

Become an IAM Policy Master in 60 Minutes or Less

Brigid Johnson
Senior Manager of Product Management
AWS Identity



© 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.



Are you interested in becoming an IAM policy master and learning about powerful techniques for controlling access to AWS resources? If your answer is “yes,” this session is for you. Join us as we cover the different types of policies and describe how they work together to control access to resources in your account and across your AWS organization. We walk through use cases that help you delegate permission management to developers by demonstrating IAM permission boundaries. We take an in-depth look at controlling access to specific AWS regions using condition keys. Finally, we explain how to use tags to scale permissions management in your account. This session requires you to know the basics of IAM policies.

Agenda

- ⚡ Recap of IAM policy language
- ⚡ Policy types and how they work together
- ⚡ Deep dive on policy with specific use cases
 - Set permission guardrails across accounts
 - Control creation of resources to specific regions
 - Enable developers to create roles safely
 - Use tags to scale permissions management

Related breakouts

Tuesday, November 27th

Sec301 - The Theory and Math Behind Data Privacy and Security Assurance
10:00 am- 11:00 am Venetian, Level 2, Titian 2204

Wednesday, Nov 28, 11:30 AM - 12:30 PM

IAM for Enterprises: How Vanguard Matured IAM Controls to Support Micro Accounts
11:30 AM - 12:30 PM Mirage, Grand Ballroom F

Recap of IAM policy language

Quick recap of IAM policies

- What are IAM policies?
- IAM policy structure
- IAM policy evaluation rules
- How to think about IAM policy evaluation (new!)

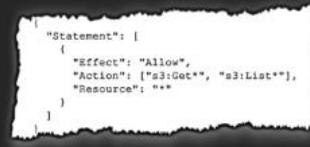
What are IAM policies?

Policies provide authorization to AWS services and resources

Two parts:

Specification: *Defining* access policies

Enforcement: *Evaluating* policies



When you *define* access policies. You specify which IAM **principals** are allowed to perform which **actions** on specific AWS **resources** and under which **conditions**.

IAM enforces this access by *evaluating* the AWS request and the policies you defined and returns either yes or no answer.

