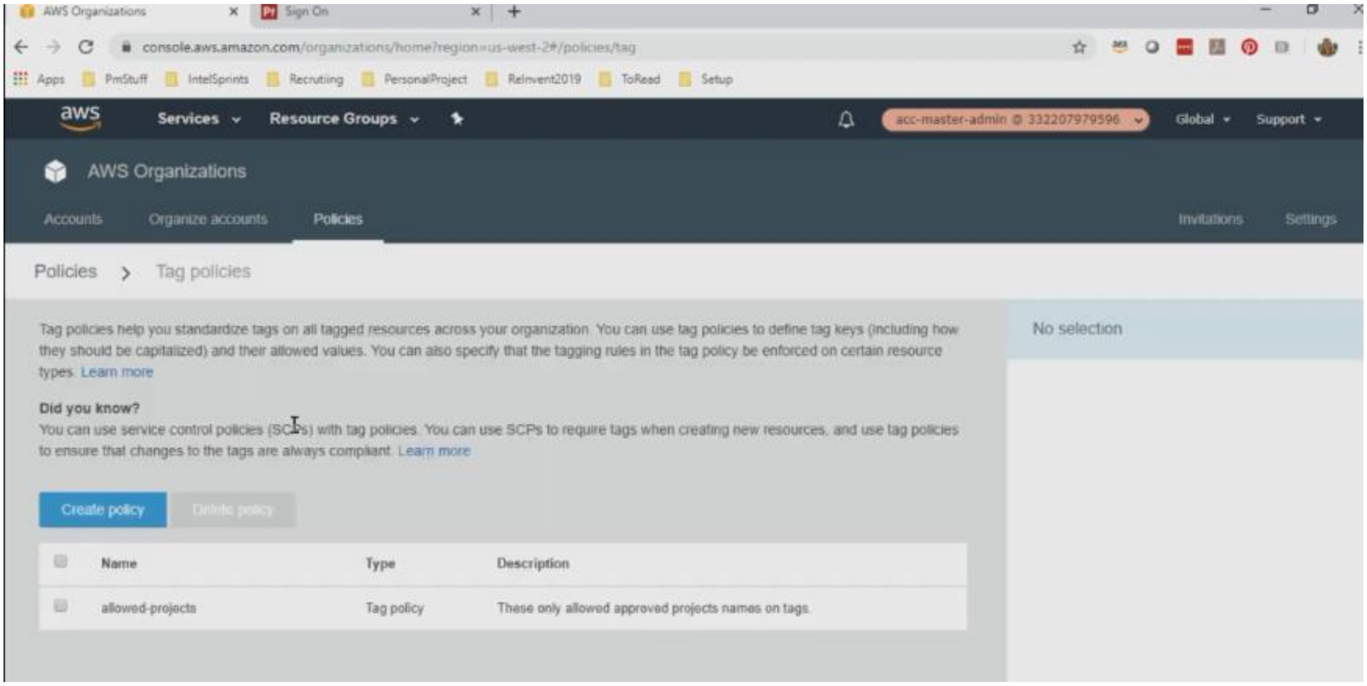


		Account name	Email	Account ID	Status
□	★	aws-reinvent-2018-ma...	brigidj+2018master@amazon.com	332207979596	Joined on 11/18/18
□		aws-reinvent-2018-uni...	brigidj+2018-unicorns-prod@amaz...	730707046050	Joined on 11/18/18
□		dashboardTest	brigidj+dashboardTest@amazon.com	710979372212	Joined on 8/10/19
□		aws-reinvent-2018-zo...	brigidj+2018-zombies-dev@amazo...	432807222178	Joined on 11/19/18
□		aws-reinvent-2018-uni...	brigidj+2018-unicorns-dev@amazo...	128609111811	Joined on 11/19/18
□		aws-reinvent-2018-zo...	brigidj+2018-zombies-prod@amazo...	125899824188	Joined on 11/19/18

Please select an account to see more details



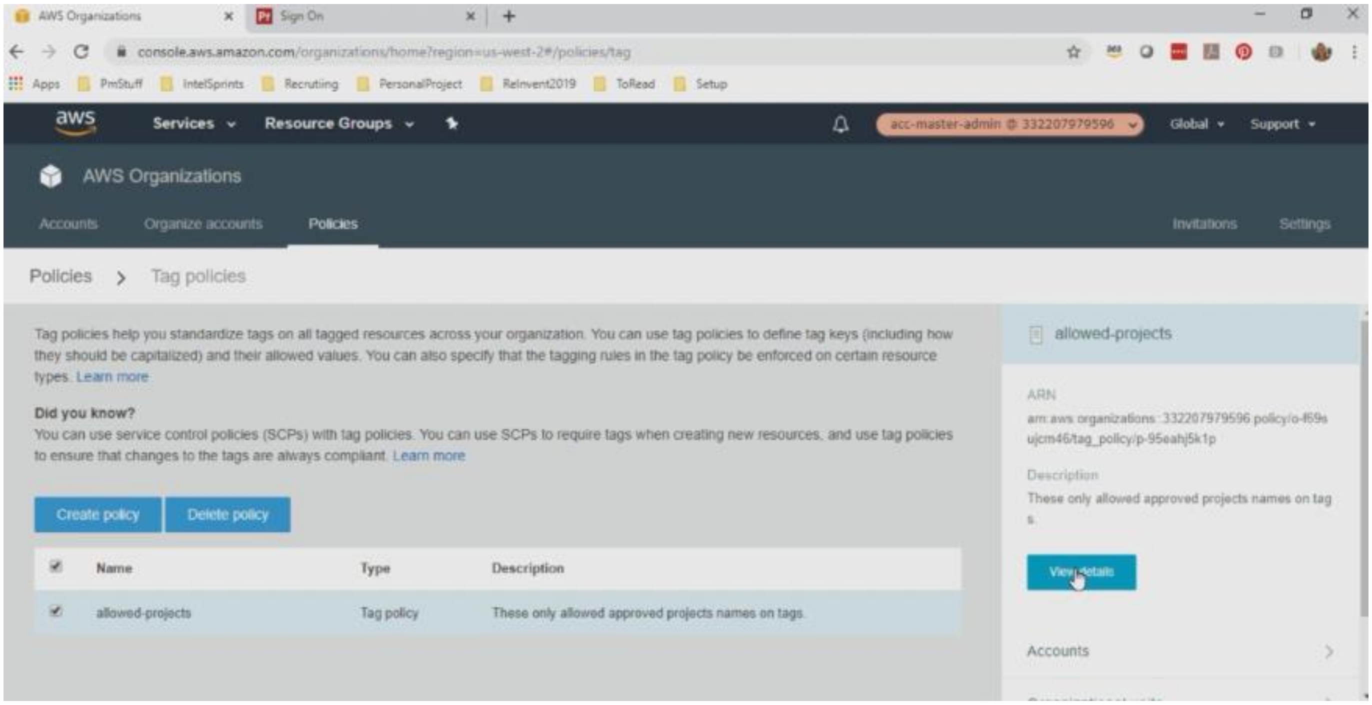
Policy type	Status
Service control policies Service control policies (SCPs) offer central control over the maximum available permissions for all accounts in your organization, allowing you to ensure your accounts stay within your organization's access control guidelines.	Enabled
Tag policies Tag policies help you standardize tags on all tagged resources across your organization. You can use tag policies to define tag keys (including how they should be capitalized) and their allowed values. You can also specify that the tagging rules in the tag policy be enforced on certain resource types.	Enabled



Tag policies help you standardize tags on all tagged resources across your organization. You can use tag policies to define tag keys (including how they should be capitalized) and their allowed values. You can also specify that the tagging rules in the tag policy be enforced on certain resource types. Learn more

Did you know?
You can use service control policies (SCPs) with tag policies. You can use SCPs to require tags when creating new resources, and use tag policies to ensure that changes to the tags are always compliant. Learn more

□	Name	Type	Description
□	allowed-projects	Tag policy	These only allowed approved projects names on tags.



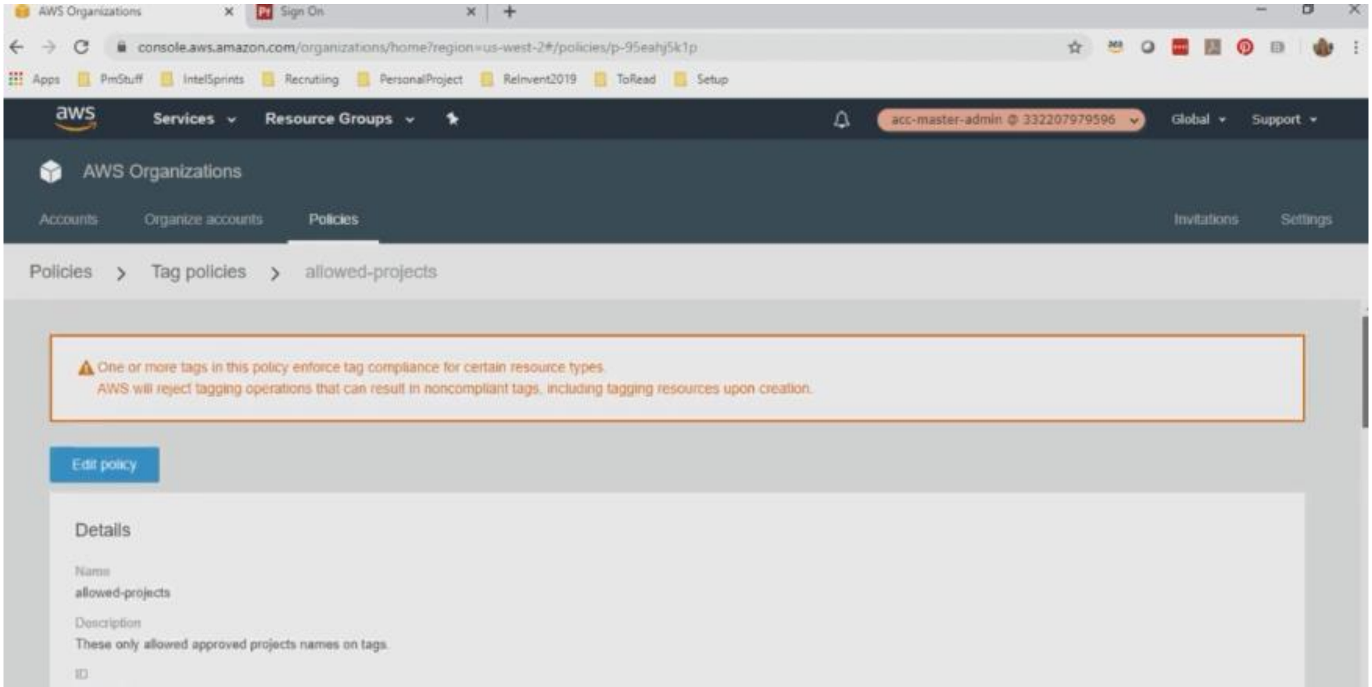
Tag policies help you standardize tags on all tagged resources across your organization. You can use tag policies to define tag keys (including how they should be capitalized) and their allowed values. You can also specify that the tagging rules in the tag policy be enforced on certain resource types. Learn more

Did you know?
You can use service control policies (SCPs) with tag policies. You can use SCPs to require tags when creating new resources, and use tag policies to ensure that changes to the tags are always compliant. Learn more

☑	Name	Type	Description
☑	allowed-projects	Tag policy	These only allowed approved projects names on tags.

ARN
arn:aws:organizations::332207979596:policy/o-f69sujcm46/tag_policy/p-95eahj5k1p

Description
These only allowed approved projects names on tags.



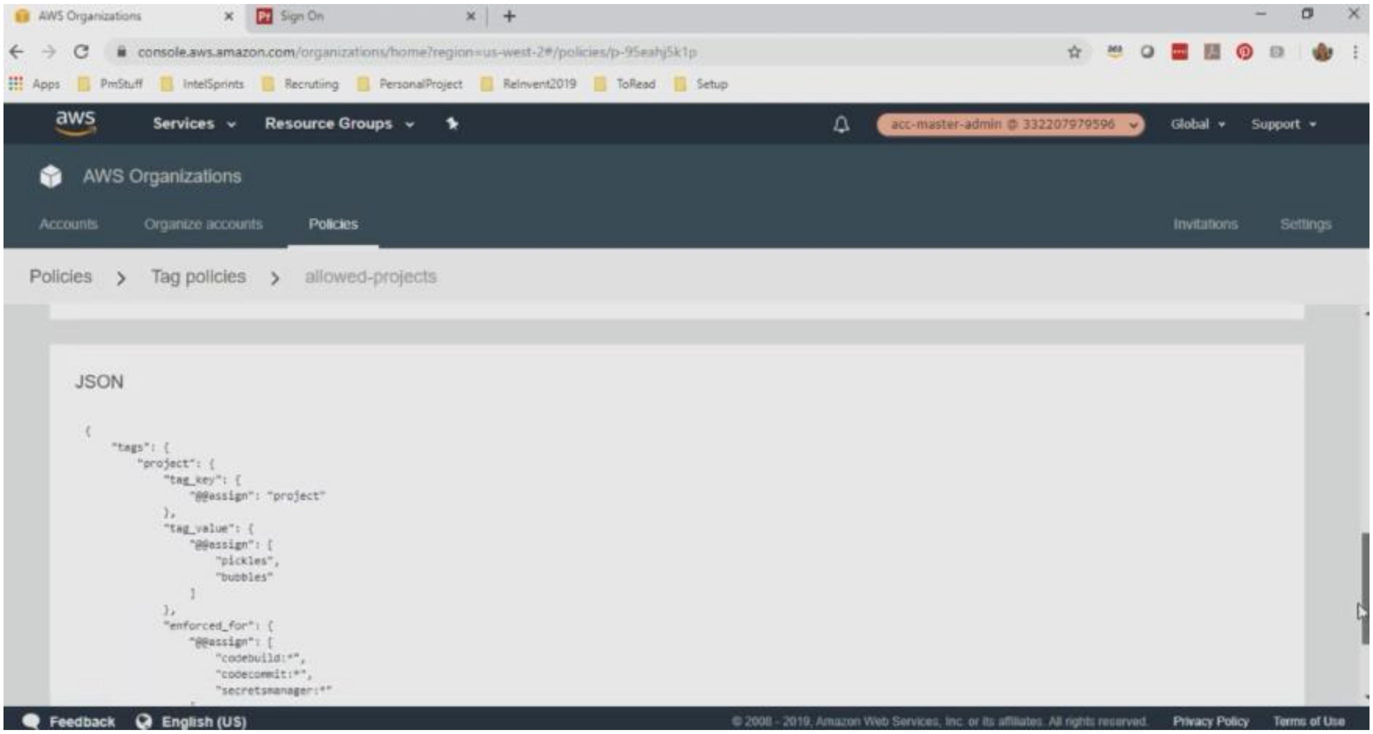
One or more tags in this policy enforce tag compliance for certain resource types.
AWS will reject tagging operations that can result in noncompliant tags, including tagging resources upon creation.

Details

Name
allowed-projects

Description
These only allowed approved projects names on tags.

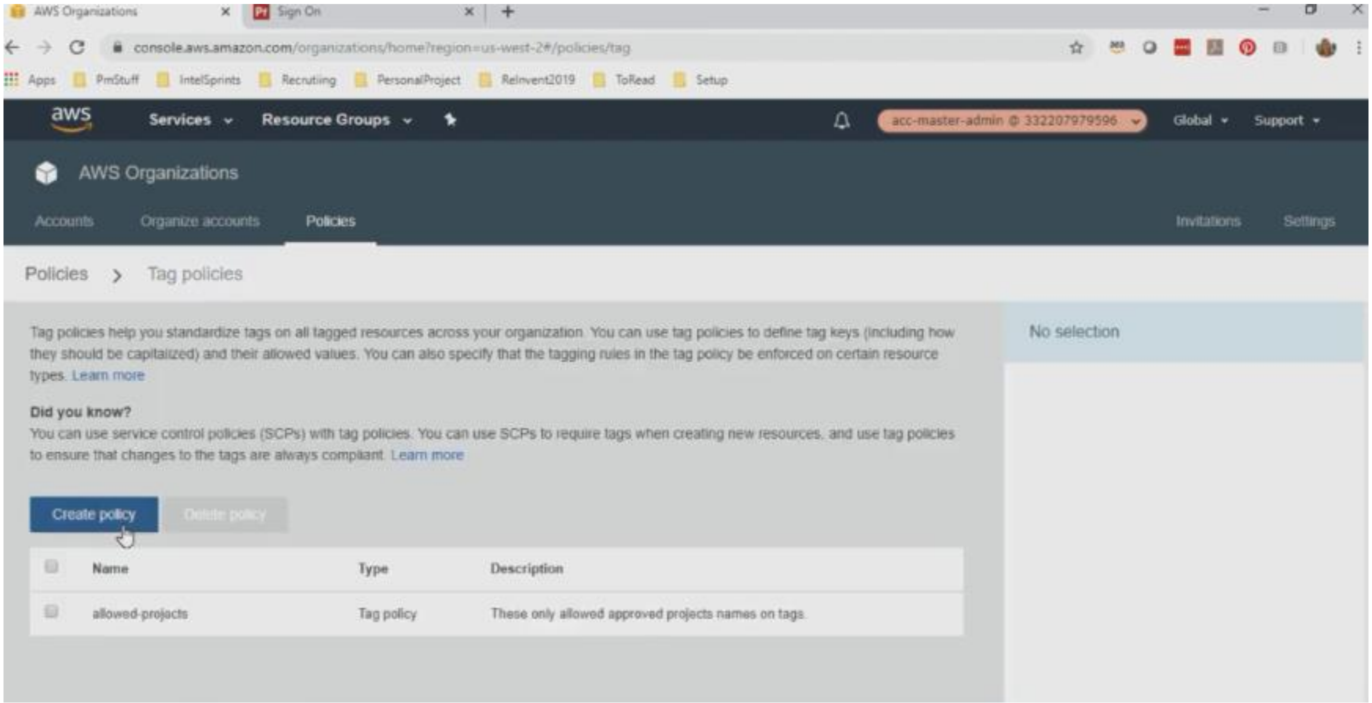
ID



JSON

```
{
    "tags": {
        "project": {
            "tag_key": {
                "@@assign": "project"
            },
            "tag_value": {
                "@@assign": [
                    "pickles",
                    "bubbles"
                ]
            },
            "enforced_for": {
                "@@assign": [
                    "codebuild:*",
                    "codecommit:*",
                    "secretsmanager:*"
```

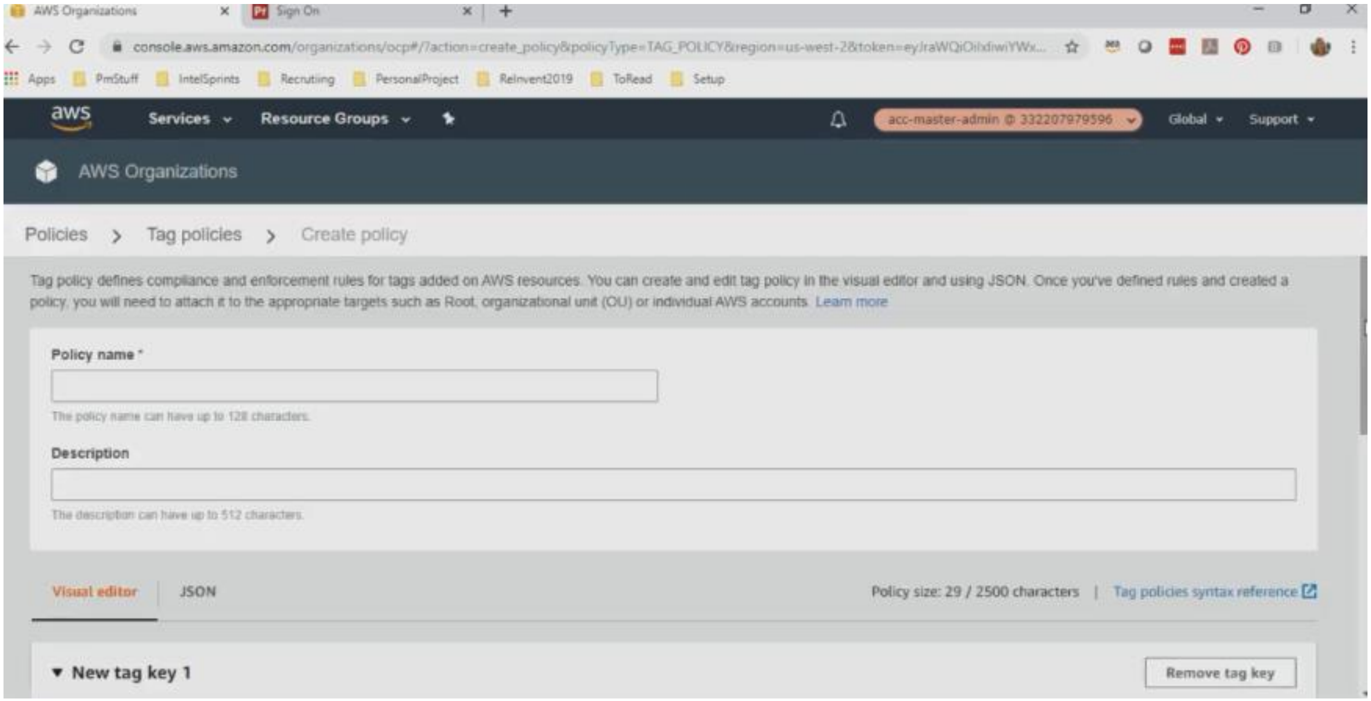
Feedback
English (US)
© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use



Tag policies help you standardize tags on all tagged resources across your organization. You can use tag policies to define tag keys (including how they should be capitalized) and their allowed values. You can also specify that the tagging rules in the tag policy be enforced on certain resource types. Learn more

Did you know?
You can use service control policies (SCPs) with tag policies. You can use SCPs to require tags when creating new resources, and use tag policies to ensure that changes to the tags are always compliant. Learn more

□	Name	Type	Description
□	allowed-projects	Tag policy	These only allowed approved projects names on tags.