IAM policy structure

```
{  
  "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 access

`"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 under the access defined is valid

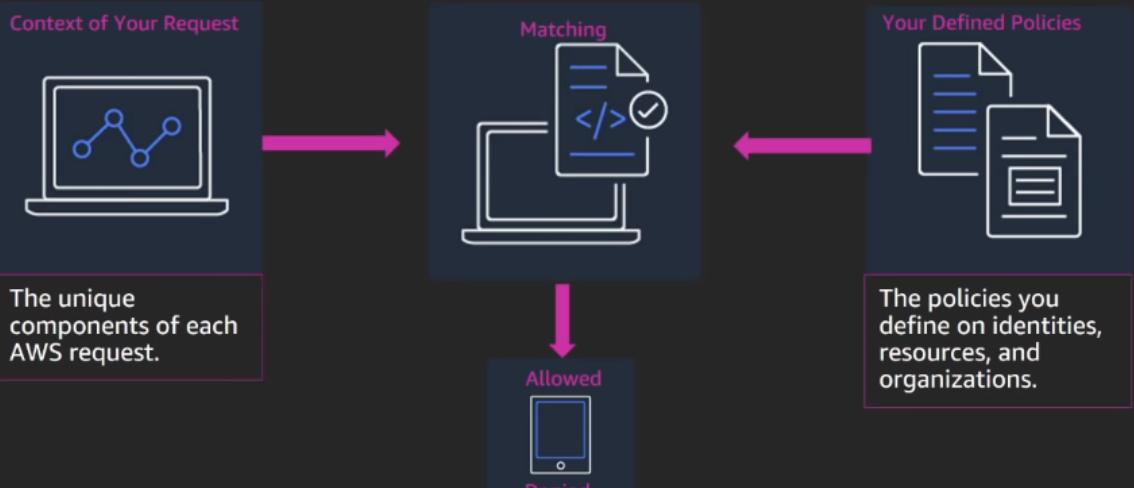
`"StringEqualsIfExists": {"aws:RequestTag/project": ["Pickles"]}`

IAM policy evaluation



Policy enforcement

Context and policies – a new way to think about evaluation



Policy types and how they work together

Policy types and core use cases

AWS Organizations

Service control policies (SCPs)

*Guardrails to disable service access
on the principals in the account*

AWS Identity and Access Management (IAM)

**As Permission Policies and
Permission Boundaries**

*Grant granular permissions on IAM
principals (users and roles) and control
the maximum permissions they can set*

AWS Security Token Service (AWS STS)

Scoped-down policies

*Reduce general shared permissions
further*

Specific AWS services

Resource-based policies

*Cross-account access and to
control access from the resource*

VPC Endpoints

Endpoint Policies

*Controls access to the service with
a VPC endpoint*

All use the same policy language

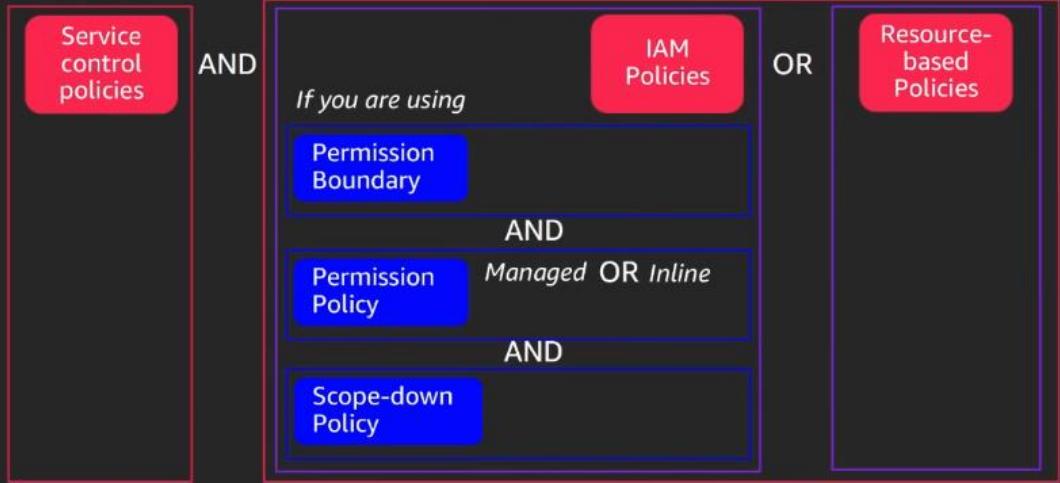
Permission boundaries – new this year!

Scale and delegate permission management to developers safely

Control the maximum permissions employees can grant

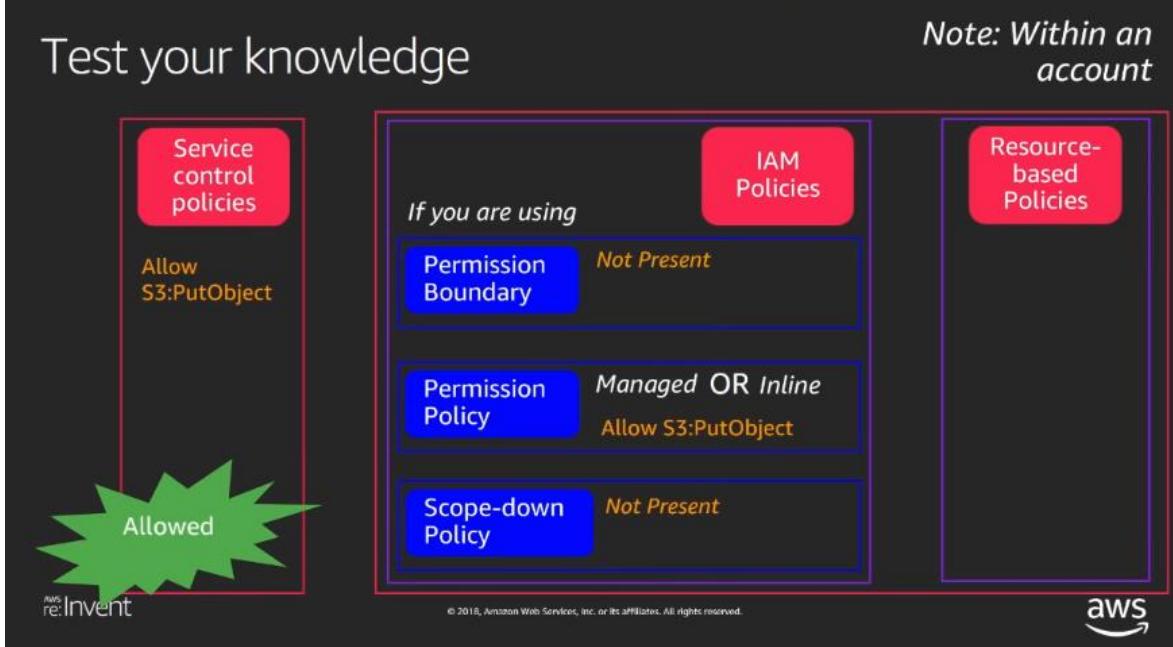


How policies work together *within an account*



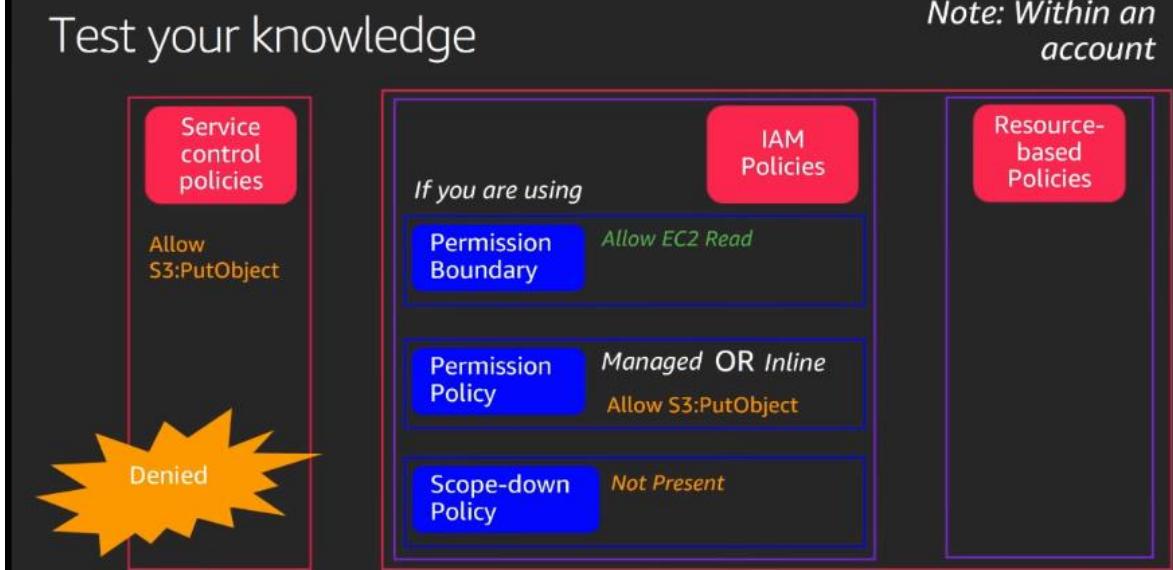
Test your knowledge

Note: Within an account



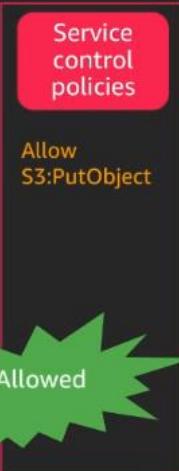
Test your knowledge

Note: Within an account



Test your knowledge

Note: Within an account

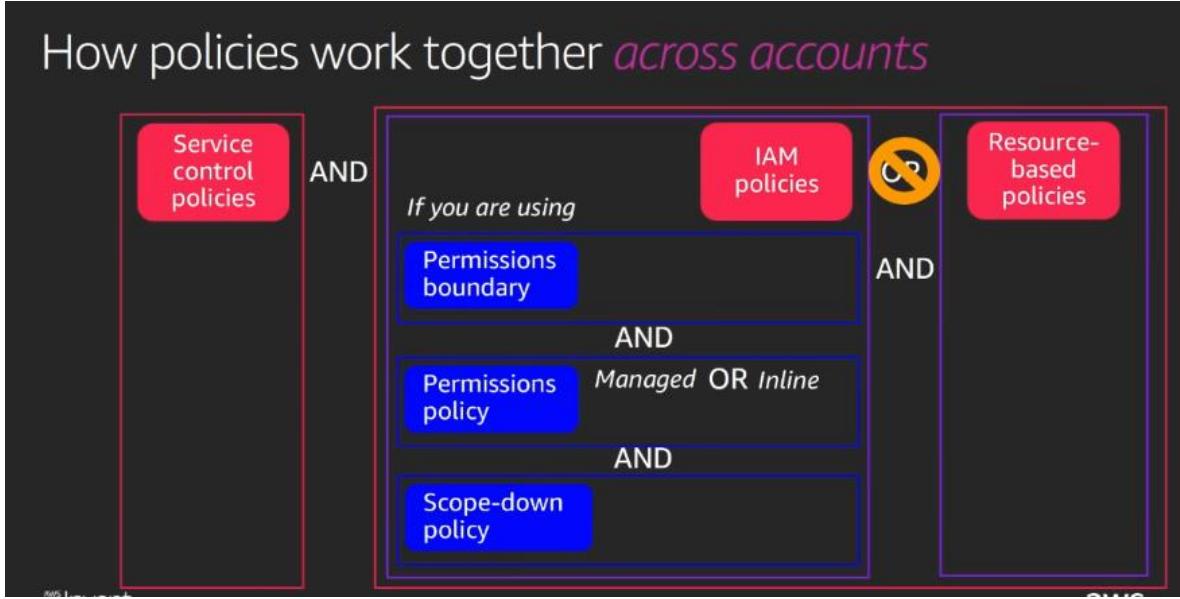


Test your knowledge

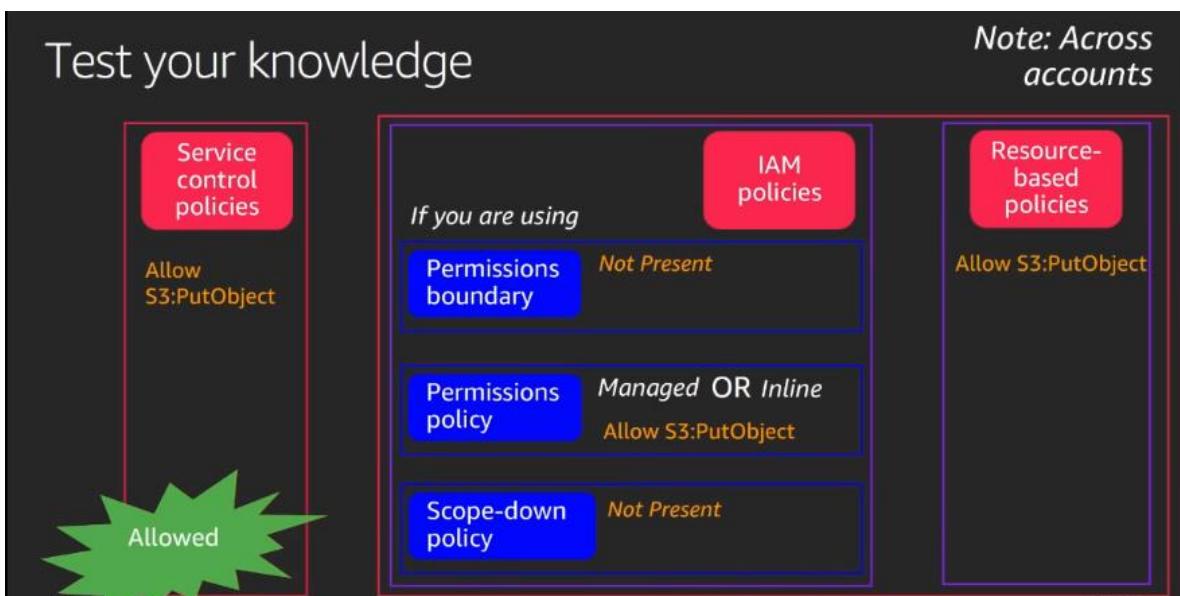
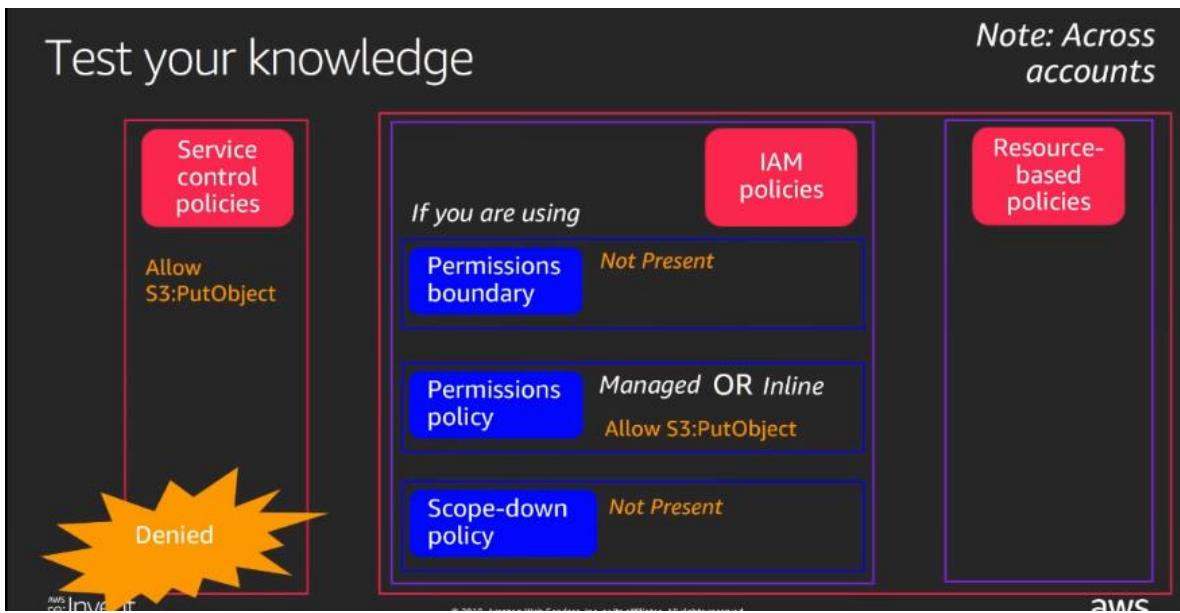
Note: Within an account



How policies work together *across accounts*



Both sides now have to allow for it to hold



Deep dive on policy with specific use cases

Congratulations – you just got a new job!



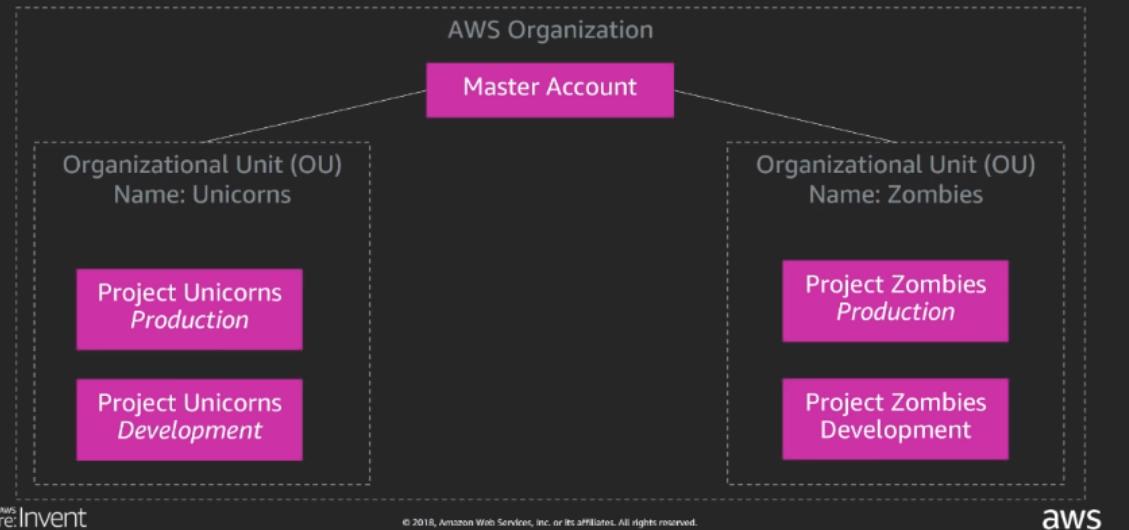
Your new position

Let's imagine you are now the lead of a central security team

Your first mission

Prevent developers from reverting setting in your AWS accounts and onboard a new two new teams!

AWS Organization and account structure



Services your organization uses



Amazon EC2 to run workloads



AWS Lambda for serverless applications



AWS Secrets Manager to store secrets for database access and third party API keys



Amazon S3 to store content objects

#1 - Set permission guardrails across accounts

#1 - Situation

Your team has gone through and set up AWS CloudTrail in all accounts. Your company also requires users to authenticate with their existing identity provider.

#1 - Challenge

Ensure developers cannot turn off CloudTrail, create IAM users, or set up AWS Directory Service.

#2 - Control creation of resources to specific regions

#2 - Situation

You've learned that you can trust your development team to create resources in AWS, however your leadership is concerned about creating resources in unapproved regions.

#2 - Challenge

Ensure your developers can create resources, but only in approved regions.

#3 - Enable developers to create roles safely

#3 - Situation

Your developers know their stuff! They mentioned they can build on AWS more quickly if they can create their own roles without going through your central security team.

#3 - Challenge

Enable your developers to create IAM roles to pass to Amazon EC2 and AWS Lambda, but ensure they cannot exceed their own permissions.

#4 - Use tags to scale permissions management

#4 - Situation

The Unicorns project has been split into two projects. Dorky Unicorns and Sneaky Unicorns. They still share an account and keep stepping on each other toes.

#4 - Challenge

Update permissions to enable developers working on Dorky Unicorns and Sneaky Unicorns to manage their own resources without managing the other project's.

Match the tool to use for each challenge



Challenge #1

Ensure developers cannot turn off CloudTrail, create IAM users, or set up AWS Directory Service.

Pro Tip: Rely on deny statements when restricting access to accounts to reduce blast radius.

We will use a service control policy SCP.

SCP for challenge #1

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "DenyUnapprovedAction",  
            "Effect": "Deny",  
            "Action": [  
                "ds:*",  
                "iam:CreateUser",  
                "cloudtrail:StopLogging"  
            ],  
            "Resource": [  
                "*"  
            ]  
        }  
    ]  
}
```

re:Invent

© 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Don't
Forget!

We also have
an Allow '*'
policy
attached to
this OU



aws

Let's see SCPs in action

- Show SCP to deny access to modify AWS CloudTrail, Create IAM users, and AWS Directory Services
- Use the CLI as an administrator in the Unicorn-dev account and try to:
 - Create an IAM user
 - List roles
 - Stop logging in AWS CloudTrail



aws

re:Invent

© 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.

This screenshot shows the AWS Organizations console in dark skin mode. The main interface includes a navigation bar with 'Services' and 'Resource Groups', and a top bar with account information and support links. The main content area displays a table of accounts, each with columns for Account name, Email, Account ID, and Status. A modal window titled 'No selection' is open, prompting the user to 'Please select an account to see more details'. At the bottom, there are links for 'Feedback', 'English (US)', and legal notices.

This dark skin mode is our master account

This screenshot shows the IAM Management Console in dark skin mode. The left sidebar lists various IAM management options like Dashboard, Groups, Users, Roles, Policies, and Identity providers. The main content area shows a search results page for roles named 'challenge'. The results table includes columns for Role name, Description, and Trusted entities. Each result row contains a checkbox and a link to the role's details.

This light skin mode is our unicorn's dev account where we will see a bunch of the activities

This screenshot shows the AWS Management Console in light skin mode. The top navigation bar includes 'Services' and 'Resource Groups'. The main content area features a large header 'AWS Management Console'. Below it, there are two main sections: 'AWS services' on the left and 'Access resources on the go' and 'Explore AWS' on the right. The 'AWS services' section has a search bar and a list of recently visited services like EC2, IAM, and Lambda. The 'Access resources on the go' section promotes the AWS Console Mobile App. The 'Explore AWS' section highlights Amazon Redshift and AWS Fargate.

This Firefox page will be the developer that will assume different roles for each challenge

Accounts Organize accounts Policies

Add account Remove account Hide Failed account creation requests

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

No selection

Please select an account to see more details

Feedback English (US)

AWS Organizations

Root

TREE VIEW Filter

Organizational units (2)

- Unicorns
- Zombies

Accounts (1)

- aws-reinvent-201...

Root

ARN: arn:aws:organizations:332207979596:root/o-f69sjcm46ri-wily

POLICIES

Service control policies

ENABLE / DISABLE POLICY TYPES

Service control policies Disable

We have attached the **DenyUnapprovedActions** policy to the root of our master account here, we can also see the 2 organizational units OUs under this master account.