AWS Organizations
Sign On
console.aws.amazon.com/organizations/ocp#/?action=create_policy&policyType=TAG_POLICY&region=us-west-2&token=eyJraWQiOiIxIiwiYWx...
Apps
PmStuff
IntelSprints
Recrutiing
PersonalProject
ReInvent2019
ToRead
Setup
aws
Services
Resource Groups
acc-master-admin @ 332207979596
Global
Support
AWS Organizations
Policies > Tag policies > Create policy
Visual editor
JSON
Tag policies syntax reference
Remove tag key

Tag policy defines compliance and enforcement rules for tags added on AWS resources. You can create and edit tag policy in the visual editor and using JSON. Once you've defined rules and created a policy, you will need to attach it to the appropriate targets such as Root, organizational unit (OU) or individual AWS accounts. Learn more

Policy name *

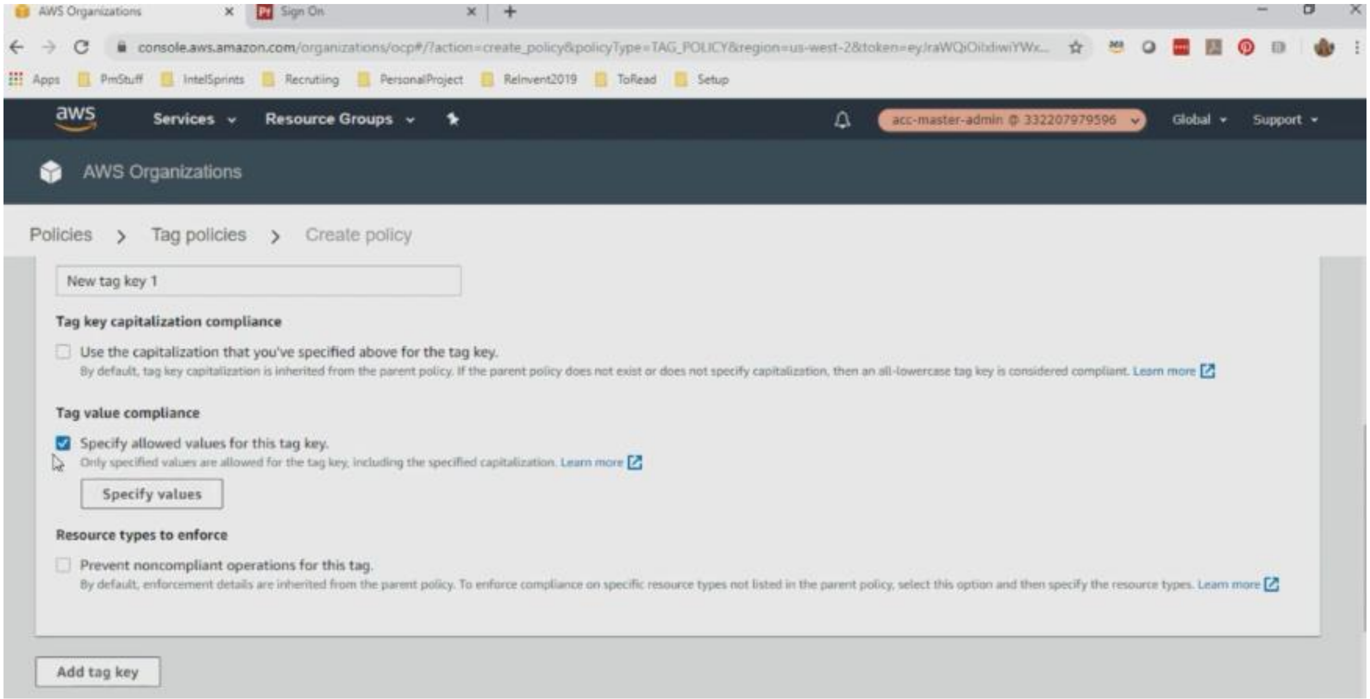
The policy name can have up to 128 characters.

Description

The description can have up to 512 characters.

Policy size: 29 / 2500 characters

▾ **New tag key 1**



AWS Organizations × | Sign On ×

console.aws.amazon.com/organizations/ocp#/?action=create_policy&policyType=TAG_POLICY&region=us-west-2&token=eyJraWQiOiIxIiwiYWx...

Apps | PmStuff | IntelSprints | Recruting | PersonalProject | ReInvent2019 | ToRead | Setup

aws | Services | Resource Groups | acc-master-admin @ 332207979596 | Global | Support

AWS Organizations

Policies > Tag policies > Create policy

New tag key 1

Tag key capitalization compliance

□ Use the capitalization that you've specified above for the tag key.
By default, tag key capitalization is inherited from the parent policy. If the parent policy does not exist or does not specify capitalization, then an all-lowercase tag key is considered compliant. Learn more

Tag value compliance

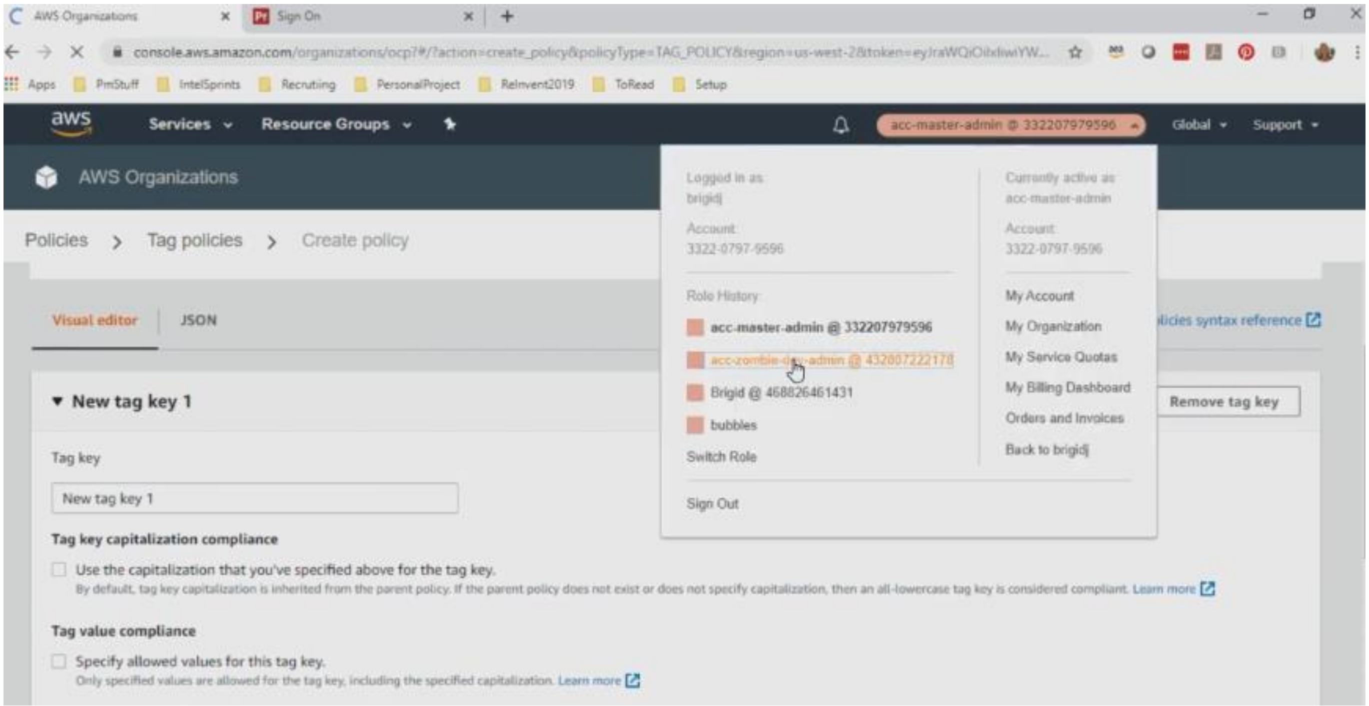
☑ Specify allowed values for this tag key.
Only specified values are allowed for the tag key, including the specified capitalization. Learn more

Specify values

Resource types to enforce

□ Prevent noncompliant operations for this tag.
By default, enforcement details are inherited from the parent policy. To enforce compliance on specific resource types not listed in the parent policy, select this option and then specify the resource types. Learn more

Add tag key



AWS Organizations × | Sign On ×

console.aws.amazon.com/organizations/ocp?#/?action=create_policy&policyType=TAG_POLICY&region=us-west-2&token=eyJraWQiOiIxIiwiYW...

Apps | PmStuff | IntelSprints | Recruting | PersonalProject | ReInvent2019 | ToRead | Setup

aws | Services | Resource Groups | acc-master-admin @ 332207979596 | Global | Support

AWS Organizations

Policies > Tag policies > Create policy

Visual editor | JSON | ...licies syntax reference

▾ **New tag key 1** | Remove tag key

Tag key

New tag key 1

Tag key capitalization compliance

□ Use the capitalization that you've specified above for the tag key.
By default, tag key capitalization is inherited from the parent policy. If the parent policy does not exist or does not specify capitalization, then an all-lowercase tag key is considered compliant. Learn more

Tag value compliance

□ Specify allowed values for this tag key.
Only specified values are allowed for the tag key, including the specified capitalization. Learn more

Logged in as
brigidj

Account:
3322-0797-9596

Currently active as
acc-master-admin

Account:
3322-0797-9596

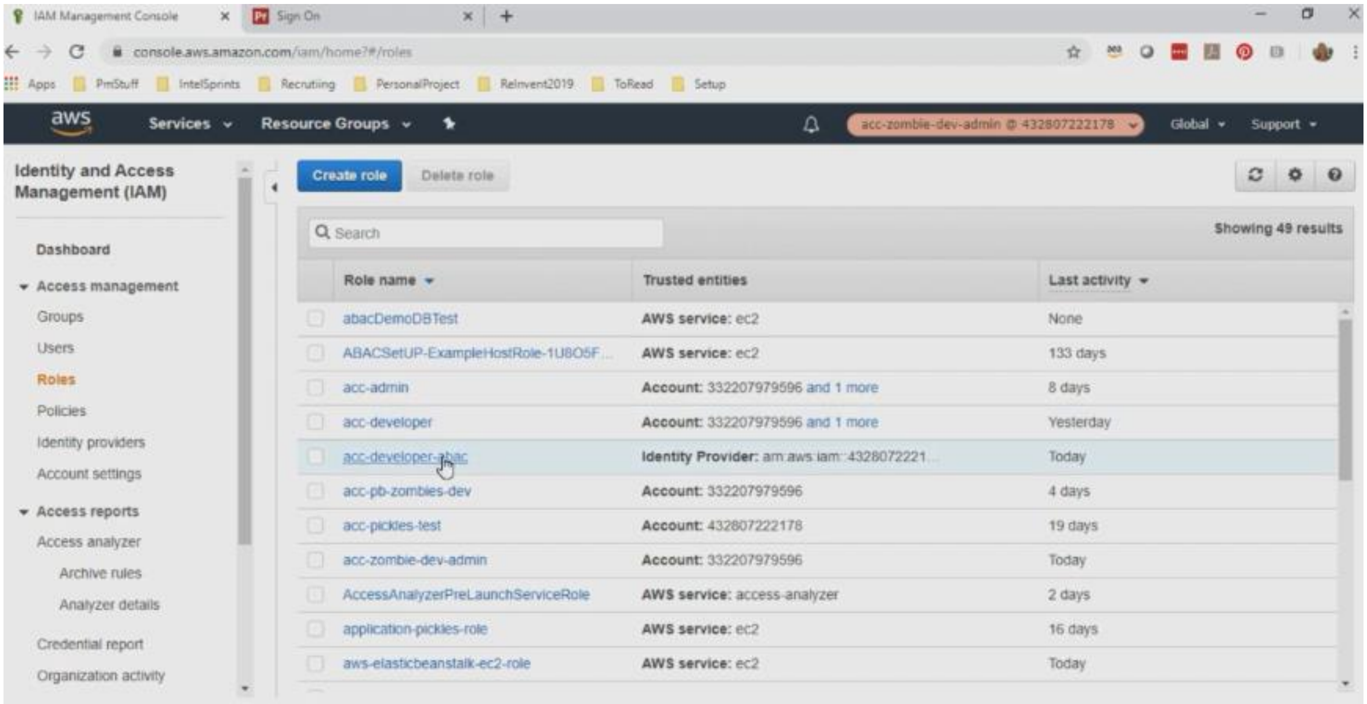
Role History:

- acc-master-admin @ 332207979596
- acc-zombie-dev-admin @ 432807222178
- Brigid @ 468826461431
- bubbles

Switch Role

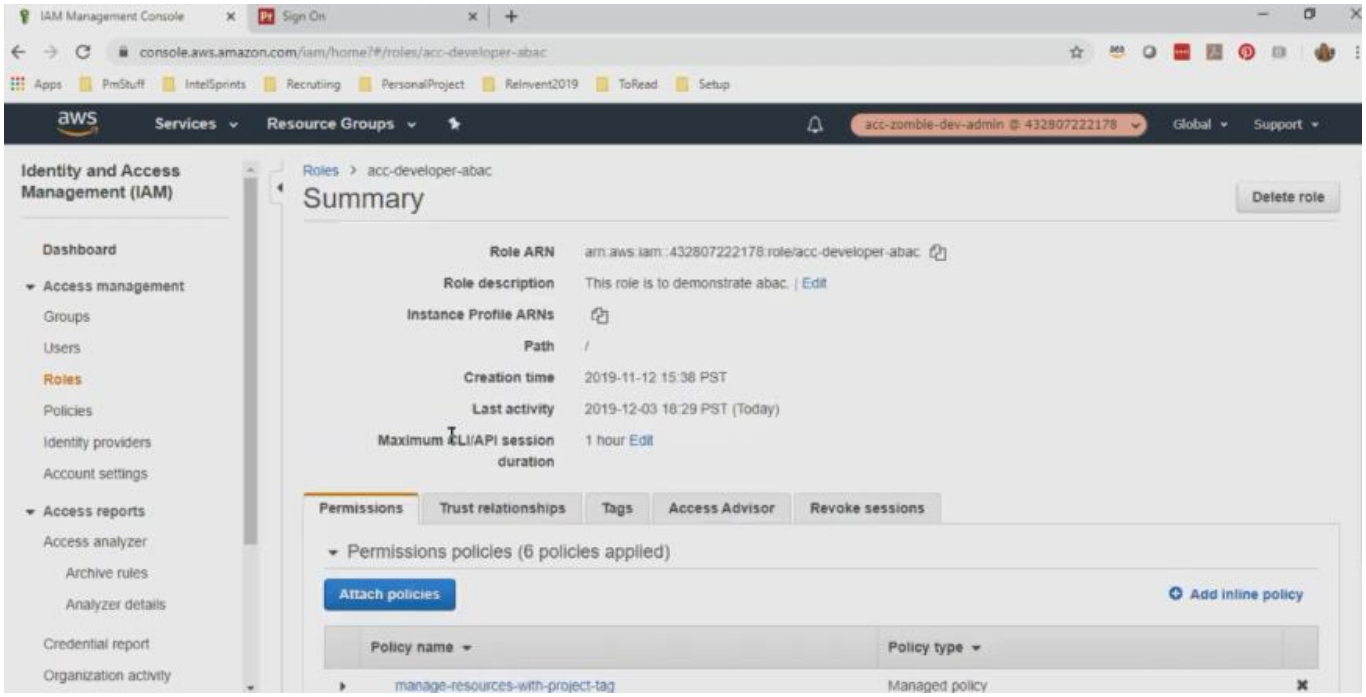
My Account
My Organization
My Service Quotas
My Billing Dashboard
Orders and Invoices
Back to brigidj

Sign Out



Showing 49 results

	Role name	Trusted entities	Last activity
□	abacDemoDBTest	**AWS service:** ec2	None
□	ABACSetUP-ExampleHostRole-1U8O5F...	**AWS service:** ec2	133 days
□	acc-admin	**Account:** 332207979596 and 1 more	8 days
□	acc-developer	**Account:** 332207979596 and 1 more	Yesterday
□	acc-developer-abac	**Identity Provider:** arn:aws:iam::4328072221...	Today
□	acc-pb-zombies-dev	**Account:** 332207979596	4 days
□	acc-pickles-test	**Account:** 432807222178	19 days
□	acc-zombie-dev-admin	**Account:** 332207979596	Today
□	AccessAnalyzerPreLaunchServiceRole	**AWS service:** access-analyzer	2 days
□	application-pickles-role	**AWS service:** ec2	16 days
□	aws-elasticbeanstalk-ec2-role	**AWS service:** ec2	Today

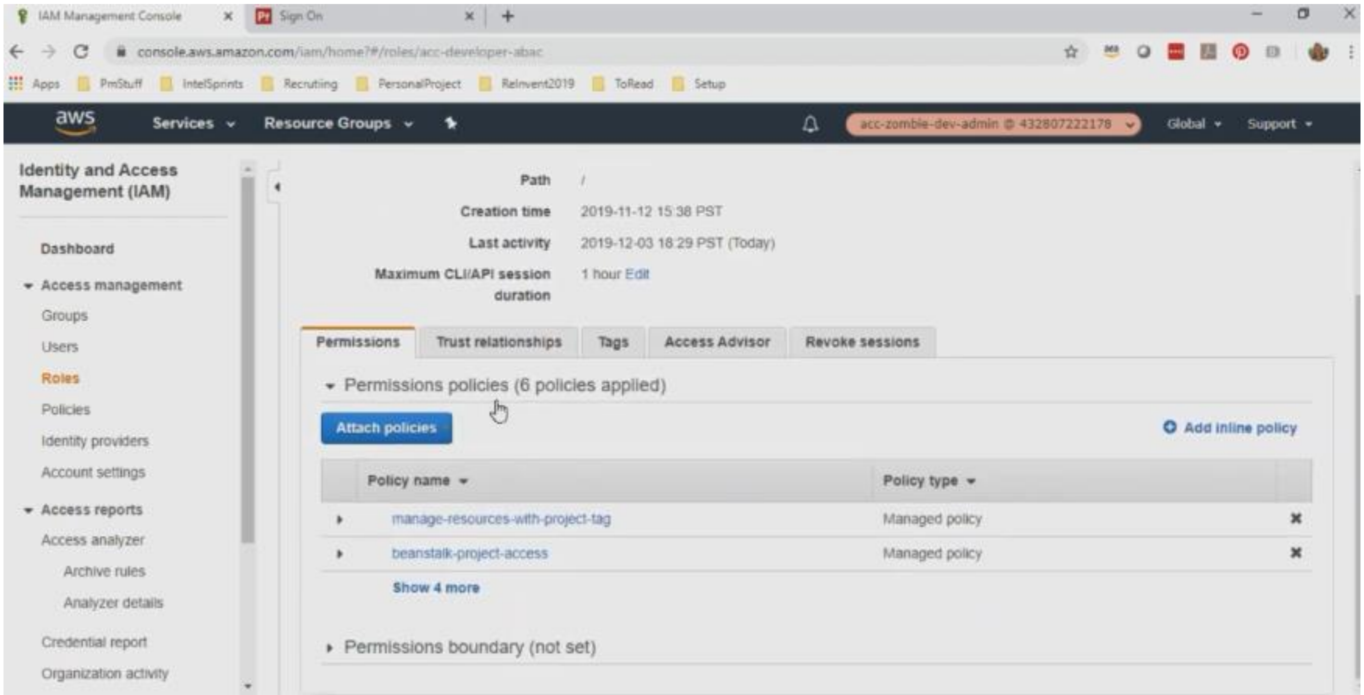


Summary

Role ARN	arn:aws:iam::432807222178:role/acc-developer-abac
Role description	This role is to demonstrate abac. | Edit
Instance Profile ARNs	
Path	/
Creation time	2019-11-12 15:38 PST
Last activity	2019-12-03 18:29 PST (Today)
Maximum CLI/API session duration	1 hour Edit

Permissions policies (6 policies applied)

Policy name	Policy type
manage-resources-with-project-tag	Managed policy

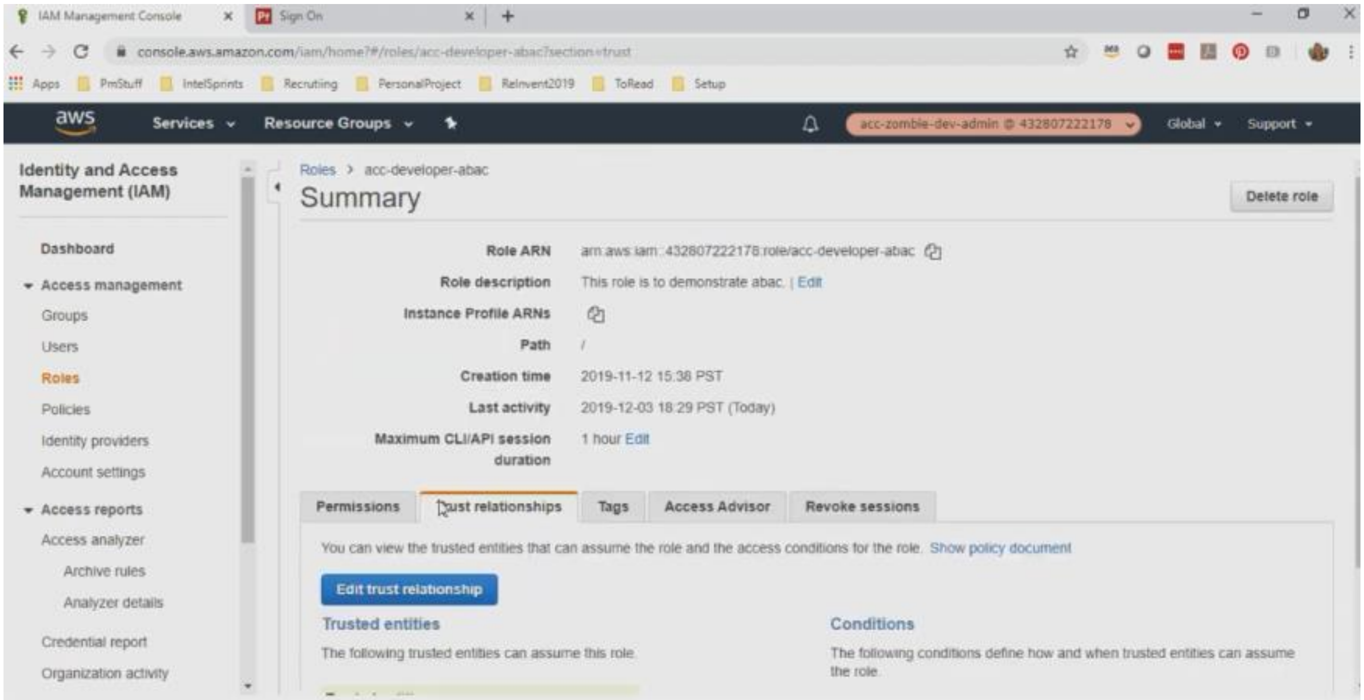


Path	/
Creation time	2019-11-12 15:38 PST
Last activity	2019-12-03 18:29 PST (Today)
Maximum CLI/API session duration	1 hour Edit