Create policy Delete policy

Policy name	Policy type	Description
FullAWSAccess	Service control	Allows access to every operation
DenyUnapprov...	Service control	This prevents unapproved actions across my organization

No selection

Please select a policy to see more details

Isengard AWS Organizations https://console.aws.amazon.com/organizations/home?region=us-east-1#/policies/p-ikjawcfn

Services Resource Groups

AWS Organizations

Accounts Organize accounts Policies Invitations Settings

All policies > DenyUnapprovedActions

Details

Name: DenyUnapprovedActions

Description: This prevents unapproved actions across my organization

ID: p-ikjawcfn

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

Feedback English (US) Privacy Policy Terms of Use

Isengard AWS Organizations https://console.aws.amazon.com/organizations/home?region=us-east-1#/policies

Services Resource Groups

AWS Organizations

Accounts Organize accounts Policies Invitations Settings

Create policy Delete policy

Policy name	Policy type	Description
FullAWSAccess	Service control	Allows access to every operation
<input checked="" type="checkbox"/> DenyUnapprovedActions	Service control	This prevents unapproved actions across my organization

DenyUnapprovedActions

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

Description: This prevents unapproved actions across my organization

Policy editor →

Accounts Organizational units Roots

Feedback English (US) Privacy Policy Terms of Use

Isengard AWS Organizations https://console.aws.amazon.com/organizations/home?region=us-east-1#/policies/p-ikjawcfn

Services Resource Groups

AWS Organizations

Accounts Organize accounts Policies Invitations Settings

All policies > DenyUnapprovedActions

Organizational Units Accounts

JSON

Edit

```
{ "Version": "2012-10-17", "Statement": [ { "Sid": "DenyUnapprovedAction", "Effect": "Deny", "Action": [ ] } ] }
```

Feedback English (US) Privacy Policy Terms of Use

The screenshot shows the AWS Organizations Policies page. The URL is https://console.aws.amazon.com/organizations/home?region=us-east-1#/policies/p-ikjawcfn. The top navigation bar includes links for AWS Identity - Home, Jade Sapphire Mong, PmStuff, AWS reInvent 2018, Audit Report (from), Travel, Consistent Authorization, Salesforce - Unlimite, and more. The main menu has options for Services, Resource Groups, and Support. The AWS logo is on the left. The page title is "AWS Organizations". Below it are tabs for Accounts, Organize accounts, and Policies, with "Policies" being the active tab. On the right, there are links for Invitations and Settings. The main content area shows the policy titled "DenyUnapprovedActions" under "All policies". An "Edit" button is highlighted with a blue box. The policy JSON is displayed:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "DenyUnapprovedAction",  
      "Effect": "Deny",  
      "Action": [  
        "ds:\"",  
        "iam:CreateUser",  
        "cloudtrail:StopLogging"  
      ],  
      "Resource": [  
        "*"  
      ]  
    }  
  ]  
}
```

IAM Management Console

https://console.aws.amazon.com/iam/home/?region=us-west-2#roles/challenge1-admin

AWS Identity - Home Jade Sapphire Mong PmStuff AWS reInvent 2018 Audit Report (from A Travel Consistent Authori... Salesforce - Unlimited...

aws-reinvent-2018-unicorns-d...

Global Support

aws Services Resource Groups

Search IAM

Role description This role is used to test challenge one for SCP restrictions with admin access. | Edit

Instance Profile ARNs

Path /

Creation time 2018-11-20 10:49 PST

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=challenge1-admin&account=128609111811

Permissions Trust relationships Tags Access Advisor Revoke sessions

▼ Permissions policies (1 policy applied)

Attach policies Add inline policy

Policy name	Policy type
AdministratorAccess	AWS managed policy

▶ Permissions boundary (not set)

Feedback English (US) © 2006 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

We will now work as an admin user here with the role above

The screenshot shows a Notepad++ window with the file path C:\Users\brigid\Desktop\reinvent2018Commands.txt. The code in the editor is as follows:

```
1 Commands
2
3 Challenge #1
4 aws iam create-user --user-name brigidj --profile challenge1-admin
5
6 aws cloudtrail stop-logging --name reinvent-challenge1 --region us-west-2 --profile challenge1-admin
7
8 aws iam list-roles --profile challenge1-admin
9
10 Challenge #2
11 Use console to store a secret (West Coast, then London)
12
13 Key: Reinvent
14 Value: isAwesome
15 Name:reInventDemo-Tuesday
16
17 Challenge #3:
18 IAM console - Create role
19 Permissions: unicorns=s3-read-write
20 Boundary: Region-restriction
21 Role Name: unicorns-reinvent-tuesday
22 Role Description: Creating a role to pass to my Lambda functions for reInvent-challenge1
23
24 Show unicorns-challenge3-with-boundary
25
26 aws s3api put-object --bucket reinvent-challenge3-success-demo-2018 --key ./pickles.jpg --body pickles.jpg --profile c
27 aws s3api put-object --bucket reinvent-challenge3-fail-demo-2018 --key ./pickles.jpg --body pickles.jpg --profile chal
28 aws s3api get-object --bucket reinvent-challenge3-success-demo-2018 --key ./pickles.jpg pickles.jpg --profile challeng
29
30 Challenge #4:
31 EC2 console in Oregon Region
32 Launch instance project=dorky
33 Stop instance with project=sneaky
```

The screenshot shows two windows side-by-side. On the left is a Notepad++ window displaying an AWS configuration file named 'config'. The file contains several IAM profile definitions, including 'challenge1-admin' which has a 'source_profile=default' setting. On the right is a screenshot of the AWS IAM 'Permissions' tab for a user named 'challenge1-admin'. A policy document is open, showing a single policy statement:

```

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "iam:CreateUser",
            "Resource": "arn:aws:iam::128609111811:user/*"
        }
    ]
}

```

The 'Add inline policy' button is visible at the bottom of the policy editor.

When the admin user tries to create a new IAM user, it gets denied because of the SCP

The screenshot shows a Windows Command Prompt window. The user runs the command `aws iam create-user --user-name brigidj --profile challenge1-admin`. The output shows an AccessDenied error:

```

Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\brigidj\Desktop\reInvent2018>aws iam create-user --user-name brigidj --profile challenge1-admin

An error occurred (AccessDenied) when calling the CreateUser operation: User: arn:aws:sts::128609111811:assumed-role/challenge1-admin/botocore-session-1543375461 is not authorized to perform: iam:CreateUser on resource: arn:aws:iam::128609111811:user/brigidj with an explicit deny

```

Trying to turn off CloudTrail logging also fails because of the SCP

The screenshot shows a Windows Command Prompt window. The user runs the command `aws cloudtrail stop-logging --name reInvent-challenge1 --region us-west-2 --profile challenge1-admin`. The output shows an AccessDeniedException error:

```

C:\Users\brigidj\Desktop\reInvent2018>aws cloudtrail stop-logging --name reInvent-challenge1 --region us-west-2 --
profile challenge1-admin

An error occurred (AccessDeniedException) when calling the StopLogging operation: User: arn:aws:sts::128609111811:assumed-role/challenge1-admin/botocore-session-1543375461 is not authorized to perform: cloudtrail:StopLogging on resource: arn:aws:cloudtrail:us-west-2:128609111811:trail/reInvent-challenge1 with an explicit deny

```

Listing of roles in the account should pass because it is not denied in the SCP

```
Command Prompt
    "Arn": "arn:aws:iam::128609111811:role/unicorns-pb-role"
},
{
    "Description": "reating a role to pass to my Lambda functions for reInvent-challenge1",
    "AssumeRolePolicyDocument": {
        "Version": "2012-10-17",
        "Statement": [
            {
                "Action": "sts:AssumeRole",
                "Effect": "Allow",
                "Principal": {
                    "Service": "lambda.amazonaws.com"
                }
            }
        ]
    },
    "MaxSessionDuration": 3600,
    "RoleId": "AROAIJY7WOR2H7FQ232B",
    "CreateDate": "2018-11-26T18:38:51Z",
    "RoleName": "unicorns-reinvent-monday",
    "Path": "/",
    "Arn": "arn:aws:iam::128609111811:role/unicorns-reinvent-monday"
}
]
}

D:\Users\brigidj\Desktop\reInvent2018>
```

Let's see SCPs in action

- Show SCP to deny access to modify AWS CloudTrail, Create IAM users, and AWS Directory Services
- Use the CLI as an administrator in the Unicorn-dev account and try to:
 - Create an IAM user
 - List roles
 - Stop logging in AWS CloudTrail



Challenge #2

Ensure your developers can create resources, but only in approved regions.

Pro Tip: Use the RequestedRegion AWS condition

Policy for challenge #2