Permissions policies (6 policies applied)

Policy name	Policy type
manage-resources-with-project-tag	Managed policy
beanstalk-project-access	Managed policy

Permissions boundary (not set)



Summary

Role ARN	arn:aws:iam::432807222178:role/acc-developer-abac
Role description	This role is to demonstrate abac. | Edit
Instance Profile ARNs	
Path	/
Creation time	2019-11-12 15:38 PST
Last activity	2019-12-03 18:29 PST (Today)
Maximum CLI/API session duration	1 hour Edit

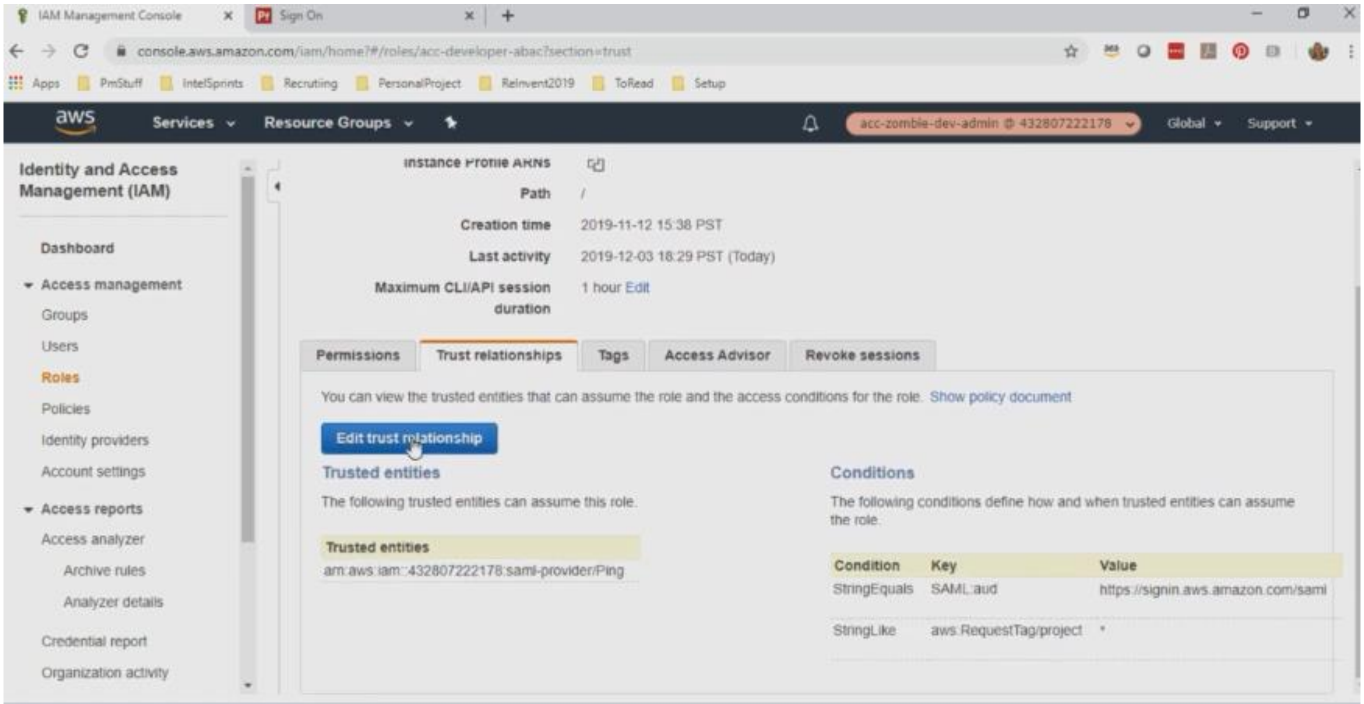
You can view the trusted entities that can assume the role and the access conditions for the role. Show policy document

Trusted entities

The following trusted entities can assume this role.

Conditions

The following conditions define how and when trusted entities can assume the role.



Instance Profile ARNs

Path /

Creation time 2019-11-12 15:38 PST

Last activity 2019-12-03 18:29 PST (Today)

Maximum CLI/API session duration 1 hour Edit

You can view the trusted entities that can assume the role and the access conditions for the role. Show policy document

Trusted entities

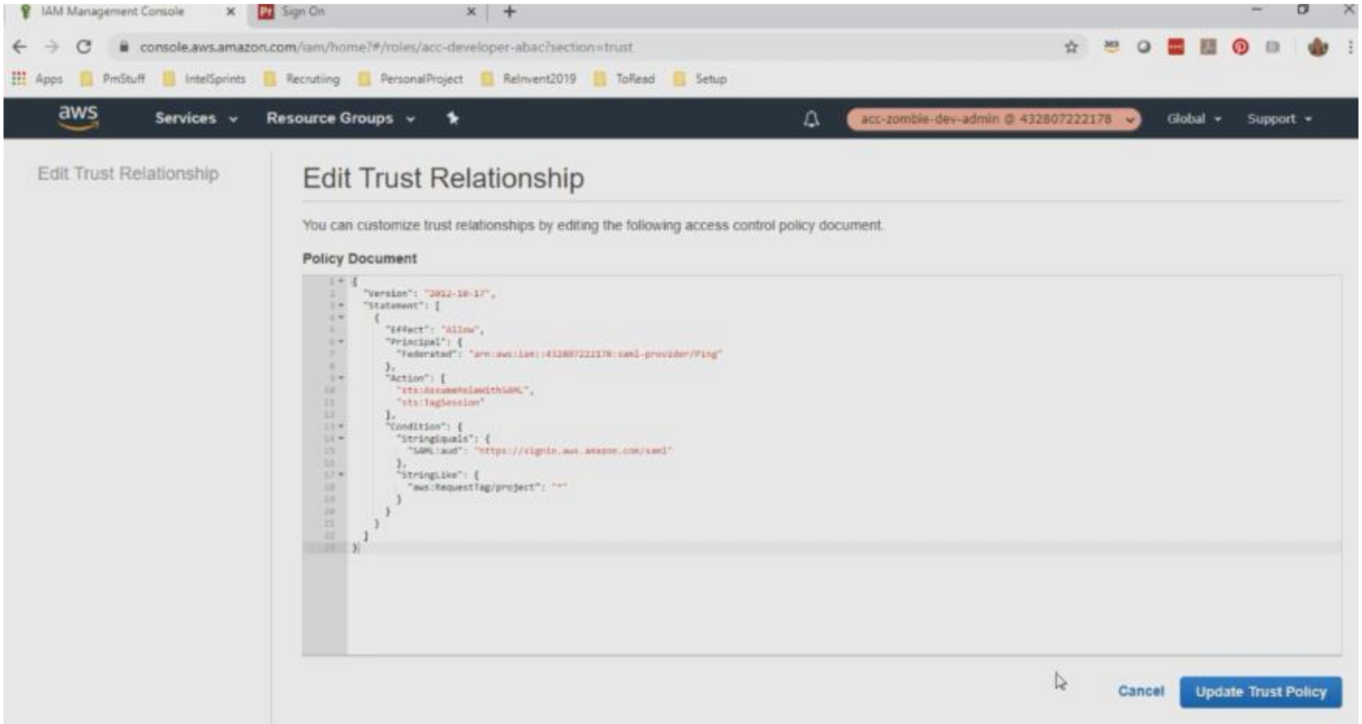
The following trusted entities can assume this role.

Trusted entities
arn:aws:iam::432807222178:saml-provider/Ping

Conditions

The following conditions define how and when trusted entities can assume the role.

Condition	Key	Value
StringEquals	SAML:aud	https://signin.aws.amazon.com/saml
StringLike	aws:RequestTag/project	*

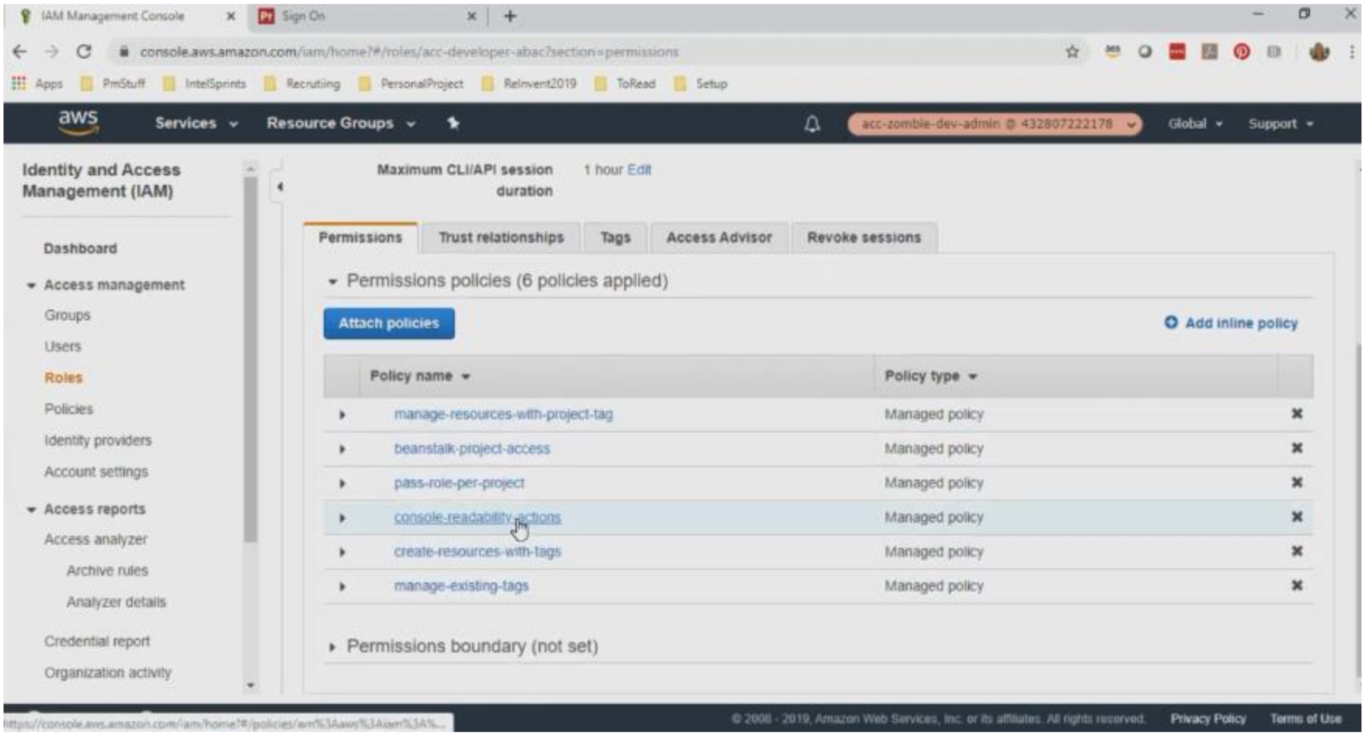


Edit Trust Relationship

You can customize trust relationships by editing the following access control policy document.

Policy Document

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Federated": "arn:aws:iam::432807222178:saml-provider/Ping"
      },
      "Action": [
        "sts:AssumeRoleWithSAML",
        "sts:TagSession"
      ],
      "Condition": {
        "StringEquals": {
          "SAML:aud": "https://signin.aws.amazon.com/saml"
        },
        "StringLike": {
          "aws:RequestTag/project": "*"
        }
      }
    }
  ]
}
```

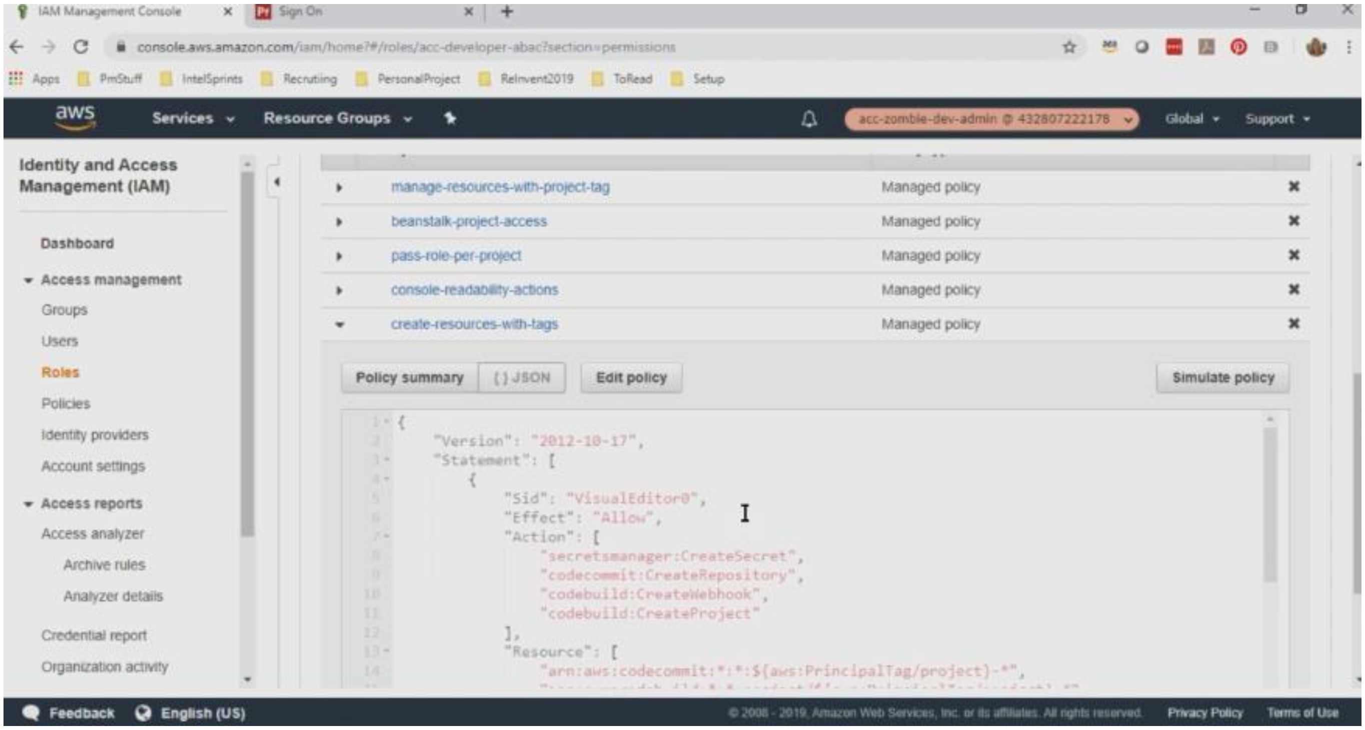


Maximum CLI/API session duration: 1 hour Edit

Permissions policies (6 policies applied)

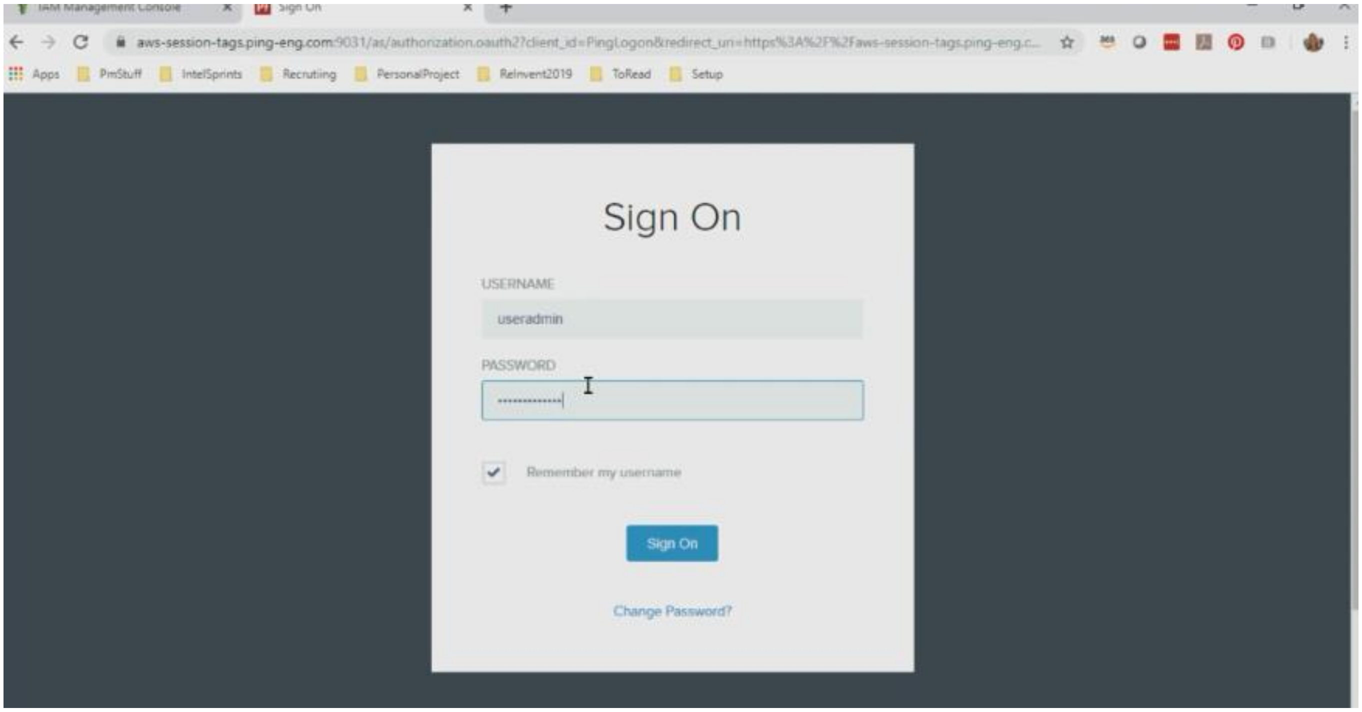
Policy name	Policy type
manage-resources-with-project-tag	Managed policy
beanstalk-project-access	Managed policy
pass-role-per-project	Managed policy
console-readability-actions	Managed policy
create-resources-with-tags	Managed policy
manage-existing-tags	Managed policy

Permissions boundary (not set)



Policy name	Policy type
manage-resources-with-project-tag	Managed policy
beanstalk-project-access	Managed policy
pass-role-per-project	Managed policy
console-readability-actions	Managed policy
create-resources-with-tags	Managed policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:CreateSecret",
                "codecommit:CreateRepository",
                "codebuild:CreateWebhook",
                "codebuild:CreateProject"
            ],
            "Resource": [
                "arn:aws:codecommit:*:*:${aws:PrincipalTag/project}-*",
```



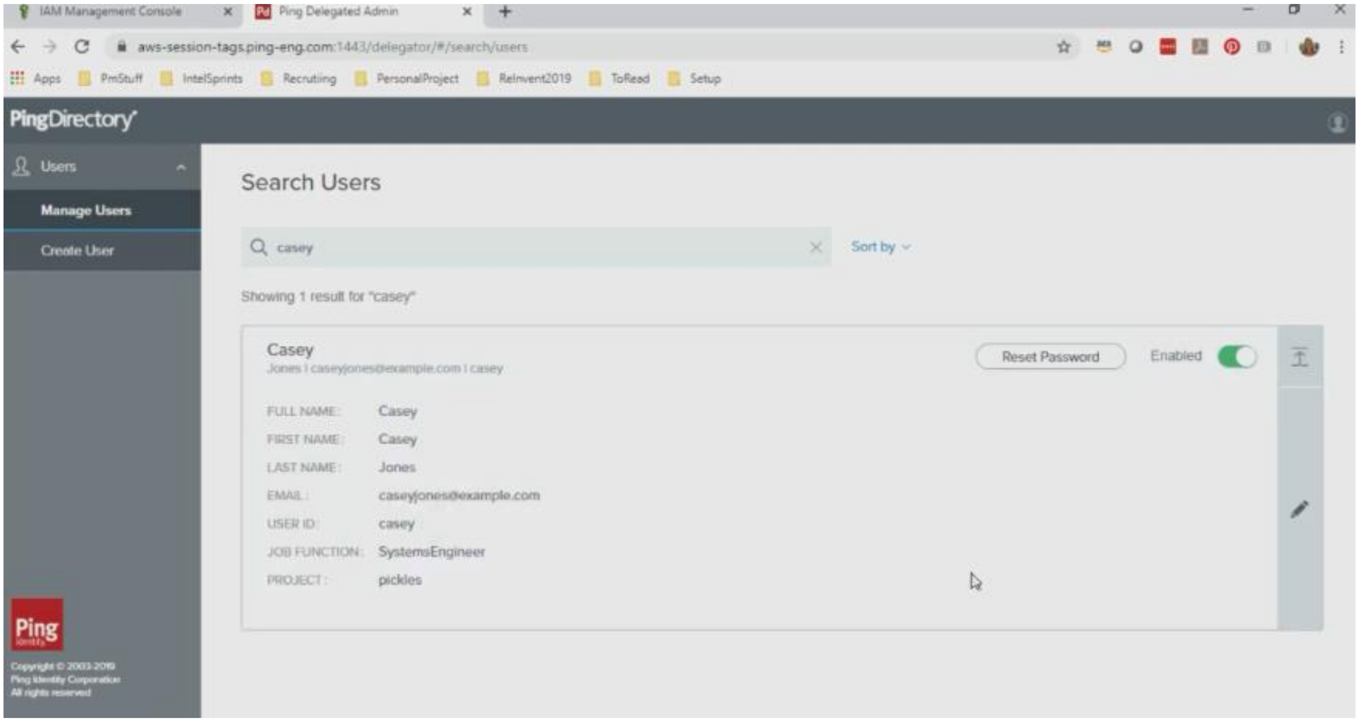
IAM Management Console
Sign On
aws-session-tags.ping-eng.com:9031/as/authorization.oauth2?client_id=PingLogon&redirect_uri=https%3A%2F%2Faws-session-tags.ping-eng.c...
Apps
PmStuff
IntelSprints
Recruiting
PersonalProject
ReInvent2019
ToRead
Setup