```
{  
    "Effect": "Allow",  
    "Action": [  
        "secretsmanager:*",  
        "lambda:*",  
        "s3:PutObject",  
        "s3:GetObject",  
        "s3:DeleteObject"  
    ],  
    "Resource": "*",  
    "Condition": {  
        "StringEquals": {  
            "aws:RequestedRegion": [  
                "us-west-1",  
                "us-west-2"  
            ]  
        }  
    }  
}
```

This is what the policy looks like and we have specified the allowed regions in the policy

Policy for challenge #2

```
{  
    "Effect": "Allow",  
    "Action": "ec2:RunInstances",  
    "Resource": [  
        "arn:aws:ec2::*:subnet/*",  
        "arn:aws:ec2::*:key-pair/*",  
        "arn:aws:ec2::*:instance/*",  
        "arn:aws:ec2::*:snapshot/*",  
        "arn:aws:ec2::*:launch-template/*",  
        "arn:aws:ec2::*:volume/*",  
        "arn:aws:ec2::*:security-group/*",  
        "arn:aws:ec2::*:placement-group/*",  
        "arn:aws:ec2::*:network-interface/*",  
        "arn:aws:ec2::*:image/*"  
    ],  
    "Condition": {  
        "StringEquals": {  
            "aws:RequestedRegion": ["us-west-1", "us-west-2"]  
        }  
    }  
}  
aws re:Invent }
```



We also specify that EC2 instances should be allowed only in the 2 specified regions

Policy for challenge #2

```
{  
    "Effect": "Allow",  
    "Action": [  
        "ec2:Describe*",  
        "ec2:Get*",  
        "s3>ListBucket",  
        "s3>ListAllMyBuckets",  
        "iam>List*"  
    ],  
    "Resource": "*"
```

Let's see region control in action

Using the developer role for challenge 2:

- Create a secret in the west region



- Create a secret in the London region



We will try and create a secret using the AWS Secrets Manager in the west region and also in the London region which should not work

The screenshot shows the AWS Management Console home page. The user is signed in as 'CrazyCasey' from the Oregon region. The sidebar on the left lists recently visited services (EC2, IAM, Secrets Manager, S3, Lambda) and provides links to 'Build a solution', 'Launch a virtual machine', 'Build a web app', and 'Build using virtual servers'. The main content area features the 'AWS Management Console' logo and various service links such as 'Access the Console', 'My Account', 'Role History', and 'Explore AWS'. A tooltip for 'Role History' indicates it can extend to other regions.

The screenshot shows the IAM Management Console. The 'Roles' section is selected in the sidebar. A search bar at the top contains the text 'challenge'. The results table shows six roles: 'challenge1-admin', 'challenge2-regions' (which is highlighted with a cursor), 'challenge3-permission-boundaries', 'challenge4-tag-permissions', 'challenge5-match-tags', and 'unicorns-challenge3-with-boundary'. Each role has a description and a 'Trusted entities' column showing the account number 332207979596.

Role name	Description	Trusted entities
challenge1-admin	This role is used to test challenge one for SCP restrictions with admin...	Account: 332207979596
challenge2-regions	This roles is used to demonstrate creating resources in unapproved r...	Account: 332207979596
challenge3-permission-boundaries	This role is used to demonstrate creating roles with permission boun...	Account: 332207979596
challenge4-tag-permissions	This role is used to demonstrate using tags to control access.	Account: 332207979596
challenge5-match-tags	This role will be used to demonstrate how you can use matching tags...	Account: 332207979596
unicorns-challenge3-with-boundary	This role is used to demonstrate permission boundaries	Account: 332207979596

IAM Management Console https://console.aws.amazon.com/iam/home?region=us-west-2#roles/challenge2-regions

AWS Identity - Home Jade Sapphire Mong PmStuff AWS reInvent 2018 Audit Report (from P Travel Consistent Authoriza Salesforce - Unlim Global Support aws-reinvent-2018-unicorns-d...

Summary

Role ARN arn:aws:iam::128609111811:role/challenge2-regions

Role description This role is used to demonstrate creating resources in unapproved regions | Edit

Instance Profile ARNs

Path /

Creation time 2018-11-20 10:51 PST

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=challenge2-regions&account=128609111811

Permissions Trust relationships Tags Access Advisor Revoke sessions

▼ Permissions policies (2 policies applied)

Attach policies Add inline policy

Policy name	Policy type
passRole-permissions	Managed policy
region-restriction	Managed policy

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

IAM Management Console https://console.aws.amazon.com/iam/home?region=us-west-2#roles/challenge2-regions

AWS Identity - Home Jade Sapphire Mong PmStuff AWS reInvent 2018 Audit Report (from P Travel Consistent Authoriza Salesforce - Unlim Global Support aws-reinvent-2018-unicorns-d...

Summary

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=challenge2-regions&account=128609111811

Permissions Trust relationships Tags Access Advisor Revoke sessions

▼ Permissions policies (2 policies applied)

Attach policies Add inline policy

Policy name	Policy type
passRole-permissions	Managed policy
region-restriction	Managed policy

Policy summary (JSON) Edit policy Simulate policy

```
1 - {
2 -     "Version": "2012-10-17",
3 -     "Statement": [
4 -         {
5 -             "Sid": "VisualEditor0",
6 -             "Effect": "Allow",
7 -             "Action": [
8 -                 "secretsmanager:*",
9 -                 "lambda:*",
10 -                 "s3:PutObject",
11 -                 "s3:GetObject",
12 -                 "s3:DeleteObject"
13 -             ],
14 -             "Resource": "*",
15 -             "Condition": {
16 -                 "StringEquals": {
17 -                     "AWS:SourceRegion": "us-east-1"
18 -                 }
19 -             }
20 -         }
21 -     ]
22 - }
```

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

IAM Management Console https://console.aws.amazon.com/iam/home?region=us-west-2#roles/challenge2-regions

AWS Identity - Home Jade Sapphire Mong PmStuff AWS reInvent 2018 Audit Report (from P Travel Consistent Authoriza Salesforce - Unlim Global Support aws-reinvent-2018-unicorns-d...

Summary

Policy name

Policy name	Policy type
passRole-permissions	Managed policy
region-restriction	Managed policy

Policy summary (JSON) Edit policy Simulate policy

```
1 - {
2 -     "Version": "2012-10-17",
3 -     "Statement": [
4 -         {
5 -             "Sid": "VisualEditor0",
6 -             "Effect": "Allow",
7 -             "Action": [
8 -                 "secretsmanager:*",
9 -                 "lambda:*",
10 -                 "s3:PutObject",
11 -                 "s3:GetObject",
12 -                 "s3:DeleteObject"
13 -             ],
14 -             "Resource": "*",
15 -             "Condition": {
16 -                 "StringEquals": {
17 -                     "AWS:SourceRegion": "us-east-1"
18 -                 }
19 -             }
20 -         }
21 -     ]
22 - }
```

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

IAM Management Console

https://console.aws.amazon.com/iam/home?region=us-west-2#/roles/challenge2-regions

AWS Identity - Home Jade Sapphire Mong PinStuff AWS re:Invent 2018 Audit Report (from Amazon) Travel Consistent Authorization Salesforce - Unlimited

Services Resource Groups

Search IAM

region-restriction Managed policy

Policy summary (JSON) Edit policy Simulate policy

```
"arn:aws:ec2:*:*:launch-template/*",
"arn:aws:ec2:*:*:volume/*",
"arn:aws:ec2:*:*:security-group/*",
"arn:aws:ec2:*:*:placement-group/*",
"arn:aws:ec2:*:*:network-interface/*",
"arn:aws:ec2:*:*:image/*"
],
"Condition": {
    "StringEquals": [
        "aws:RequestedRegion": [
            "us-west-1",
            "us-west-2"
        ]
    ]
}
},
```

Permissions boundary (not set)

The screenshot shows the IAM Management Console with the URL <https://console.aws.amazon.com/iam/home?region=us-west-2#/roles/challenge2-regions>. The left sidebar is collapsed, and the main area displays the 'switch roles in the console' interface. The 'Permissions' tab is active, showing 'Permissions policies (2 policies applied)'. One policy, 'passRole-permissions', is listed as a Managed policy. The policy summary shows the following JSON configuration:

```
1 - {
2     "Version": "2012-10-17",
3     "Statement": [
4         {
5             "Sid": "VisualEditor0",
6             "Effect": "Allow",
7             "Action": "iam:PassRole",
8             "Resource": [
9                 "arn:aws:iam::128609111811:role/EC2-role-to-pass",
10                "arn:aws:iam::128609111811:role/lambda-role-to-pass",
11                "arn:aws:iam::128609111811:role/unicorns-*"
12            ]
13        }
14    ]
15}
```

The screenshot shows the AWS IAM Management Console interface. On the left, there's a sidebar with navigation links: Dashboard, Groups, Users, Roles (which is selected), Policies, Identity providers, Account settings, Credential report, and Encryption keys. The main area has tabs for Policy summary, JSON, and Edit policy. The current view is the Policy summary tab.

Policy summary

region-restriction Managed policy

Policy JSON

```
1  {
2      "Version": "2012-10-17",
3      "Statement": [
4          {
5              "Sid": "VisualEditor0",
6              "Effect": "Allow",
7              "Action": "iam:PassRole",
8              "Resource": [
9                  "arn:aws:iam::128609111811:role/EC2-role-to-pass",
10                 "arn:aws:iam::128609111811:role/lambda-role-to-pass",
11                 "arn:aws:iam::128609111811:role/unicorns-*"
12             ]
13         }
14     ]
15 }
```

Policy summary

region-restriction Managed policy

Policy JSON

```
1  {
2      "Version": "2012-10-17",
3      "Statement": [
4          {
5              "Sid": "VisualEditor0",
6              "Effect": "Allow",
7              "Action": "ec2:LaunchTemplate/*",
8              "Resource": [
9                  "arn:aws:ec2:/*/*:launch-template/*",
10                 "arn:aws:ec2:/*/*:volume/*",
11                 ...
12             ]
13         }
14     ]
15 }
```

This **passRole** policy will allow us to test launching some EC2 instances or some Lambda functions.

AWS Management Console

Services Resource Groups challenge2 Oregon Support

AWS Management Console

AWS services

Find a service by name or feature (for example, EC2, S3 or VM, storage)

Recently visited services: EC2, Secrets Manager, IAM, Lambda

All services

Build a solution

Get started with simple wizards and automated workflows.

Launch a virtual machine Build a web app Build using virtual servers

Access resources on the go

Access the Management Console using the AWS Console Mobile App. Learn more

Explore AWS

Amazon Redshift

Fast, simple, cost-effective data warehouse that can extend queries to your data lake. Learn more

Run Serverless Containers with AWS Fargate

AWS Fargate runs and scales your containers without having to manage servers or clusters. Learn more

Secrets Manager

Services Resource Groups challenge2 Oregon Support

Secrets

Store a new secret

Secret name	Description	Last retrieved
reinvent-demo	This is for a reinventDemo	11/20/2018
reinventDemo2	my demo secret	-
reinvent-demo-monday	this is cool	-

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Secrets Manager

Services Resource Groups challenge2 Oregon Support

Failed to fetch a list of RDS databases
User: arn:aws:sts::128609111811:assumed-role/challenge2-regions/CrazyCasey is not authorized to perform: rds:DescribeDBInstances

Step 1 AWS Secrets Manager > Secrets > Store a new secret

Secret type

Step 2 Name and description

Step 3 Configure rotation

Step 4 Review

Store a new secret

Select secret type Info

Credentials for RDS database Credentials for other database Other type of secrets (e.g. API key)

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

Secret key/value Plaintext

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Secrets Manager Services Resource Groups

Review Step 3

Credentials for RDS database Credentials for other database Other type of secrets (e.g. API key)

Specify the key/value pairs to be stored in this secret [Info](#)

Secret key/value Plaintext

reinventDay2 isAwesome [+ Add row](#)

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](#)

Cancel [Next](#)

Feedback English (US) © 2008-2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Secrets Manager Services Resource Groups

Review Step 4

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

Description - optional
storing my secret Maximum 250 characters

Tags - optional

Key Value - optional
 [Remove](#) [Add](#)

Cancel [Previous](#) [Next](#)

Feedback English (US) © 2008-2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Secrets Manager Services Resource Groups

Review Step 3

If you enable automatic rotation, the first rotation will happen immediately when you store this secret. If this secret is already in use, you must update your applications to retrieve it from AWS Secrets Manager. Read the [getting started guide](#) on rotation.

Configure automatic rotation - optional info Configure AWS Secrets Manager to rotate this secret automatically. Read the [getting started guide](#) on rotation.

Disable automatic rotation Recommended when your applications are using this secret and have not been updated to use AWS Secrets Manager.

Enable automatic rotation Recommended when your applications are not using this secret yet.

Select rotation interval info This secret will be rotated based on the schedule you determine.
 Maximum 365 days

Choose an AWS Lambda function info Select an AWS Lambda function that has permissions to rotate this secret.
 [Create function](#)

Cancel [Previous](#) [Next](#)

Feedback English (US) © 2008-2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

The screenshot shows the AWS Secrets Manager console with the Java tab selected. A code snippet is displayed:

```

1 // Use this code snippet in your app.
2 // If you need more information about configurations or implementing the sample code, visit the
3 // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/java-dg-samples.html#prerequisite
4
5 public static void getSecret() {
6
7     String secretName = "mytestsecret-tues";
8     String region = "us-west-2";
9
10    // Create a Secrets Manager client
11    AWSSecretsManager client = AWSSecretsManagerClientBuilder.standard()
12        .withRegion(region)
13        .build();

```

Below the code, there is a link to "Download AWS SDK for Java". At the bottom right, there are "Cancel", "Previous", and "Store" buttons, with "Store" being highlighted.

This is the code we can use in our app to retrieve the secret

The screenshot shows the AWS Secrets Manager console with a green success message at the top: "Your secret mytestsecret-tues has been successfully stored. Use the sample code to update your applications to retrieve this secret." Below this, the "Secrets" section is shown with a table:

Secret name	Description	Last retrieved
reinvent-demo	This is for a reinventDemo	11/20/2018
reinventDemo2	my demo secret	-
reinvent-demo-monday	this is cool	-
mytestsecret-tues	storing my secret	-

We are successfully allowed to create a secret in the west region due to our policy.

The screenshot shows the AWS Secrets Manager console in the London region. A red error message at the top states: "You do not have permission to list secrets in Secrets Manager. You do not have permission to list secrets. As a result, you cannot view or select from existing secrets in your account. Contact your administrator to obtain ListSecrets access." Below this, the "Secrets" section is shown with a table:

Secret name	Description	Last retrieved
No secrets		

We are now in the London region and cannot even list the available secrets, we will now try to create a secret here

Secret Manager Services Resource Groups Step 2 Name and description Step 3 Configure rotation Step 4 Review

Store a new secret

Select secret type Info

Credentials for RDS database Credentials for other database Other type of secrets (e.g. API key)

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

Secret key/value Plaintext

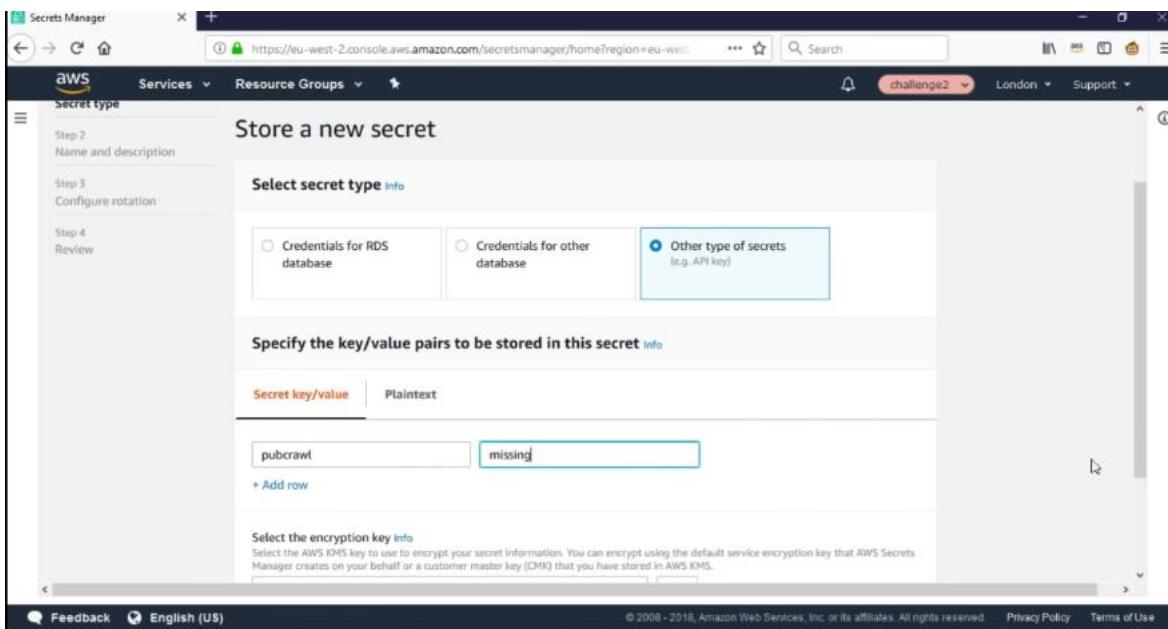
pubcrawl

+ Add row

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.

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use



Secret Manager Services Resource Groups Review

Credentials for RDS database Credentials for other database Other type of secrets (e.g. API key)

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

Secret key/value Plaintext

pubcrawl

+ Add row

Select the encryption key Info

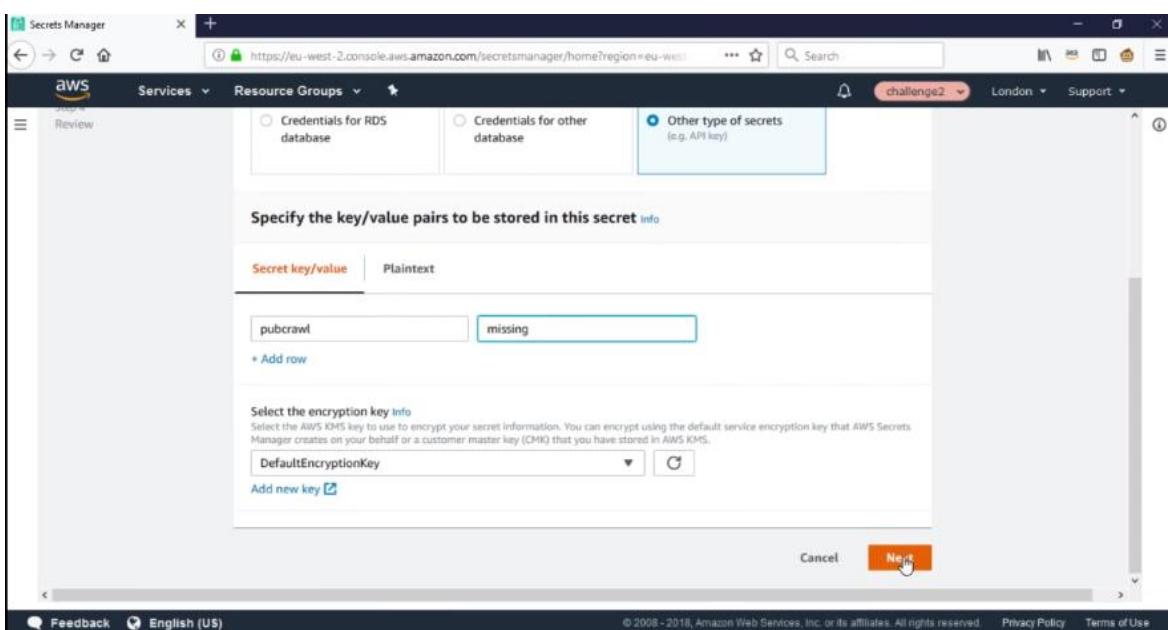
Select the AWS KMS key to use to encrypt your secret information. You can encrypt using the default service encryption key that AWS Secrets Manager creates on your behalf or a customer master key (CMK) that you have stored in AWS KMS.

DefaultEncryptionKey

Add new key

Cancel

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use



Secret Manager Services Resource Groups Step 4 Review

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

pubcrawl

Secret name must contain only alphanumeric characters and the characters /, *, ., @.

Description - optional

missing beer

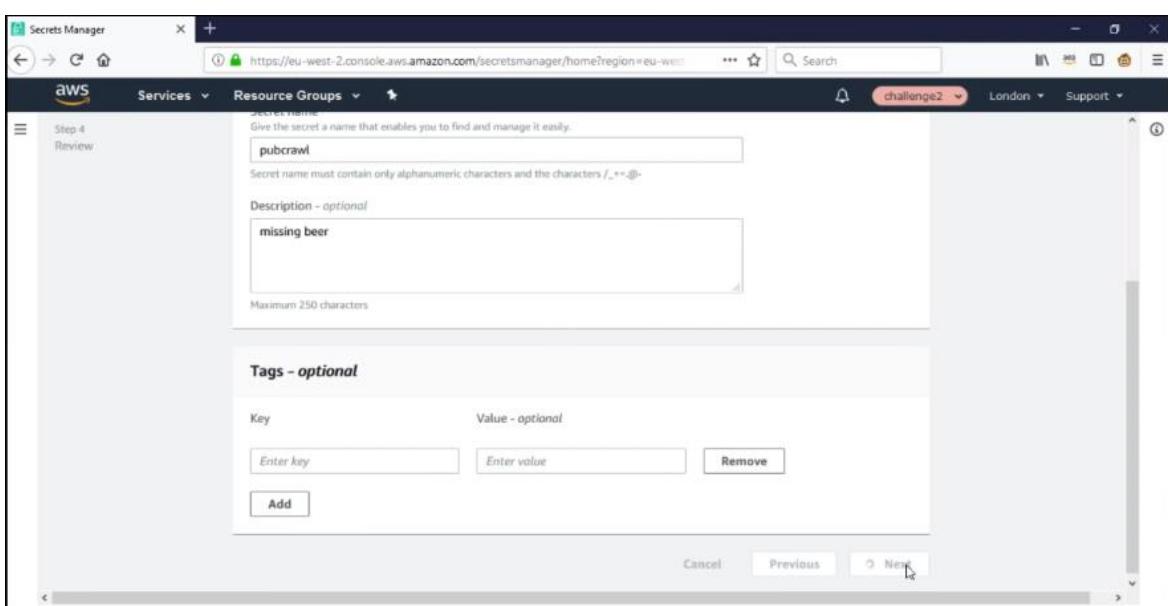
Maximum 250 characters

Tags - optional

Key Value - optional Remove

Add

Cancel Previous



Configure automatic rotation - optional info

Configure AWS Secrets Manager to rotate this secret automatically. Read the [getting started guide on rotation](#).

Disable automatic rotation
Recommended when your applications are using this secret and have not been updated to use AWS Secrets Manager.

Enable automatic rotation
Recommended when your applications are not using this secret yet.

Select rotation interval info

This secret will be rotated based on the schedule you determine.

30 days

Maximum 365 days

You need permissions

You do not have permission to perform this operation. Ask your administrator to add permissions.

Choose an AWS Lambda function info

Select an AWS Lambda function that has permissions to rotate this secret.

Create function

Cancel Previous Next

Feedback English (US)

© 2008-2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Java Javascript C# Python3 Ruby Go

```
1 // Use this code snippet in your app.
2 // If you need more information about configurations or implementing the sample code, visit the
3 // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/java-dg-samples.html#prerequisites
4
5 public static void getSecret() {
6
7     String secretName = "pubcrawl";
8     String region = "eu-west-2";
9
10    // Create a Secrets Manager client
11    AWSSecretsManager client = AWSSecretsManagerClientBuilder.standard()
12        .withRegion(region)
13        .build();
14}
```

Download AWS SDK for Java

Cancel Previous Next

Failed to fetch a list of RDS databases

User: arn:aws:sts::128609111811:assumed-role/challenge2-regions/CrazyCasey is not authorized to perform: rds:DescribeDBInstances

Failed to create secret

Your request has a problem. User: arn:aws:sts::128609111811:assumed-role/challenge2-regions/CrazyCasey is not authorized to perform: secretsmanager:CreateSecret on resource: arn:aws:secretsmanager:eu-west-2:128609111811:secret:pubcrawl-YFmkRw

Step 1 AWS Secrets Manager > Secrets > Store a new secret

Step 2 Name and description

Step 3 Configure rotation

Step 4 Review

Secret type
Other type of secret

Encryption key
DefaultEncryptionKey

Secret name
pubcrawl

Description

Cancel Previous Next

This failed as expected in the London region

Let's see region control in action

Using the developer role for challenge 2:

- Create a secret in the west region



- Create a secret in the London region



Challenge #3

Enable your developers to create IAM roles to pass to EC2 and Lambda, but ensure they cannot exceed their own permissions.

Pro Tip: Require and use role naming conventions to control the roles developers can manage.

We want to allow the developers to create their IAM roles to pass to EC2 or Lambda, but we want to make sure they cannot escalate their privileges beyond this scope. We have to constrain the developers with the use of permission boundaries and also to their own little world using naming-prefixes.

Four parts required for permission boundaries

- ⚡ Allow create managed policies
- ⚡ Allow create role, but only with a specific permission boundary
This is a condition with a pointer to an existing managed policy
- ⚡ Allow attach managed policies, but only to roles with a specific boundary
- ⚡ Allow passRole for these roles using a naming requirement

Policy for Challenge #3

Allow create managed policies