Sign On

USERNAME: useradmin

PASSWORD: ••••••••••••

☑ Remember my username



Search Users

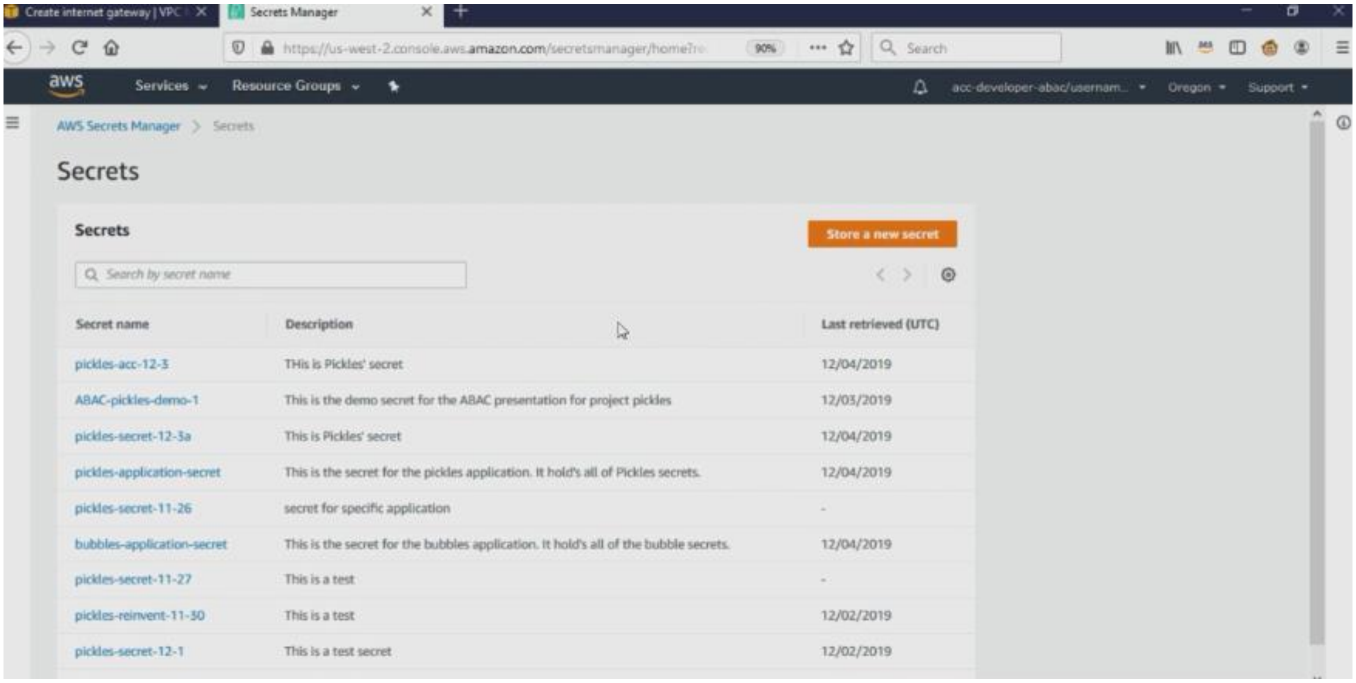
casey

Showing 1 result for "casey"

Casey
Jones | caseyjones@example.com | casey

FULL NAME: Casey
FIRST NAME: Casey
LAST NAME: Jones
EMAIL: caseyjones@example.com
USER ID: casey
JOB FUNCTION: SystemsEngineer
PROJECT: pickles

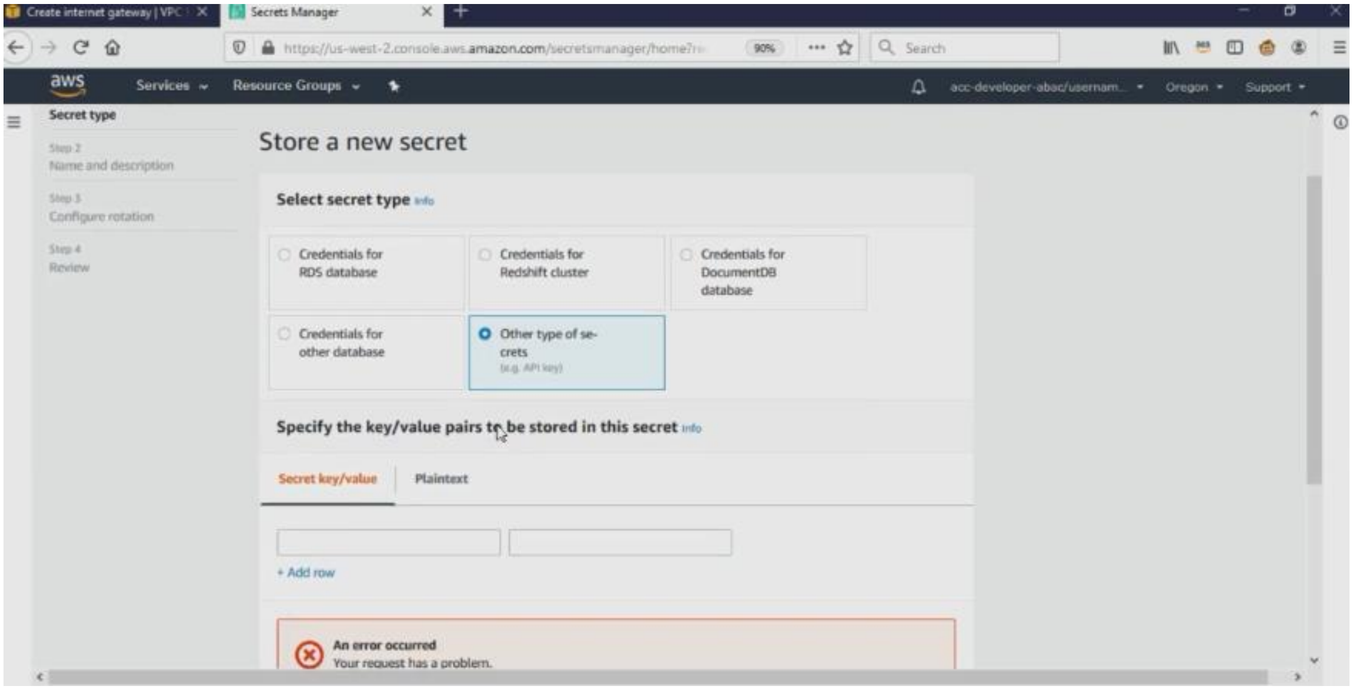
This context will all come along when Casey federates using Ping


Create internet gateway | VPC
Secrets Manager
https://us-west-2.console.aws.amazon.com/secretsmanager/home?r
90%
Search
aws
Services
Resource Groups
acc-developer-abac/usernam...
Oregon
Support
AWS Secrets Manager > Secrets
Store a new secret
Search by secret name

Secrets

Secrets

Secret name	Description	Last retrieved (UTC)
pickles-acc-12-3	THis is Pickles' secret	12/04/2019
ABAC-pickles-demo-1	This is the demo secret for the ABAC presentation for project pickles	12/03/2019
pickles-secret-12-3a	This is Pickles' secret	12/04/2019
pickles-application-secret	This is the secret for the pickles application. It hold's all of Pickles secrets.	12/04/2019
pickles-secret-11-26	secret for specific application	-
bubbles-application-secret	This is the secret for the bubbles application. It hold's all of the bubble secrets.	12/04/2019
pickles-secret-11-27	This is a test	-
pickles-reinvent-11-30	This is a test	12/02/2019
pickles-secret-12-1	This is a test secret	12/02/2019



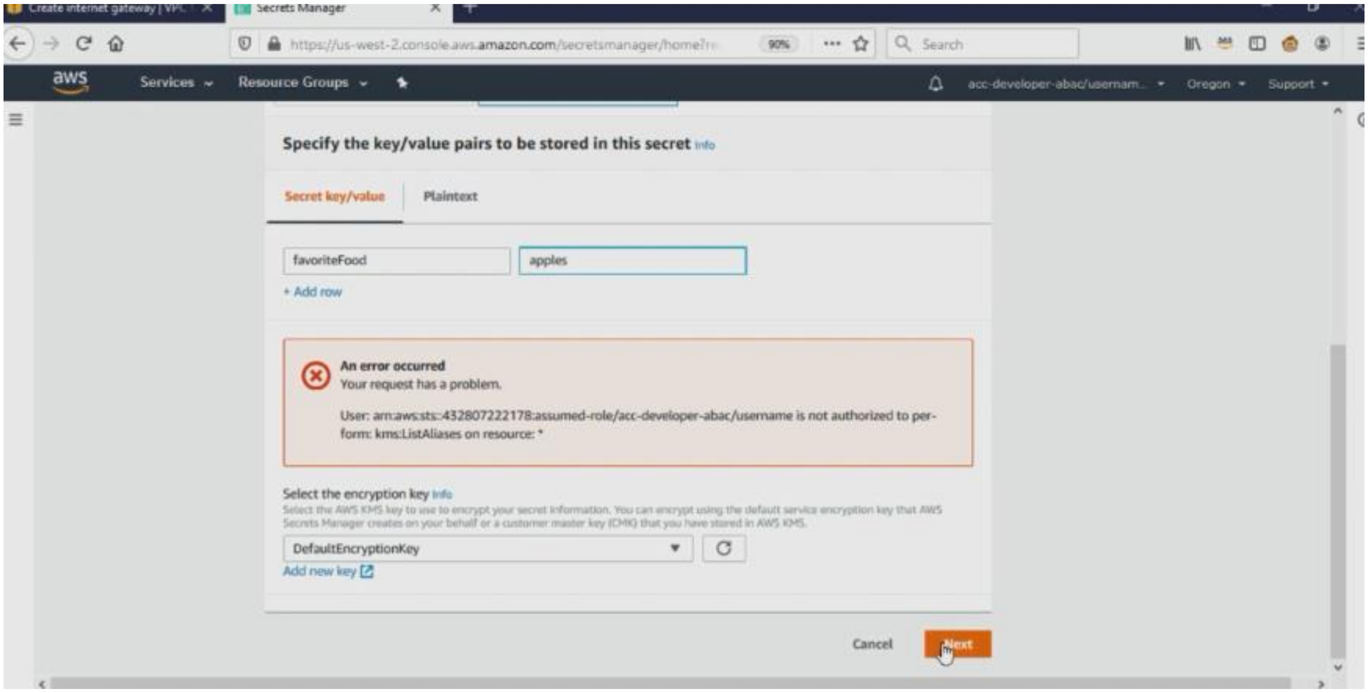
Store a new secret

Select secret type

○ Credentials for RDS database
○ Credentials for Redshift cluster
○ Credentials for DocumentDB database
○ Credentials for other database
● Other type of secrets (e.g. API key)

Specify the key/value pairs to be stored in this secret

An error occurred
Your request has a problem.


Create internet gateway | VPC
Secrets Manager
https://us-west-2.console.aws.amazon.com/secretsmanager/home?r
90%
Search
aws
Services
Resource Groups
acc-developer-abac/usernam...
Oregon
Support

Specify the key/value pairs to be stored in this secret Info

Secret key/value | Plaintext

favoriteFood | apples

+ Add row

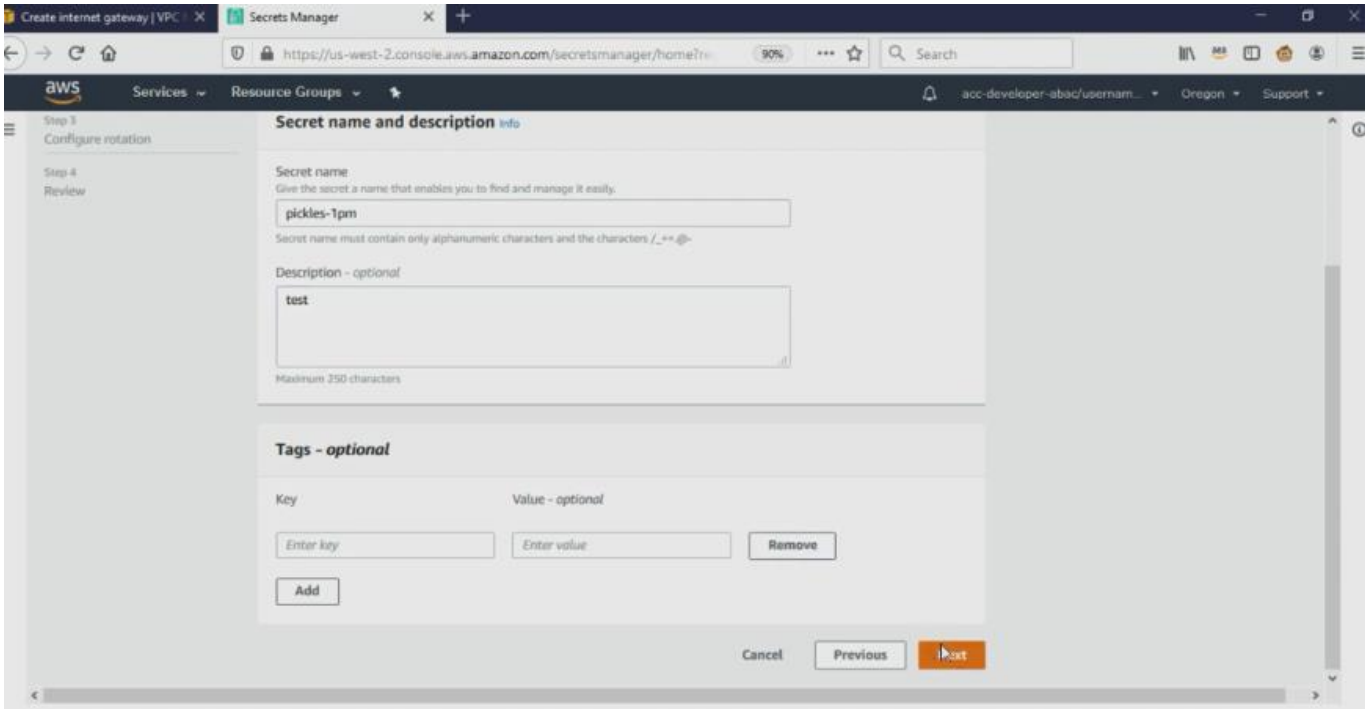
An error occurred
Your request has a problem.

User: arn:aws:sts::432807222178:assumed-role/acc-developer-abac/username is not authorized to perform: kms:ListAliases on resource: *

Select the encryption key Info
Select the AWS KMS key to use to encrypt your secret information. You can encrypt using the default service encryption key that AWS Secrets Manager creates on your behalf or a customer master key (CMK) that you have stored in AWS KMS.

DefaultEncryptionKey

Add new key


Create internet gateway | VPC
Secrets Manager
https://us-west-2.console.aws.amazon.com/secretsmanager/home?r
90%
Search
aws
Services
Resource Groups
acc-developer-abac/usernam...
Oregon
Support
Step 3
Configure rotation
Step 4
Review

Secret name and description Info

Secret name
Give the secret a name that enables you to find and manage it easily.

pickles-1pm

Secret name must contain only alphanumeric characters and the characters /_+=.@-

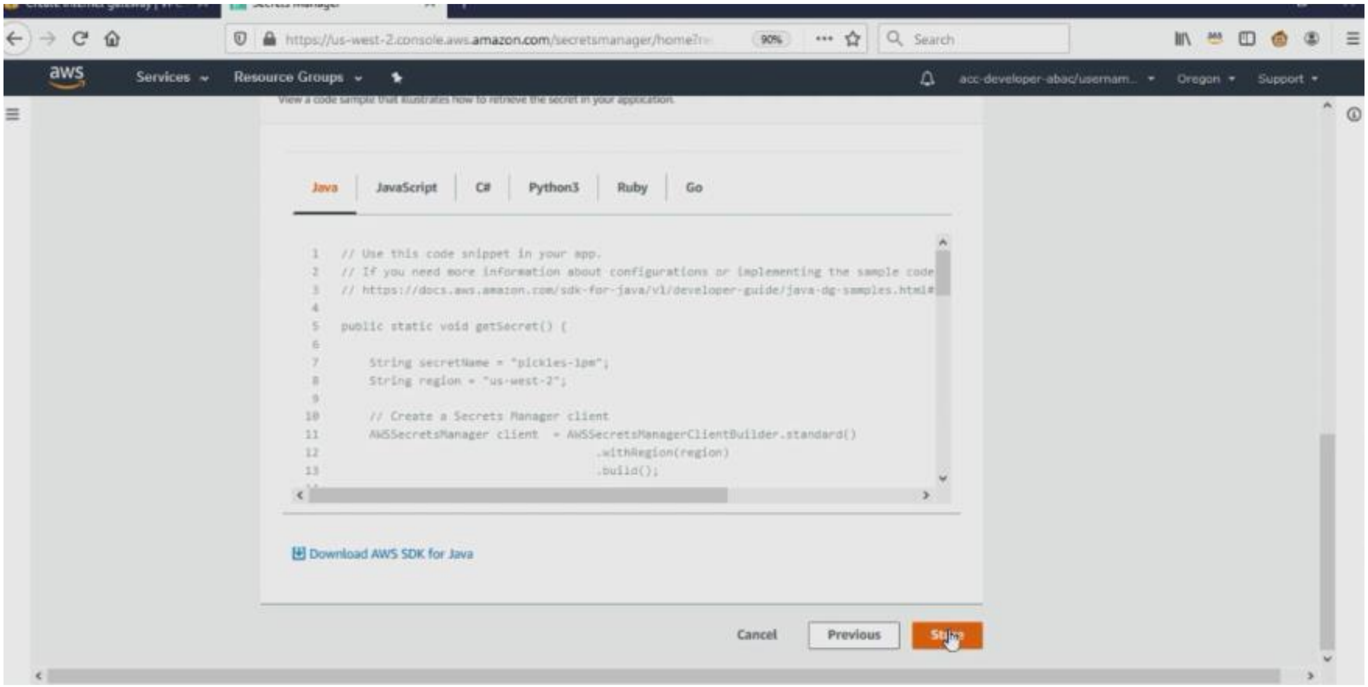
Description - *optional*

test

Maximum 250 characters

Tags - *optional*

Key | Value - *optional*



View a code sample that illustrates how to retrieve the secret in your application.

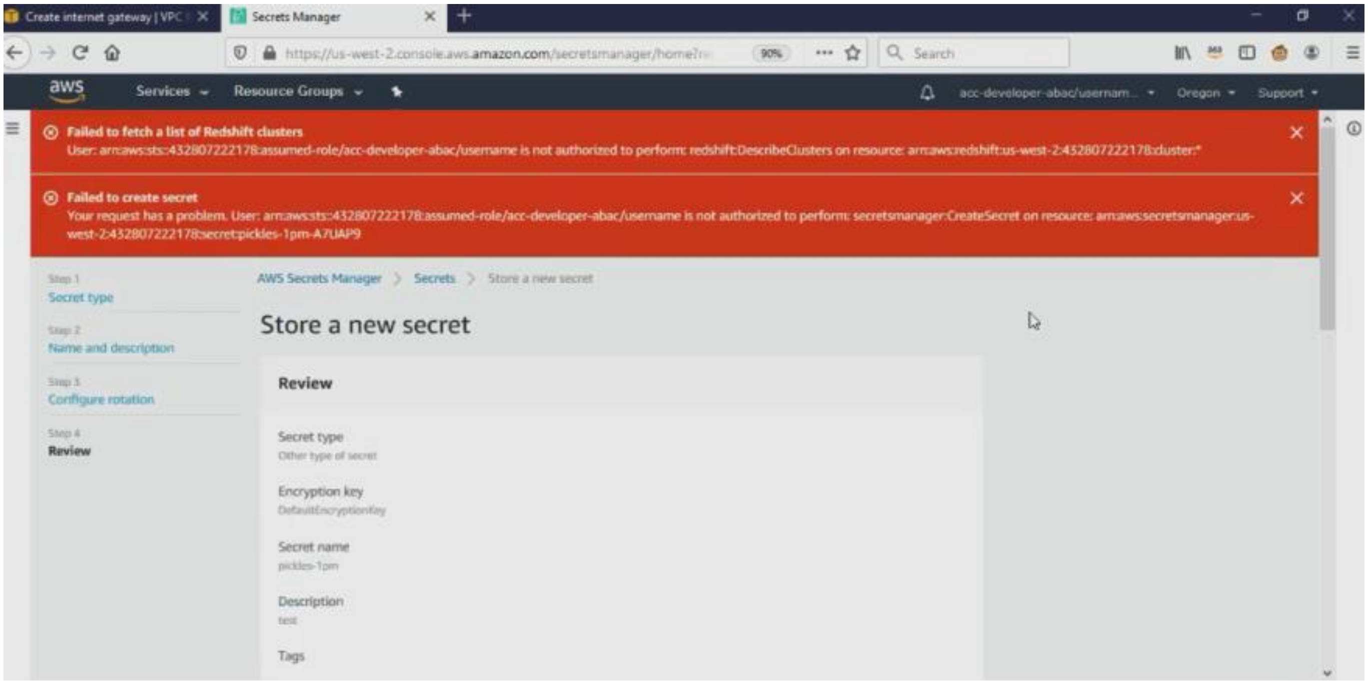
```
// Use this code snippet in your app.
// If you need more information about configurations or implementing the sample code
// https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/java-dg-samples.html#

public static void getSecret() {

    String secretName = "pickles-1pm";
    String region = "us-west-2";

    // Create a Secrets Manager client
    AWSSecretsManager client  = AWSSecretsManagerClientBuilder.standard()
                                    .withRegion(region)
                                    .build();
```

Download AWS SDK for Java
Cancel
Previous
Store



Create internet gateway | VPC
Secrets Manager
https://us-west-2.console.aws.amazon.com/secretsmanager/home?r
Services
Resource Groups
acc-developer-abac/usernam...
Oregon
Support

Failed to fetch a list of Redshift clusters
User: arn:aws:sts::432807222178:assumed-role/acc-developer-abac/username is not authorized to perform: redshift:DescribeClusters on resource: arn:aws:redshift:us-west-2:432807222178:cluster:*

Failed to create secret
Your request has a problem. User: arn:aws:sts::432807222178:assumed-role/acc-developer-abac/username is not authorized to perform: secretsmanager:CreateSecret on resource: arn:aws:secretsmanager:us-west-2:432807222178:secret:pickles-1pm-A7UAP9

Store a new secret

Review

Secret type
Other type of secret

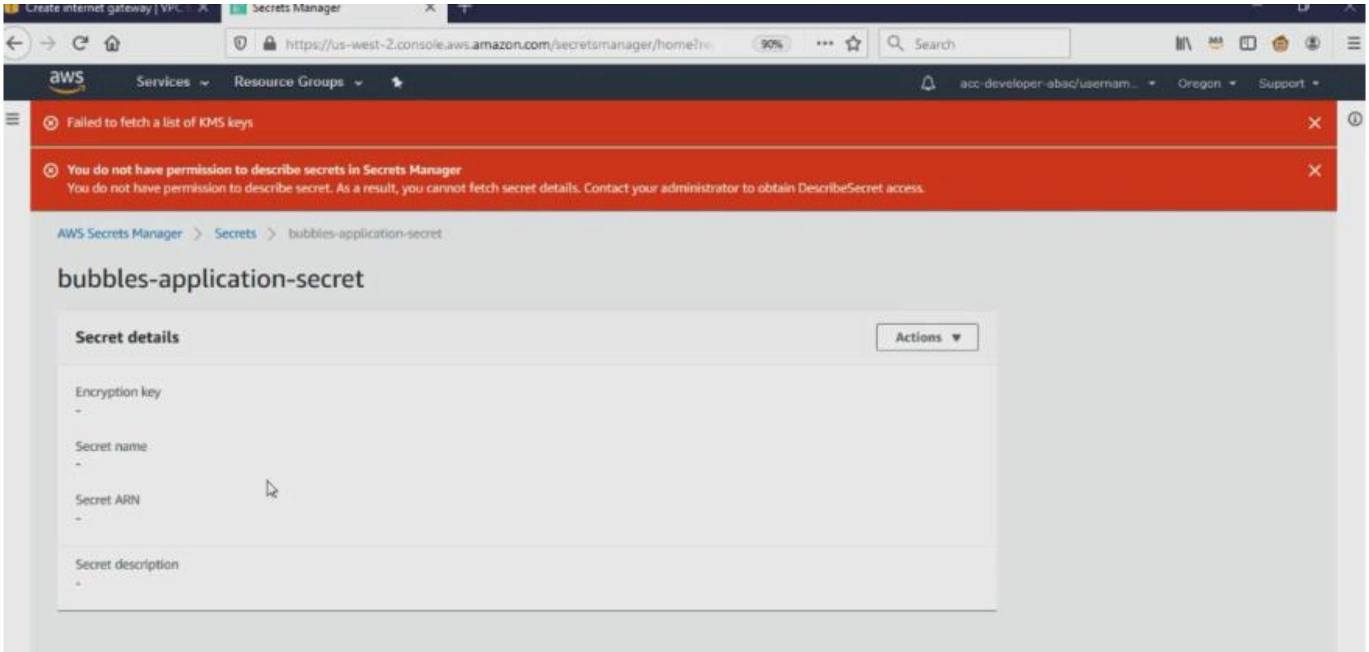
Encryption key
DefaultEncryptionKey

Secret name
pickles-1pm

Description
test

Tags

Casey needs to tag it with the project name starting with Pickles from the Ping federation



Create internet gateway | VPC
Secrets Manager
https://us-west-2.console.aws.amazon.com/secretsmanager/home?re
90%
Search
aws
Services
Resource Groups
acc-developer-abac/usernam...
Oregon
Support

Failed to fetch a list of KMS keys

You do not have permission to describe secrets in Secrets Manager
You do not have permission to describe secret. As a result, you cannot fetch secret details. Contact your administrator to obtain DescribeSecret access.

AWS Secrets Manager > Secrets > bubbles-application-secret

bubbles-application-secret

Secret details

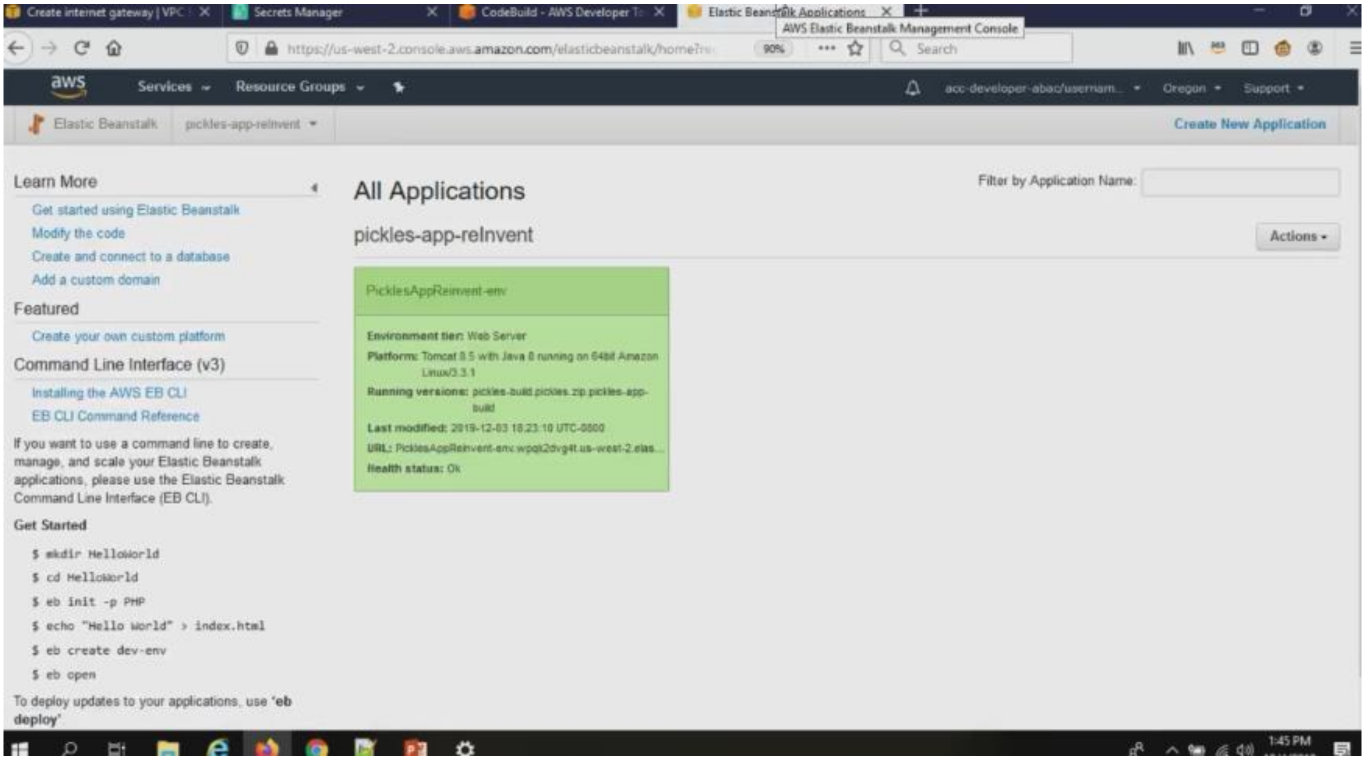
Encryption key
-

Secret name
-

Secret ARN
-

Secret description
-

Casey cannot see Bubbles secrets, only Pickles



Create internet gateway | VPC
Secrets Manager
CodeBuild - AWS Developer T
Elastic Beanstalk Applications
AWS Elastic Beanstalk Management Console
https://us-west-2.console.aws.amazon.com/elasticbeanstalk/home?re
90%
Search
aws
Services
Resource Groups
acc-developer-abac/usernam...
Oregon
Support
Elastic Beanstalk
pickles-app-relnvent
Create New Application
Learn More
Get started using Elastic Beanstalk
Modify the code
Create and connect to a database
Add a custom domain
Featured
Create your own custom platform
Command Line Interface (v3)
Installing the AWS EB CLI
EB CLI Command Reference

If you want to use a command line to create, manage, and scale your Elastic Beanstalk applications, please use the Elastic Beanstalk Command Line Interface (EB CLI).

Get Started

```
$ mkdir HelloWorld
$ cd HelloWorld
$ eb init -p PHP
$ echo "Hello World" > index.html
$ eb create dev-env
$ eb open
```

To deploy updates to your applications, use '**eb deploy**'

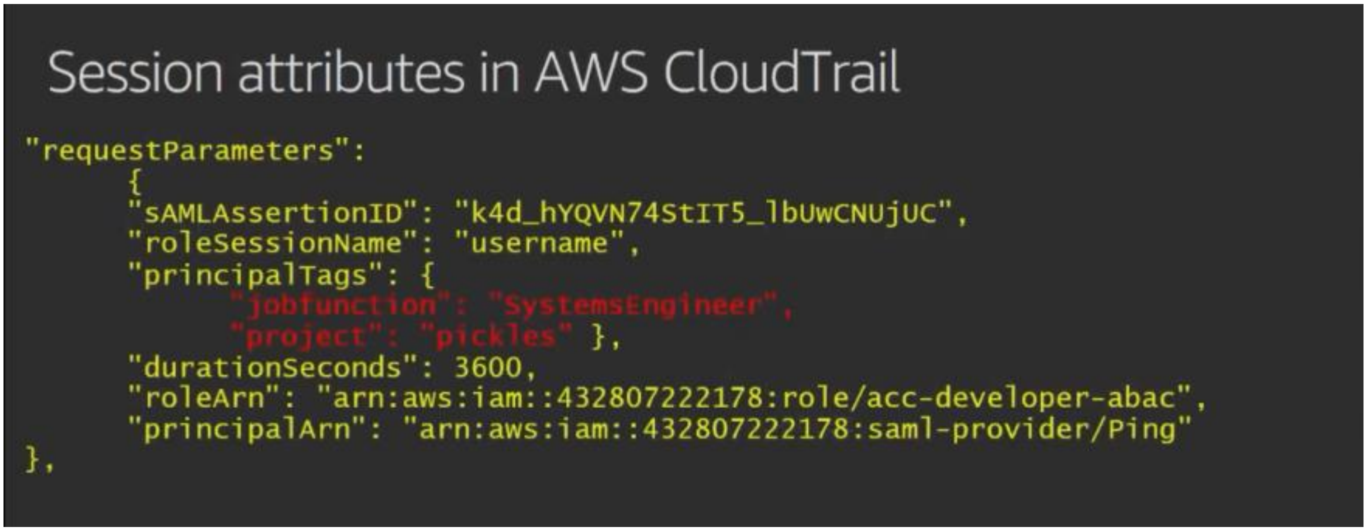
All Applications

Filter by Application Name:
Actions

pickles-app-relnvent

PicklesAppReinvent-env

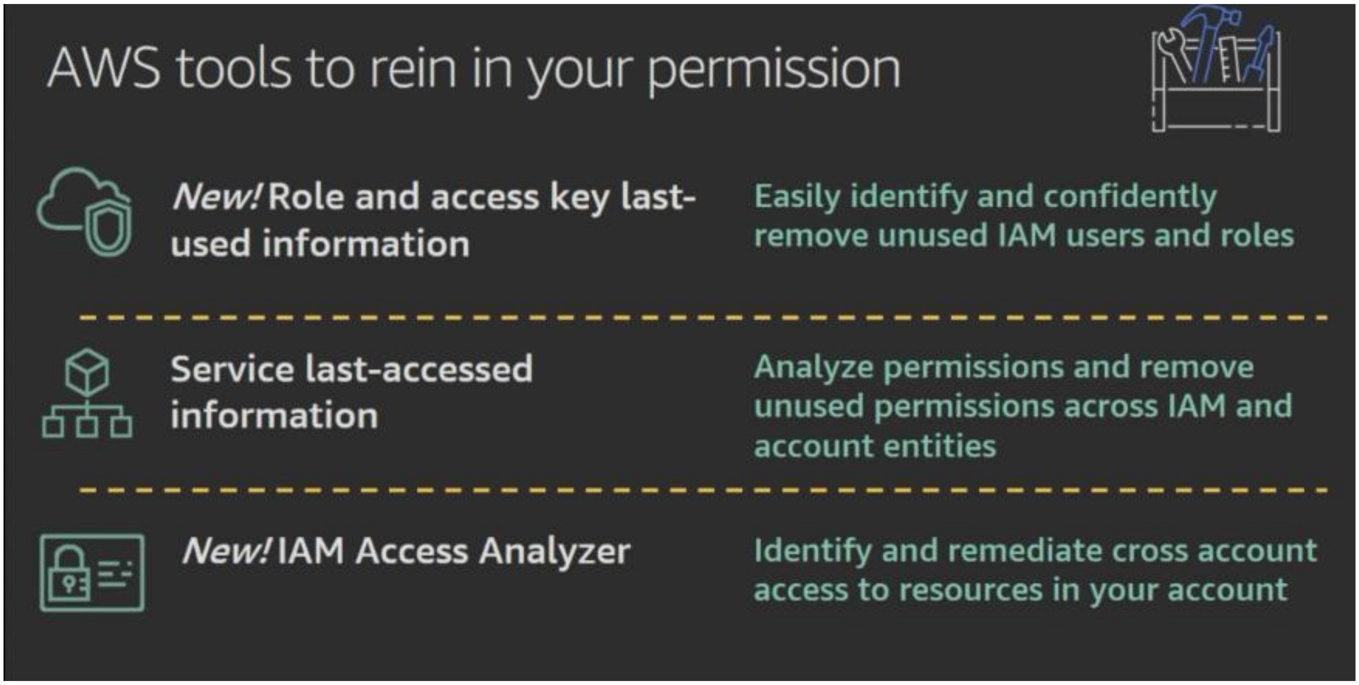
Environment tier: Web Server
Platform: Tomcat 8.5 with Java 8 running on 64bit Amazon Linux/3.3.1
Running versions: pickles-build.pickles.zip.pickles-app-build
Last modified: 2019-12-03 18:23:10 UTC-0800
URL: PicklesAppReinvent-env.wpqt2dvg4t.us-west-2.elas...
Health status: Ok



Session attributes in AWS CloudTrail

```
"requestParameters":
      {
      "sAMLAssertionID": "k4d_hYQVN74StIT5_lbUwCNUjUC",
      "roleSessionName": "username",
      "principalTags": {
            "jobfunction": "SystemsEngineer",
            "project": "pickles" },
      "durationSeconds": 3600,
      "roleArn": "arn:aws:iam::432807222178:role/acc-developer-abac",
      "principalArn": "arn:aws:iam::432807222178:saml-provider/Ping"
},
```

Use analytics to rein in permissions

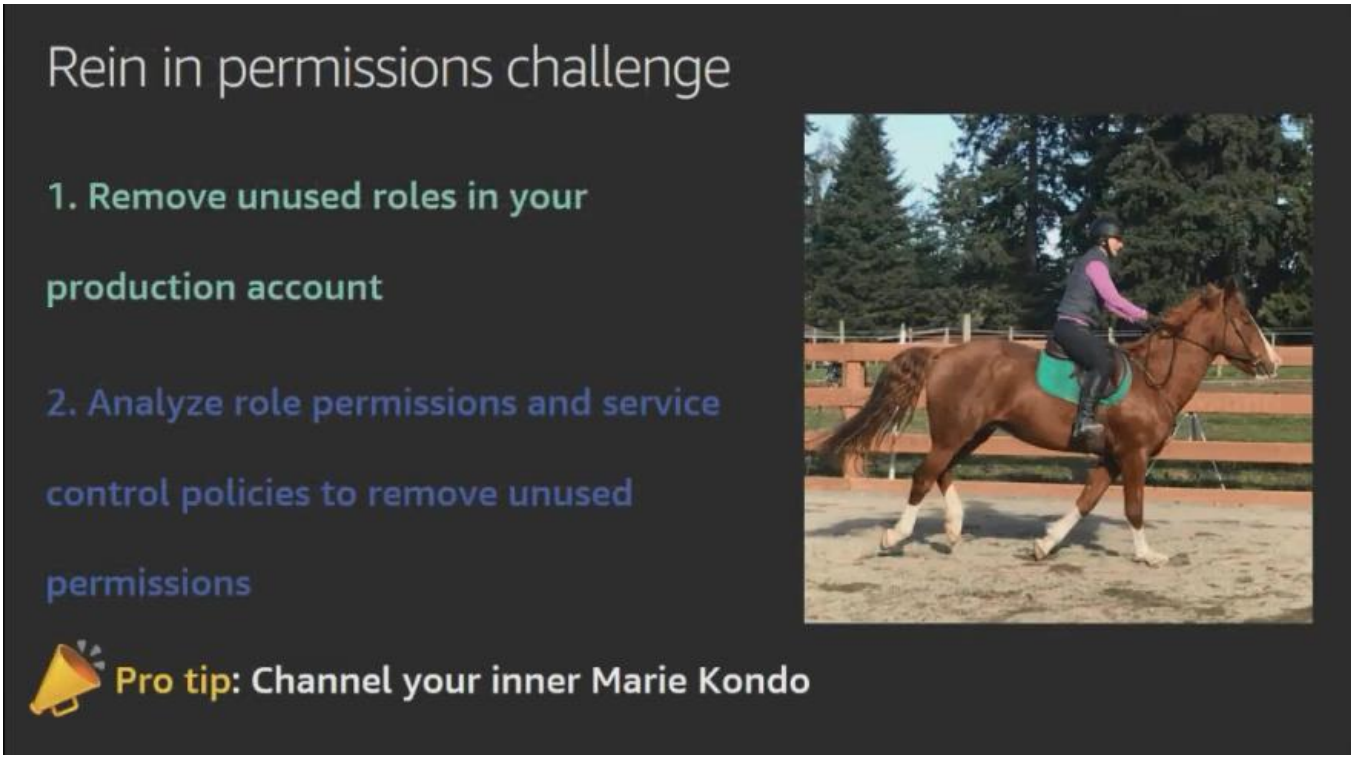


AWS tools to rein in your permission

New! **Role and access key last-used information** — Easily identify and confidently remove unused IAM users and roles

Service last-accessed information — Analyze permissions and remove unused permissions across IAM and account entities

New! **IAM Access Analyzer** — Identify and remediate cross account access to resources in your account

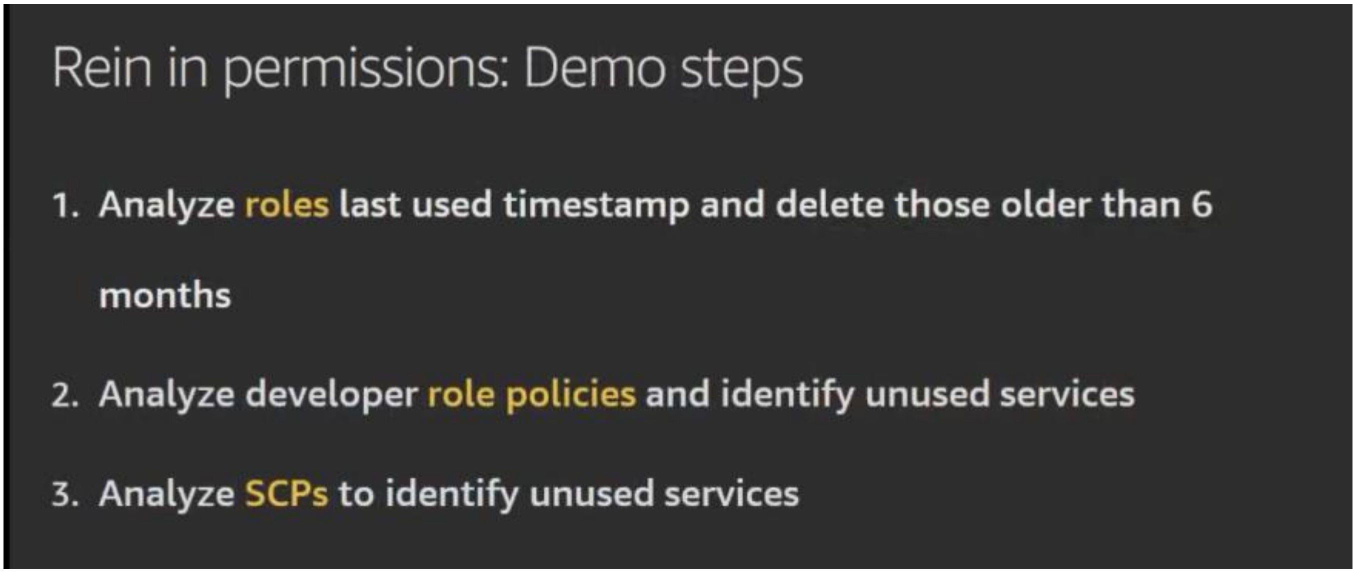


Rein in permissions challenge

1. Remove unused roles in your production account

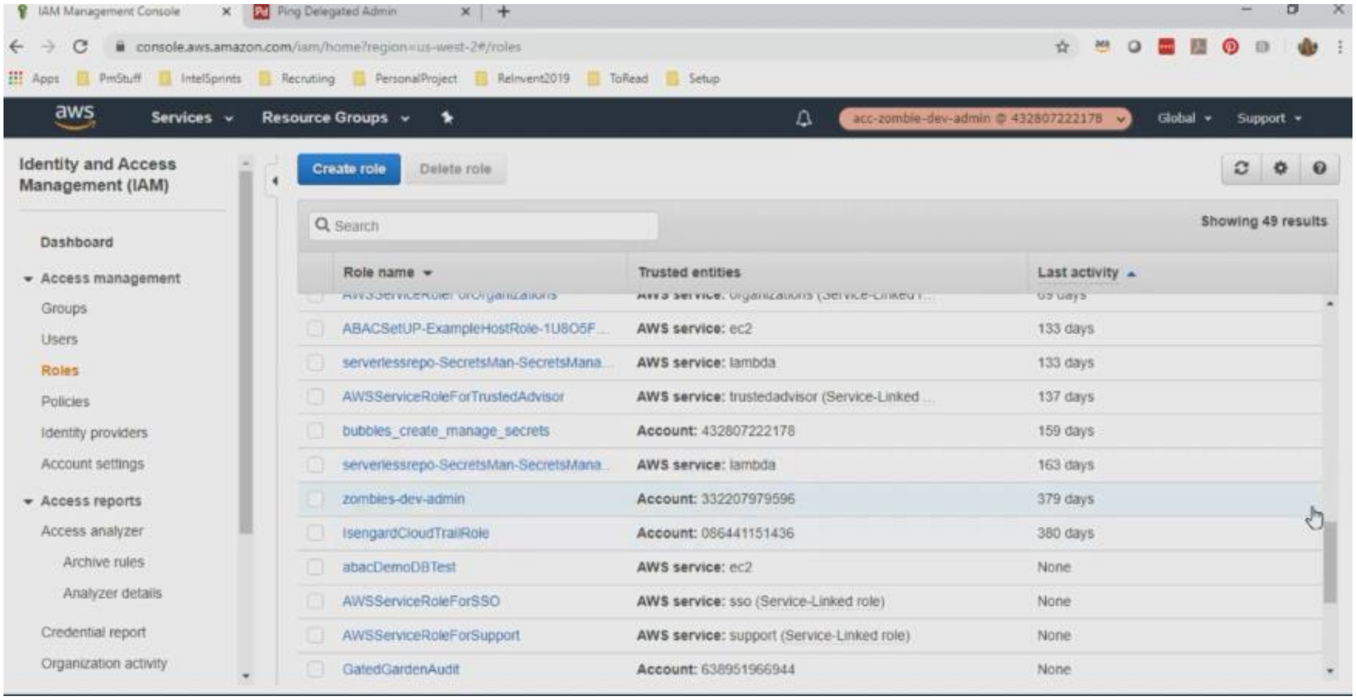
2. Analyze role permissions and service control policies to remove unused permissions

📣 **Pro tip**: Channel your inner Marie Kondo



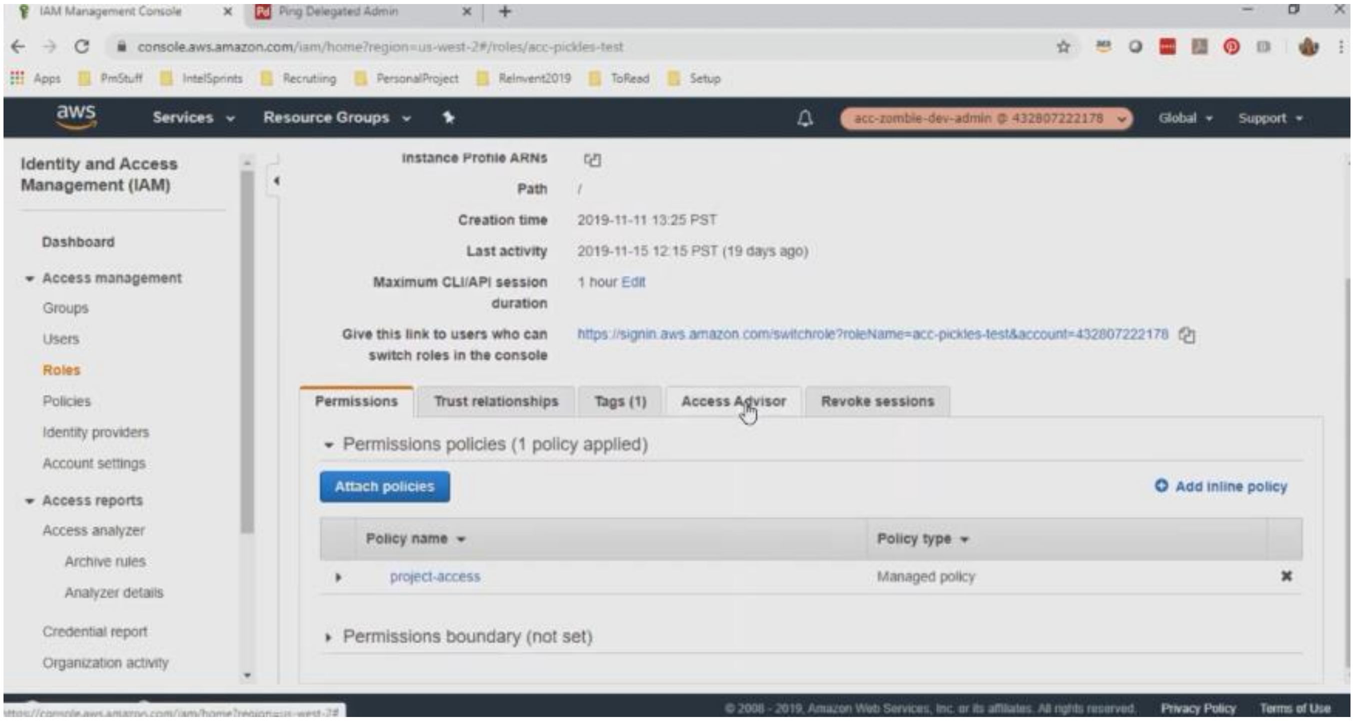
Rein in permissions: Demo steps

1. Analyze **roles** last used timestamp and delete those older than 6 months
2. Analyze developer **role policies** and identify unused services
3. Analyze **SCPs** to identify unused services



Showing 49 results

Role name	Trusted entities	Last activity
AWSServiceRoleForOrganizations	**AWS service:** organizations (Service-Linked r...	69 days
ABACSetUP-ExampleHostRole-1U8O5F...	**AWS service:** ec2	133 days
serverlessrepo-SecretsMan-SecretsMana...	**AWS service:** lambda	133 days
AWSServiceRoleForTrustedAdvisor	**AWS service:** trustedadvisor (Service-Linked ...	137 days
bubbles_create_manage_secrets	**Account:** 432807222178	159 days
serverlessrepo-SecretsMan-SecretsMana...	**AWS service:** lambda	163 days
zombies-dev-admin	**Account:** 332207979596	379 days
IsengardCloudTrailRole	**Account:** 086441151436	380 days
abacDemoDBTest	**AWS service:** ec2	None
AWSServiceRoleForSSO	**AWS service:** sso (Service-Linked role)	None
AWSServiceRoleForSupport	**AWS service:** support (Service-Linked role)	None
GatedGardenAudit	**Account:** 638951966944	None



Instance Profile ARNs

Path: /

Creation time: 2019-11-11 13:25 PST

Last activity: 2019-11-15 12:15 PST (19 days ago)

Maximum CLI/API session duration: 1 hour Edit

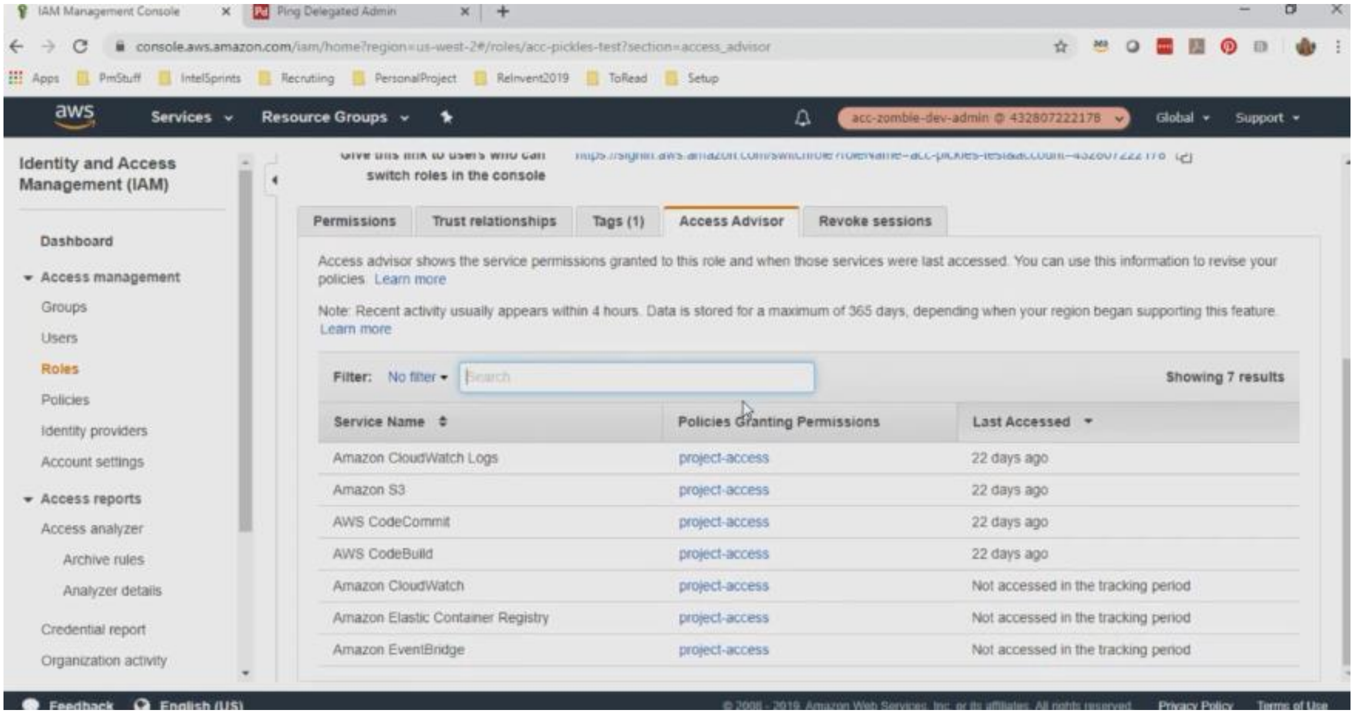
Give this link to users who can switch roles in the console: https://signin.aws.amazon.com/switchrole?roleName=acc-pickles-test&account=432807222178

Permissions policies (1 policy applied)

Policy name	Policy type
project-access	Managed policy

Permissions boundary (not set)

© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use



switch roles in the console

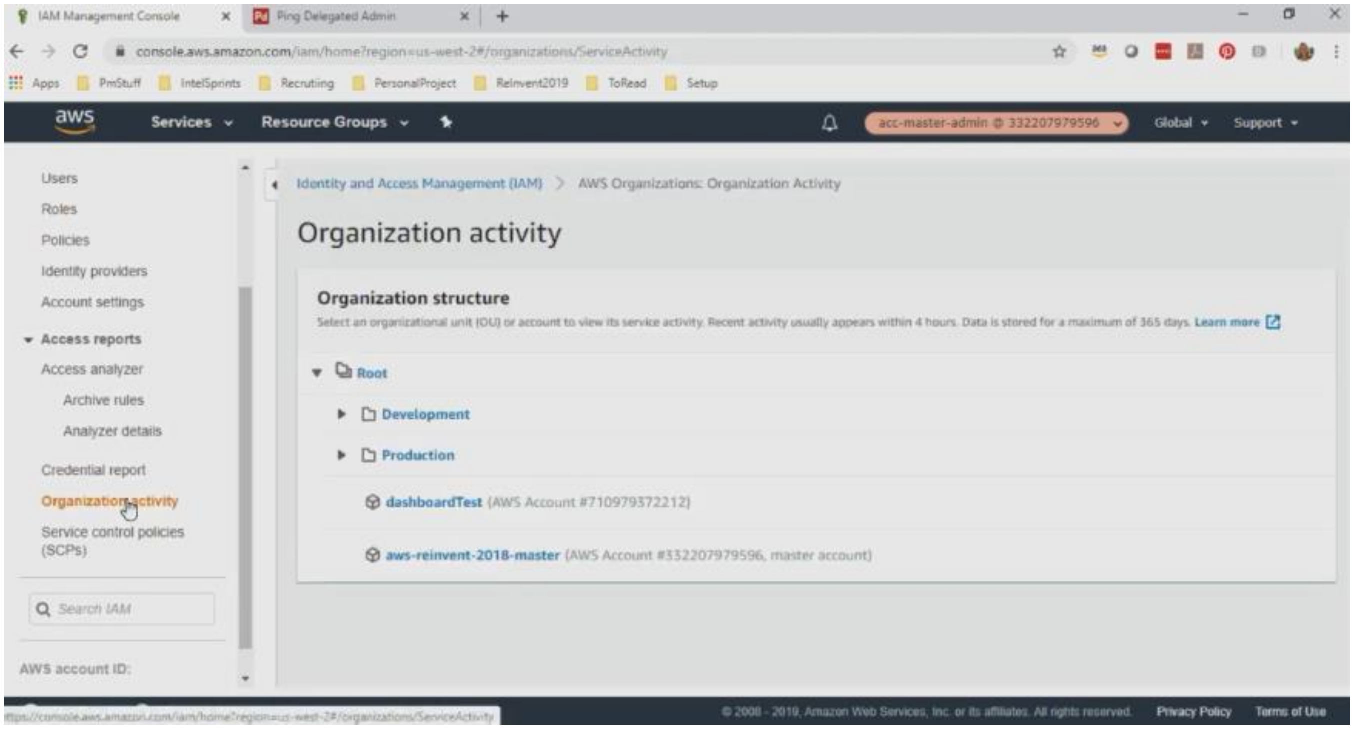
Permissions | Trust relationships | Tags (1) | Access Advisor | Revoke sessions

Access advisor shows the service permissions granted to this role and when those services were last accessed. You can use this information to revise your policies. Learn more

Note: Recent activity usually appears within 4 hours. Data is stored for a maximum of 365 days, depending when your region began supporting this feature. Learn more

Filter: No filter Search — Showing 7 results

Service Name	Policies Granting Permissions	Last Accessed
Amazon CloudWatch Logs	project-access	22 days ago
Amazon S3	project-access	22 days ago
AWS CodeCommit	project-access	22 days ago
AWS CodeBuild	project-access	22 days ago
Amazon CloudWatch	project-access	Not accessed in the tracking period
Amazon Elastic Container Registry	project-access	Not accessed in the tracking period
Amazon EventBridge	project-access	Not accessed in the tracking period



Identity and Access Management (IAM) > AWS Organizations: Organization Activity

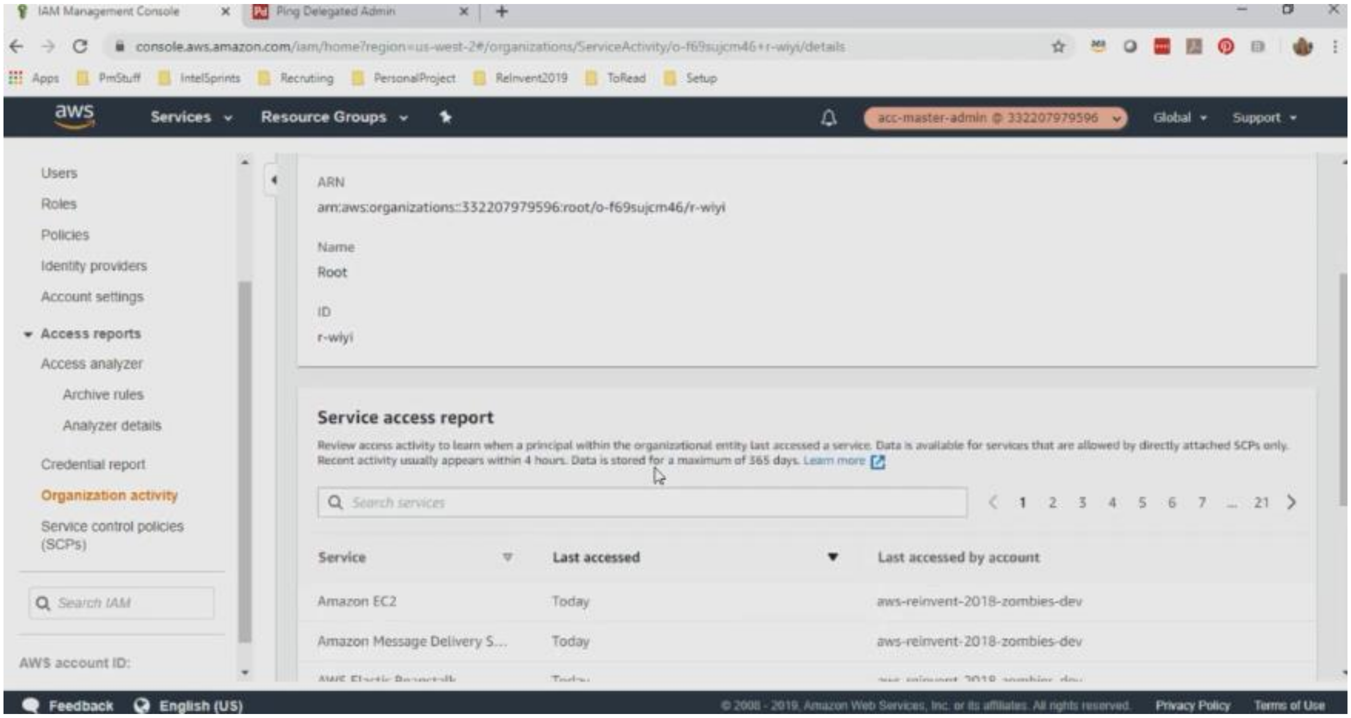
Organization activity

Organization structure

Select an organizational unit (OU) or account to view its service activity. Recent activity usually appears within 4 hours. Data is stored for a maximum of 365 days. Learn more

- Root
  - Development
  - Production
  - dashboardTest (AWS Account #710979372212)
  - aws-reinvent-2018-master (AWS Account #332207979596, master account)

This will allow you to set your SCP



ARN
arn:aws:organizations::332207979596:root/o-f69sujcm46/r-wiyi

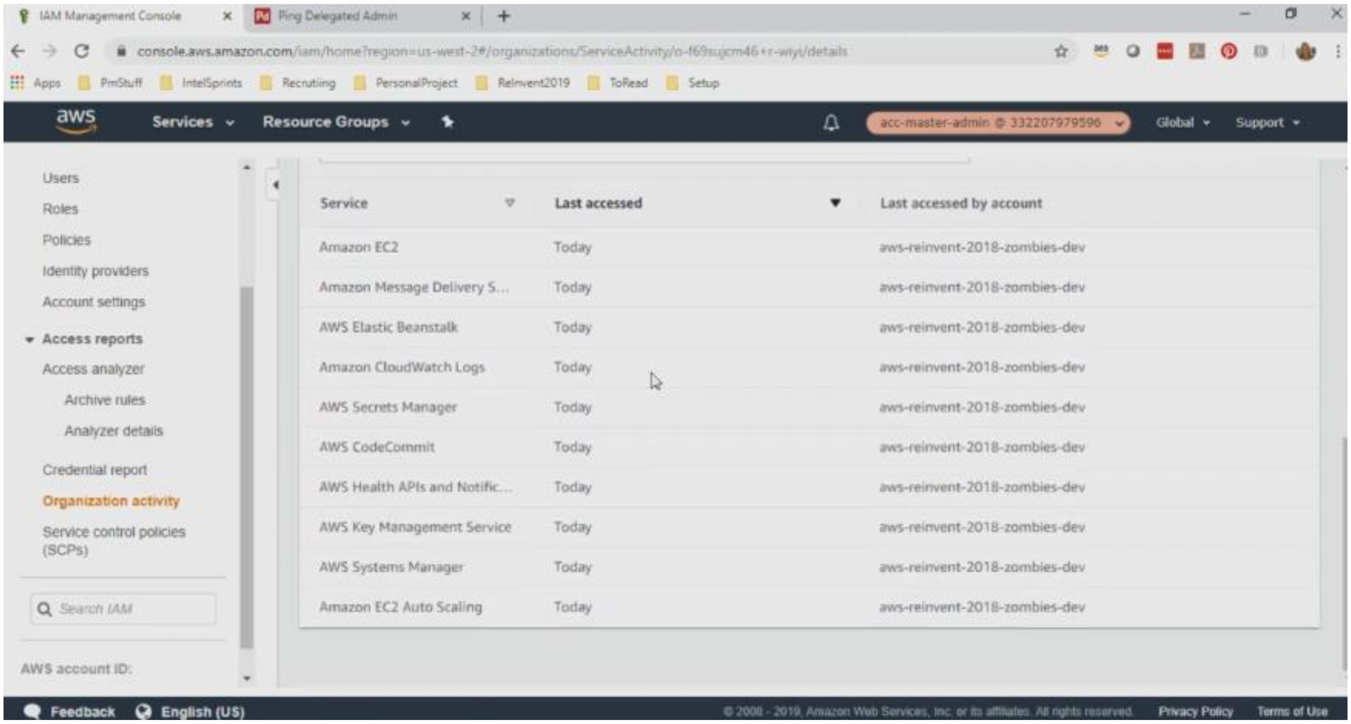
Name
Root

ID
r-wiyi

Service access report

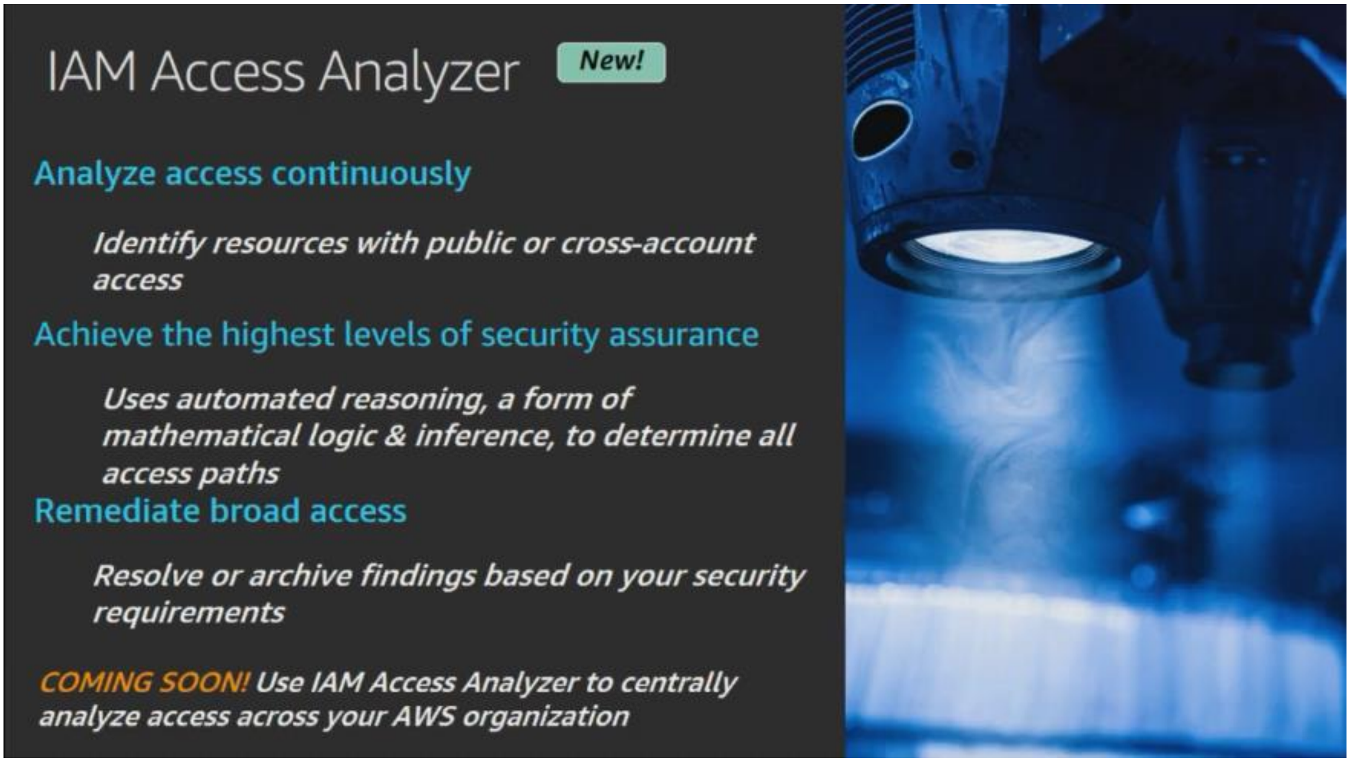
Review access activity to learn when a principal within the organizational entity last accessed a service. Data is available for services that are allowed by directly attached SCPs only. Recent activity usually appears within 4 hours. Data is stored for a maximum of 365 days. Learn more

Service	Last accessed	Last accessed by account
Amazon EC2	Today	aws-reinvent-2018-zombies-dev
Amazon Message Delivery S...	Today	aws-reinvent-2018-zombies-dev



Service	Last accessed	Last accessed by account
Amazon EC2	Today	aws-reinvent-2018-zombies-dev
Amazon Message Delivery S...	Today	aws-reinvent-2018-zombies-dev
AWS Elastic Beanstalk	Today	aws-reinvent-2018-zombies-dev
Amazon CloudWatch Logs	Today	aws-reinvent-2018-zombies-dev
AWS Secrets Manager	Today	aws-reinvent-2018-zombies-dev
AWS CodeCommit	Today	aws-reinvent-2018-zombies-dev
AWS Health APIs and Notific...	Today	aws-reinvent-2018-zombies-dev
AWS Key Management Service	Today	aws-reinvent-2018-zombies-dev
AWS Systems Manager	Today	aws-reinvent-2018-zombies-dev
Amazon EC2 Auto Scaling	Today	aws-reinvent-2018-zombies-dev

This is a list for what every service principals uses



IAM Access Analyzer New!

Analyze access continuously

Identify resources with public or cross-account access

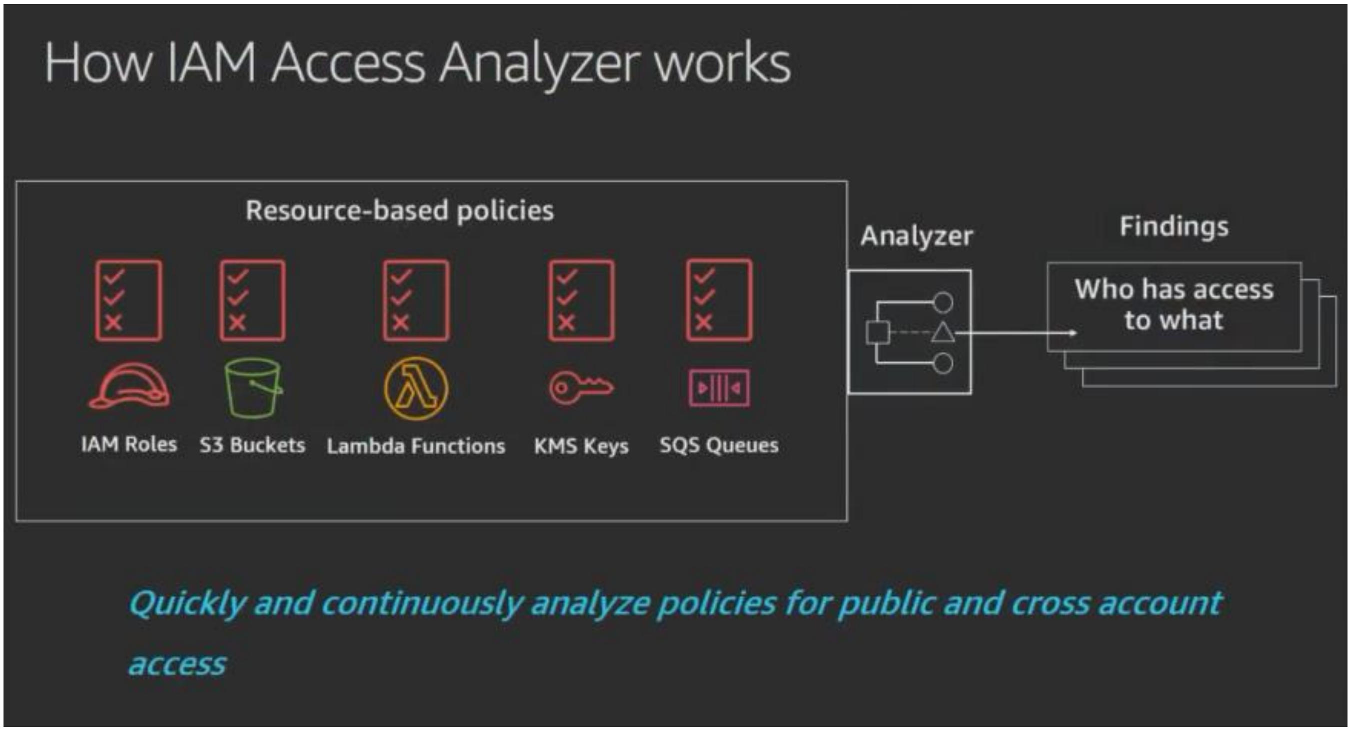
Achieve the highest levels of security assurance

Uses automated reasoning, a form of mathematical logic & inference, to determine all access paths

Remediate broad access

Resolve or archive findings based on your security requirements

COMING SOON! Use IAM Access Analyzer to centrally analyze access across your AWS organization



How IAM Access Analyzer works

Resource-based policies

IAM Roles

S3 Buckets

Lambda Functions

KMS Keys

SQS Queues

Analyzer

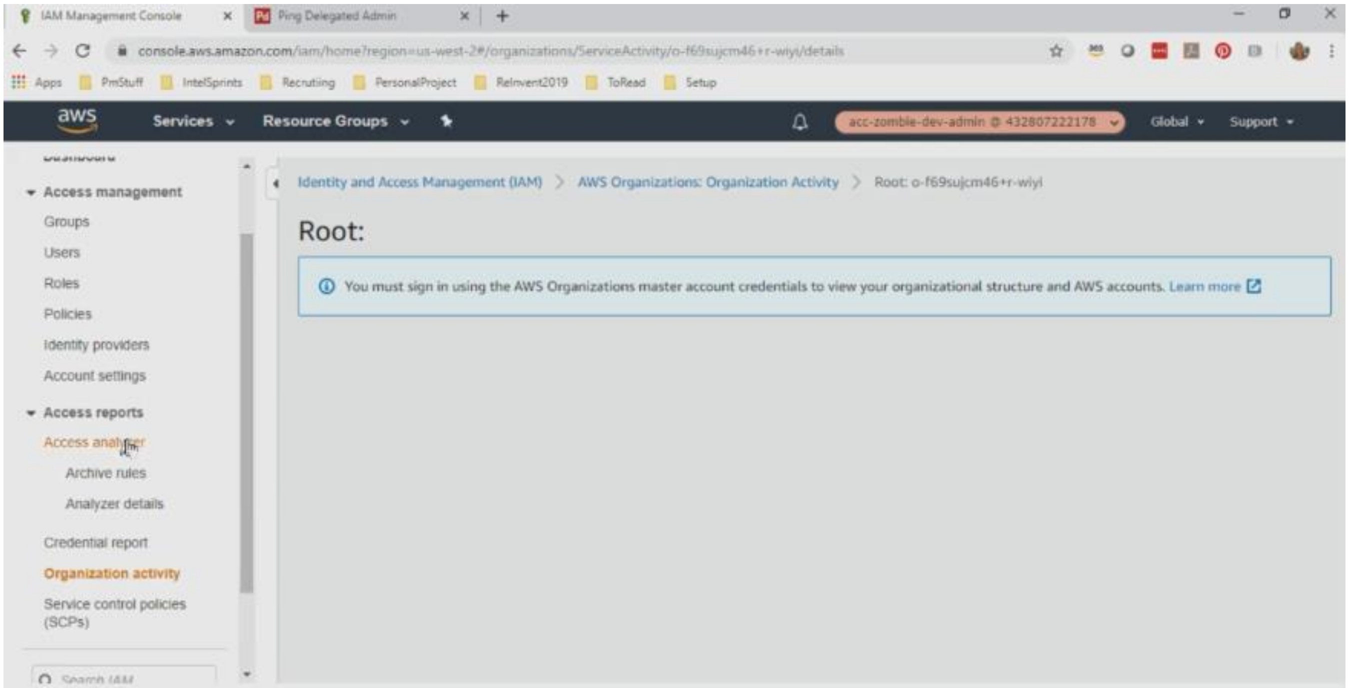
Findings

Who has access to what

Quickly and continuously analyze policies for public and cross account access

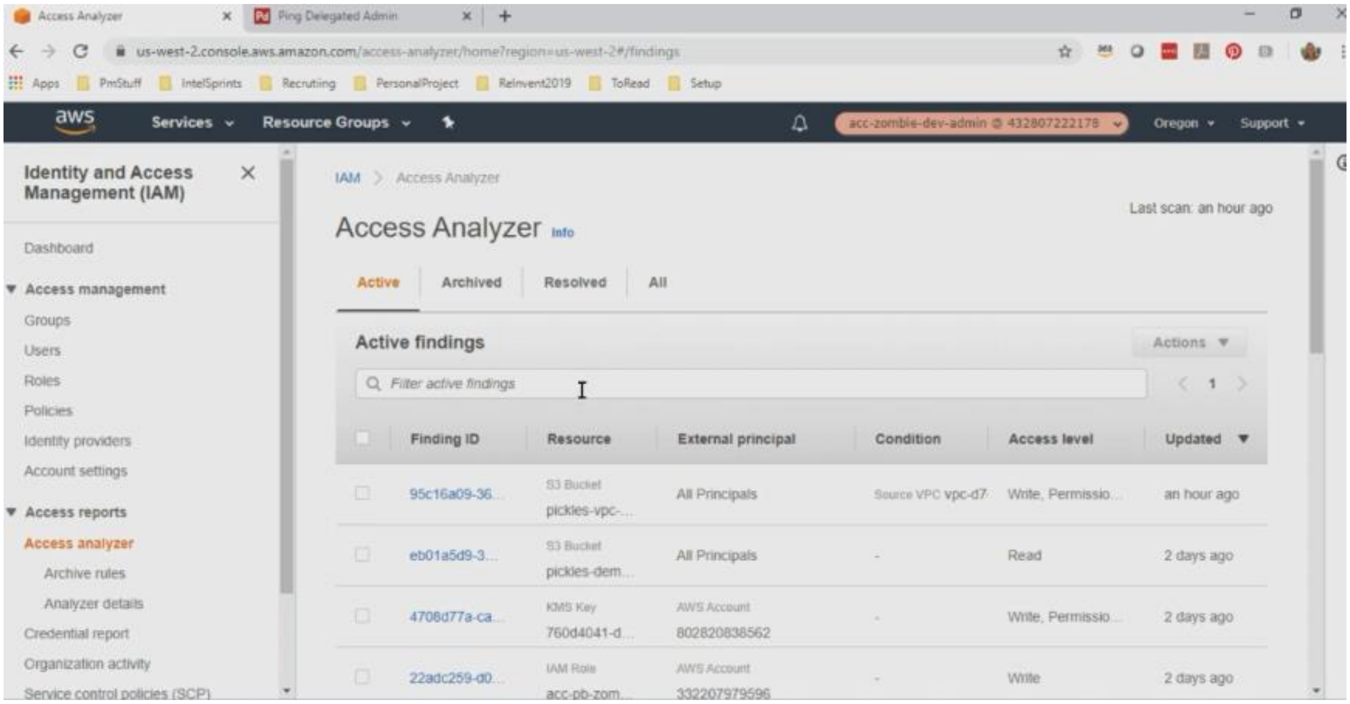
Let's analyze some access

1. Visit Access Analyzer in the IAM console
2. See the finding for a bucket with broad permissions
3. Determine our next step



Root:

You must sign in using the AWS Organizations master account credentials to view your organizational structure and AWS accounts. Learn more

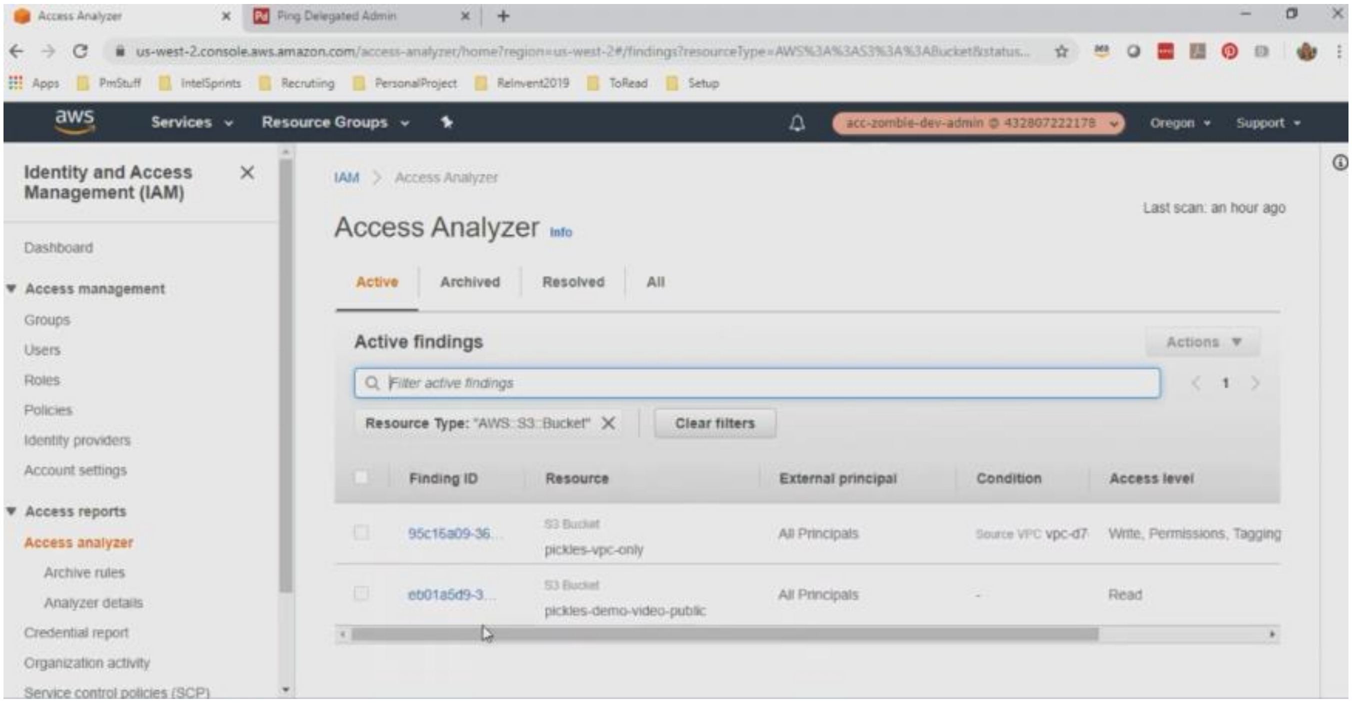


Access Analyzer

Last scan: an hour ago

Active findings

□	Finding ID	Resource	External principal	Condition	Access level	Updated
□	95c16a09-36...	S3 Bucket pickles-vpc-...	All Principals	Source VPC vpc-d7	Write, Permissio...	an hour ago
□	eb01a5d9-3...	S3 Bucket pickles-dem...	All Principals	-	Read	2 days ago
□	4708d77a-ca...	KMS Key 760d4041-d...	AWS Account 802820838562	-	Write, Permissio...	2 days ago
□	22adc259-d0...	IAM Role acc-pb-zom...	AWS Account 332207979596	-	Write	2 days ago



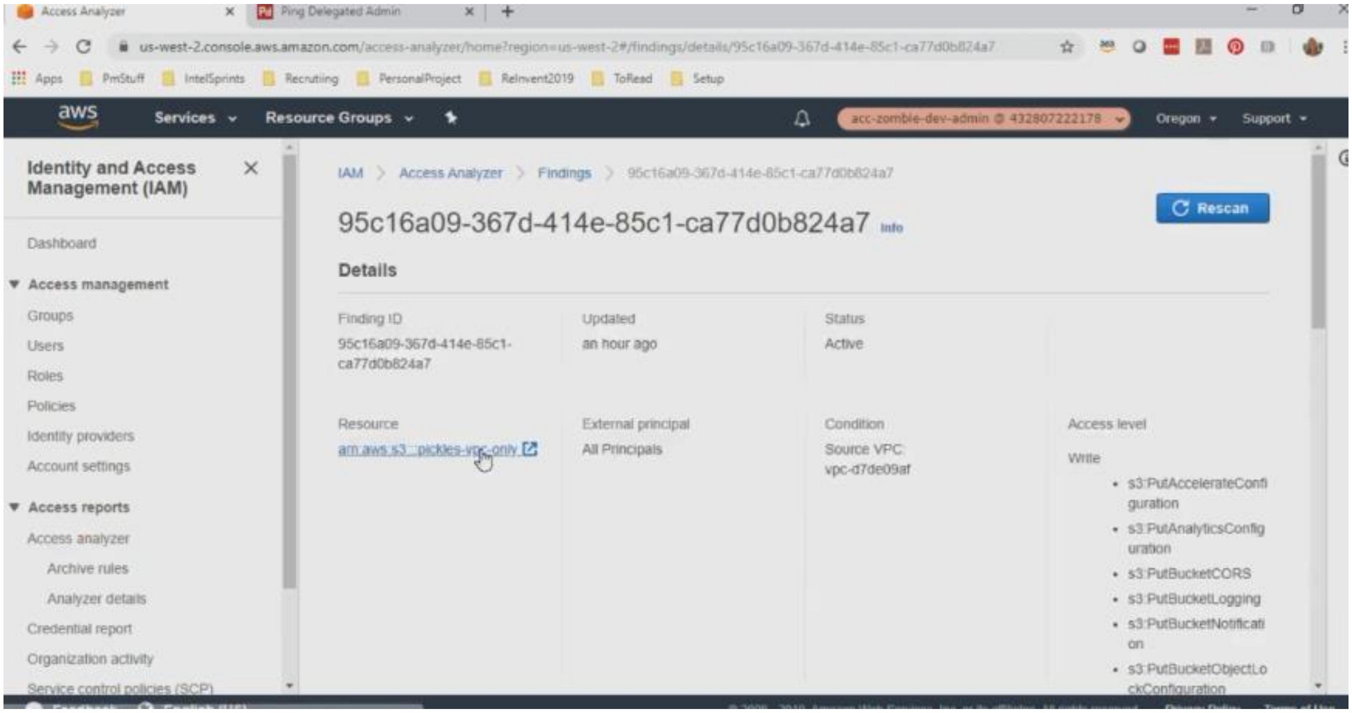
Access Analyzer

Last scan: an hour ago

Active findings

Resource Type: "AWS::S3::Bucket"

Finding ID	Resource	External principal	Condition	Access level
95c16a09-36...	S3 Bucket pickles-vpc-only	All Principals	Source VPC vpc-d7	Write, Permissions, Tagging
eb01a5d9-3...	S3 Bucket pickles-demo-video-public	All Principals	-	Read



95c16a09-367d-414e-85c1-ca77d0b824a7

Details

Finding ID: 95c16a09-367d-414e-85c1-ca77d0b824a7

Updated: an hour ago

Status: Active

Resource: arn:aws:s3:::pickles-vpc-only

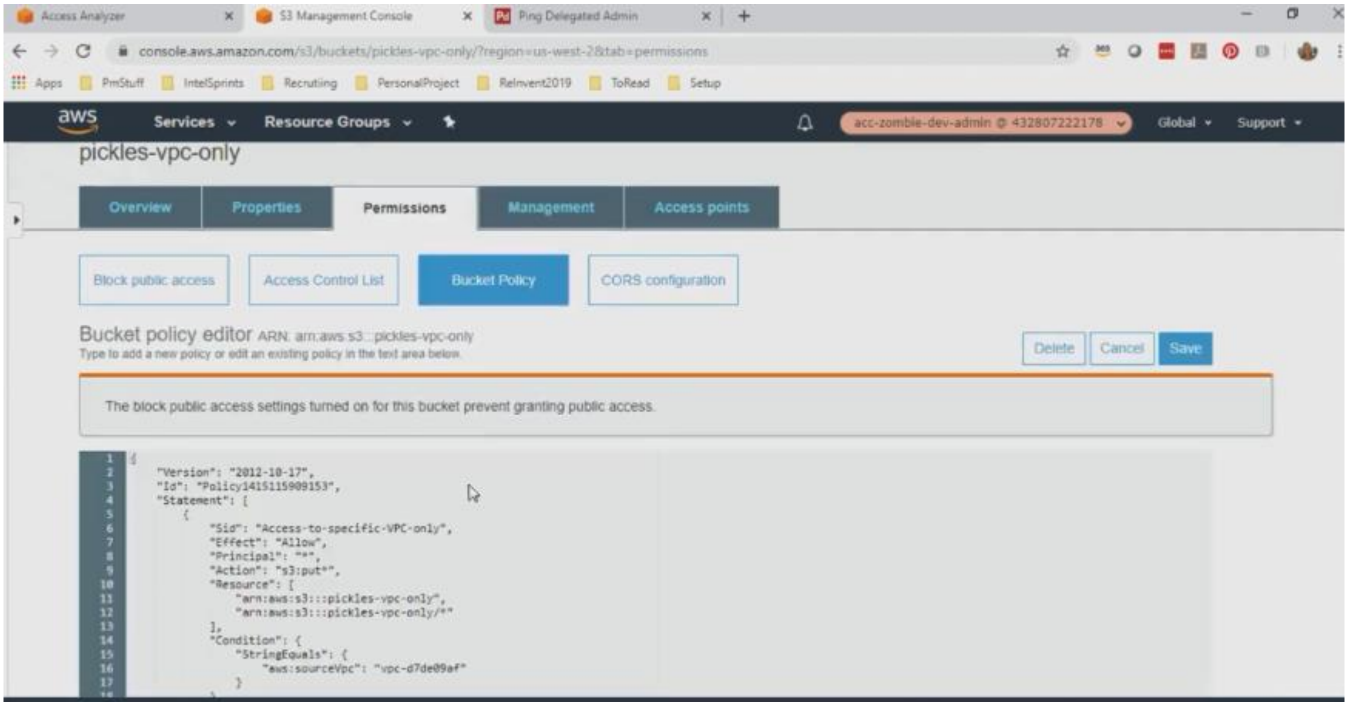
External principal: All Principals

Condition: Source VPC: vpc-d7de09af

Access level:

Write

- s3:PutAccelerateConfiguration
- s3:PutAnalyticsConfiguration
- s3:PutBucketCORS
- s3:PutBucketLogging
- s3:PutBucketNotification
- s3:PutBucketObjectLockConfiguration



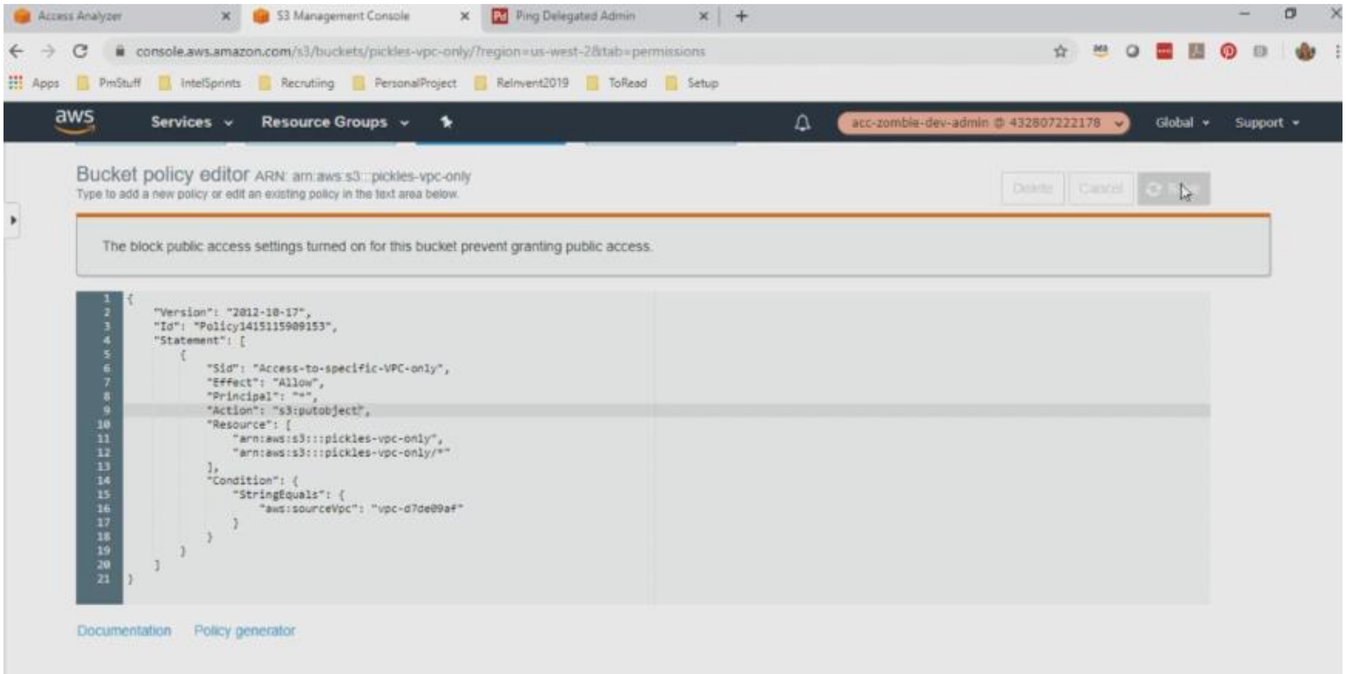
pickles-vpc-only

Bucket policy editor ARN: arn:aws:s3:::pickles-vpc-only

Type to add a new policy or edit an existing policy in the text area below.

The block public access settings turned on for this bucket prevent granting public access.

```
{
    "Version": "2012-10-17",
    "Id": "Policy1415115909153",
    "Statement": [
        {
            "Sid": "Access-to-specific-VPC-only",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:put*",
            "Resource": [
                "arn:aws:s3:::pickles-vpc-only",
                "arn:aws:s3:::pickles-vpc-only/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:sourceVpc": "vpc-d7de09af"
                }
```



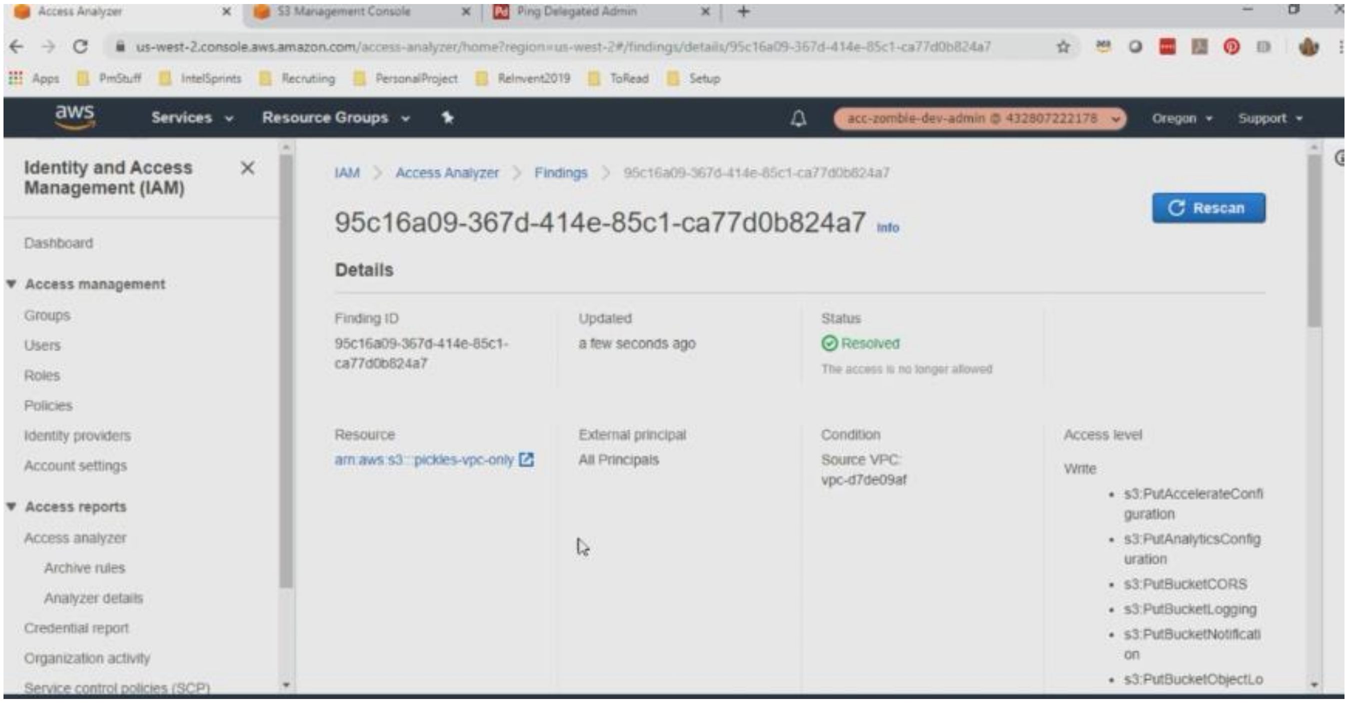
Bucket policy editor ARN: arn:aws:s3:::pickles-vpc-only

Type to add a new policy or edit an existing policy in the text area below.

The block public access settings turned on for this bucket prevent granting public access.

```
{
    "Version": "2012-10-17",
    "Id": "Policy1415115909153",
    "Statement": [
        {
            "Sid": "Access-to-specific-VPC-only",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:putobject",
            "Resource": [
                "arn:aws:s3:::pickles-vpc-only",
                "arn:aws:s3:::pickles-vpc-only/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:sourceVpc": "vpc-d7de09af"
                }
            }
        }
    ]
}
```

We change the put/* to putObject, then do a re-scan to check if its fixed



95c16a09-367d-414e-85c1-ca77d0b824a7

Details

Finding ID: 95c16a09-367d-414e-85c1-ca77d0b824a7

Updated: a few seconds ago

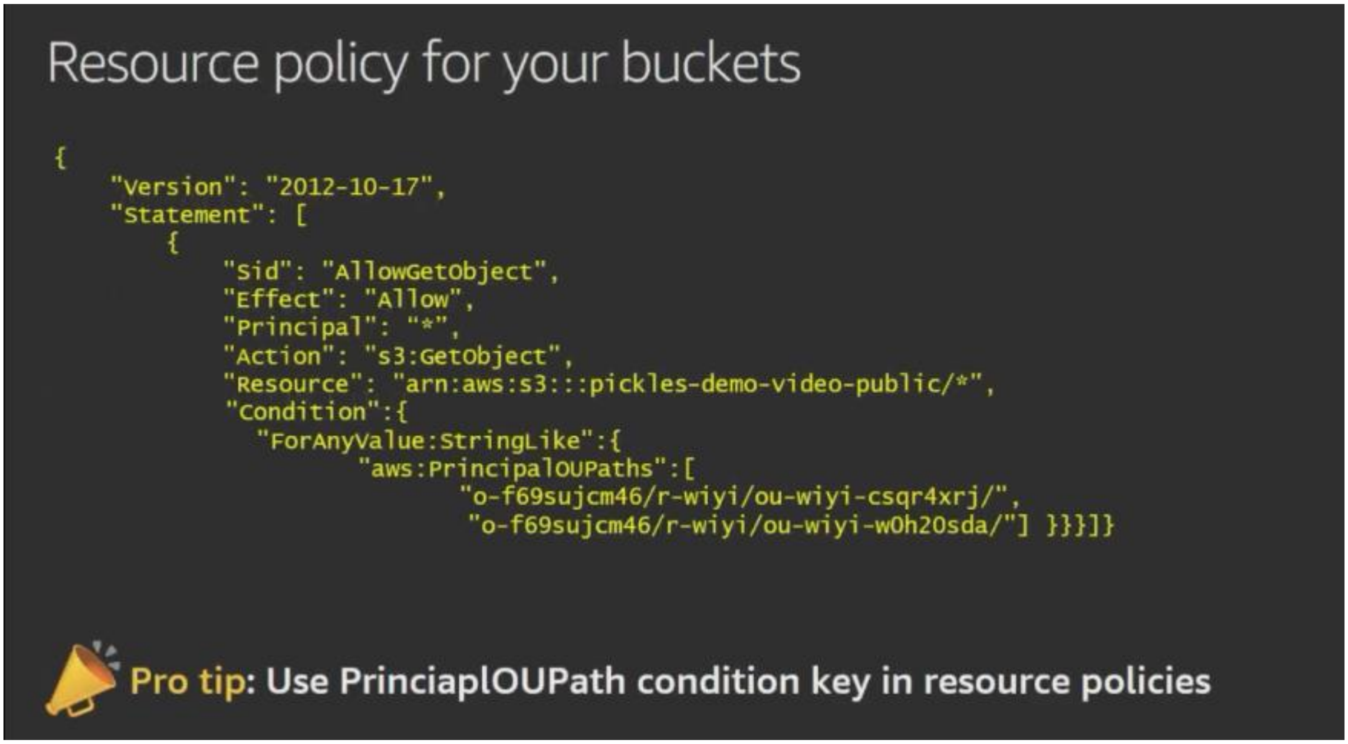
Status: Resolved
The access is no longer allowed

Resource: arn:aws:s3:::pickles-vpc-only

External principal: All Principals

Condition: Source VPC: vpc-d7de09af

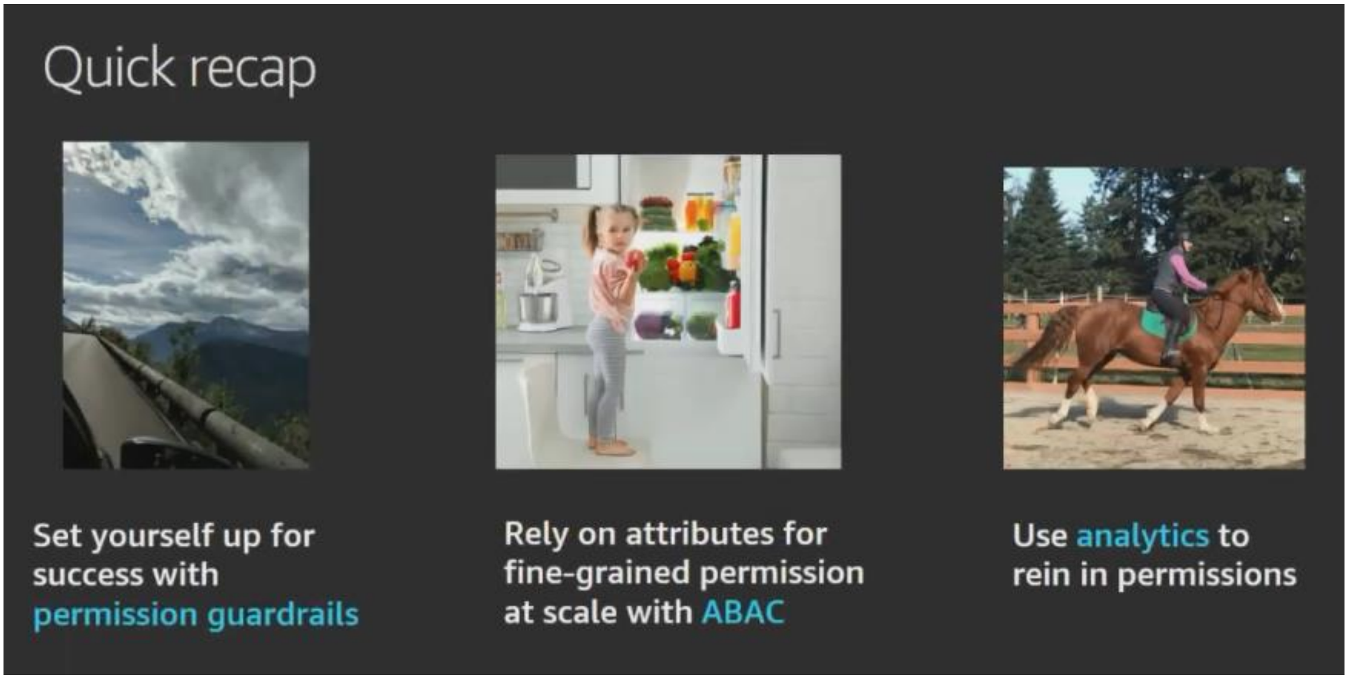
Access level: Write
- s3:PutAccelerateConfiguration
- s3:PutAnalyticsConfiguration
- s3:PutBucketCORS
- s3:PutBucketLogging
- s3:PutBucketNotification
- s3:PutBucketObjectLo



Resource policy for your buckets

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowGetObject",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::pickles-demo-video-public/*",
            "Condition":{
              "ForAnyValue:StringLike":{
                     "aws:PrincipalOUPaths":[
                            "o-f69sujcm46/r-wiyi/ou-wiyi-csqr4xrj/",
                             "o-f69sujcm46/r-wiyi/ou-wiyi-w0h20sda/"] }}}]}
```

Pro tip: Use PrinciaplOUPath condition key in resource policies

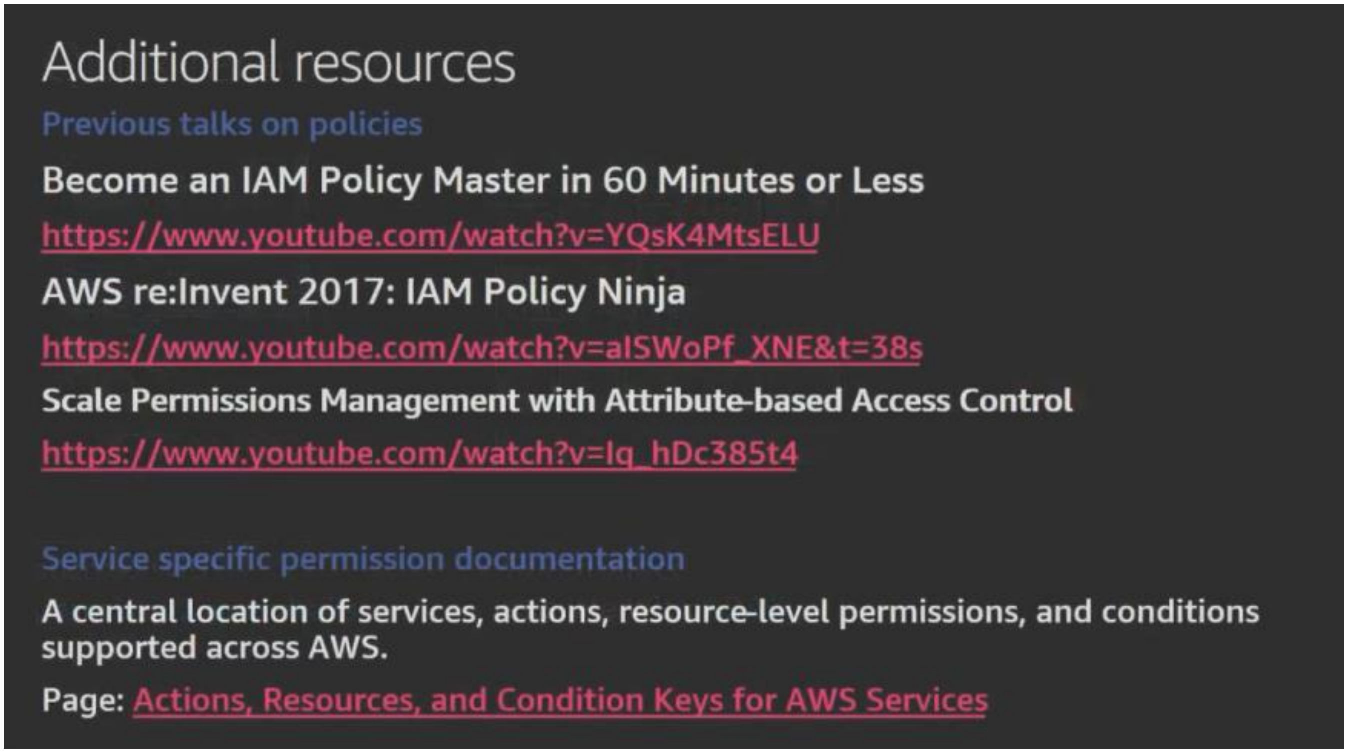


Quick recap

Set yourself up for success with permission guardrails

Rely on attributes for fine-grained permission at scale with ABAC

Use analytics to rein in permissions



Additional resources

Previous talks on policies

Become an IAM Policy Master in 60 Minutes or Less

https://www.youtube.com/watch?v=YQsK4MtsELU

AWS re:Invent 2017: IAM Policy Ninja

https://www.youtube.com/watch?v=aISWoPf_XNE&t=38s

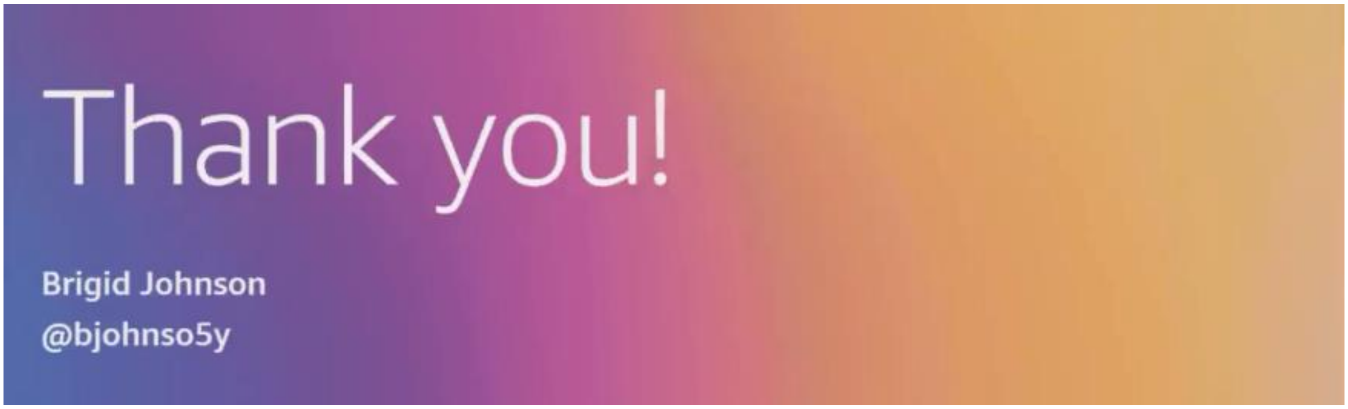
Scale Permissions Management with Attribute-based Access Control

https://www.youtube.com/watch?v=Iq_hDc385t4

Service specific permission documentation

A central location of services, actions, resource-level permissions, and conditions supported across AWS.

Page: Actions, Resources, and Condition Keys for AWS Services



Thank you!

Brigid Johnson
@bjohnso5y