```
{  
    "Effect": "Allow",  
    "Action": [  
        "iam:CreatePolicy",  
        "iam:CreatePolicyVersion",  
        "iam:DeletePolicyVersion"  
    ],  
    "Resource": "arn:aws:iam::128609111811:policy/unicorns-*"  
}
```

This is how we can constrain the developer to the unicorn-* prefixes.

Policy for Challenge #3

Allow create role, but only with a specific permission boundary

Allow attach managed policies, but only to roles with a specific boundary

```
{  
    "Effect": "Allow",  
    "Action": [  
        "iam:DetachRolePolicy",  
        "iam:CreateRole",  
        "iam:AttachRolePolicy"  
    ],  
    "Resource": "arn:aws:iam::128609111811:role/unicorns-*",  
    "Condition": {  
        "StringEquals": {  
            "iam:PermissionsBoundary":  
                "arn:aws:iam::128609111811:policy/region-restriction"  
        }  
    }  
}
```

The **StringEquals** condition uses a pointer to a managed policy that provides the region restriction to only the 2 regions allowed as the permission boundary.

Permission boundary workflows



Admin creates maximum permissions



Admin **allows** developers to create role with maximum permissions



Developer creates role with **maximum permissions** and **specific permissions**



Developers passes the role to application resources

Let's see permission boundaries in action

- Using the developer role, create a role with a permission boundary
- Use a role with a permission boundary to put data in S3 for approved regions and for unapproved regions.

Screenshot of the AWS IAM Management Console showing a search results page for roles. The search term 'challenge' is entered in the search bar. The results table shows six entries:

Role name	Description	Trusted entities
challenge1-admin	This role is used to test challenge one for SCP restrictions with adm...	Account: 332207979596
challenge2-regions	This roles is used to demonstrate creating resources in unapproved r...	Account: 332207979596
challenge3-permission-boundaries	This role is used to demonstrate creating roles with permission boun...	Account: 332207979596
challenge4-tag-permissions	This role is used to demonstrate using tags to control access.	Account: 332207979596
challenge5-match-tags	This role will be used to demonstrate how you can use matching tags...	Account: 332207979596
unicorns-challenge3-with-boundary	This role is used to demonstrate permission boundaries	Account: 332207979596

Screenshot of the AWS IAM Management Console showing the 'Summary' tab for the 'challenge3-permission-boundaries' role. The role ARN is listed as arn:aws:iam::128609111811:role/challenge3-permission-boundaries. The role description states: 'This role is used to demonstrate creating roles with permission boundaries.' The creation time is 2018-11-20 10:52 PST. The maximum CLI/API session duration is 1 hour. A link is provided for users to switch roles in the console: https://signin.aws.amazon.com/switchrole?roleName=challenge3-permission-boundaries&account=128609111811. The 'Permissions' tab is selected, showing three policies applied: passRole-permissions. The policy type is Managed policy.

The screenshot shows the AWS IAM Management Console with a role named 'challenge3-permission-boundaries'. The 'Permissions' tab is active, displaying three policies applied to the role:

- passRole-permissions (Managed policy)
- region-restriction (Managed policy)
- create-role-with-region-boundary (Managed policy)

The 'create-role-with-region-boundary' policy is currently selected, indicated by a cursor icon.

We already had the **passRole** policy and the region restriction policy **region-restriction** that restricts the developer to be able to create resources in only 2 regions. Then we just added the new policy **create-role-with-region-boundary** that adds the boundary to what they can actually do.

The screenshot shows the 'Edit policy' view for the 'create-role-with-region-boundary' policy. The JSON code for the policy is as follows:

```
1 - {
2 -     "Version": "2012-10-17",
3 -     "Statement": [
4 -         {
5 -             "Sid": "VisualEditor0",
6 -             "Effect": "Allow",
7 -             "Action": [
8 -                 "iam:DetachRolePolicy",
9 -                 "iam:CreateRole",
10 -                 "iam:AttachRolePolicy"
11 -             ],
12 -             "Resource": "arn:aws:iam::128609111811:role/unicorns-*",
13 -             "Condition": {
14 -                 "StringEquals": {
15 -                     "iam:PermissionsBoundary": "arn:aws:iam::128609111811:policy/region-restriction"
16 -                 }
17 -             }
18 -         }
19 -     ]
20 - }
```

The policy summary shows two managed policies attached to the role: 'region-restriction' and 'create-role-with-region-boundary'.

IAM Management Console https://console.aws.amazon.com/iam/home?region=us-west-2#/roles/challenge3-permission-boundaries

AWS Identity - Home Jade Sapphire Mong PmStuff AWS Reinvent 2018 Audit Report (from A Travel Consistent Authorization Salesforce - Unlimite

aws-reinvent-2018-unicorns-d... Global Support

Search IAM

Dashboard Groups Users Roles Policies Identity providers Account settings Credential report Encryption keys

region-restriction Managed policy

create-role-with-region-boundary Managed policy

Policy summary JSON Edit policy Simulate policy

```

15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
  "Sid": "VisualEditor1",
  "Effect": "Allow",
  "Action": [
    "iam:CreatePolicy",
    "iam:CreatePolicyVersion",
    "iam:DeletePolicyVersion"
  ],
  "Resource": "arn:aws:iam::128609111811:policy/unicorns-*"
},
{
  "Sid": "VisualEditor2",
  "Effect": "Allow",

```

Permissions boundary (not set)

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

IAM Management Console https://console.aws.amazon.com/iam/home?region=us-west-2#/roles/challenge3-permission-boundaries

AWS Identity - Home Jade Sapphire Mong PmStuff AWS Reinvent 2018 Audit Report (from A Travel Consistent Authorization Salesforce - Unlimite

aws-reinvent-2018-unicorns-d... Global Support

Search IAM

Dashboard Groups Users Roles Policies Identity providers Account settings Credential report Encryption keys

region-restriction Managed policy

create-role-with-region-boundary Managed policy

Policy summary JSON Edit policy Simulate policy

```

26
27
28
29
30
31
32
33
34
35
36
37
38
39
39
40
41
42
  "Resource": "arn:aws:iam::128609111811:policy/unicorns-*"
},
{
  "Sid": "VisualEditor2",
  "Effect": "Allow",
  "Action": [
    "iam:GetRole",
    "iam:GetPolicyVersion",
    "iamGetInstanceProfile",
    "iam:GetPolicy",
    "iam:GetRolePolicy"
  ],
  "Resource": "*"
}

```

Permissions boundary (not set)

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

IAM Management Console https://console.aws.amazon.com/iam/home?region=eu-west-2#/home

challenge3 Global Support

Search IAM

Dashboard Groups Users Roles Policies Identity providers Account settings Credential report Encryption keys

Welcome to Identity and Access Management

IAM users sign-in link: <https://t128609111811.signin.aws.amazon.com/console> | Customize

We encountered the following errors while processing your request:

User: arn:aws:sts::128609111811:assumed-role/challenge3-permission-boundaries/CrazyCasey is not authorized to perform: iam:GetAccountSummary on resource: *

Security Status 0 out of 4 complete. 4 checks failed.

- Activate MFA on your root account
- Create individual IAM users
- Use groups to assign permissions
- Apply an IAM password policy

Feature Spotlight

Introduction to AWS IAM

Additional Information

IAM best practices IAM documentation Web Identity Federation Playground Policy Simulator Videos, IAM release history and additional resources

IAM Management Console

Services Resource Groups

Create role Delete role

Search IAM

Showing 24 results

Role name	Description	Trusted entities
aws-reinvent-2018-unicorns-dev		Account: 727820809195
AwsSecurityAudit		Account: 877377650033
AwsSecurityNacundaAudit		Account: 350429053849
AWSServiceRoleForIsengardCo.	This role will allow Isengard to manage and audit your account. This role is used by AWS Organizations to enable integrati...	Account: 727820809195 (Service-Linked role)
AWSServiceRoleForOrganizations	Service-linked role used by AWS Organizations to enable integrati...	AWS service: organizations (Service-Linked role)
AWSServiceRoleForSupport	Enables resource access for AWS to provide billing, administrative...	AWS service: support (Service-Linked role)
AWSServiceRoleForTrustedAdv...	Access for the AWS Trusted Advisor Service to help reduce cost, i...	AWS service: trustedadvisor (Service-Linked role)
challenge1-admin	This role is used to test challenge one for SCP restrictions with ad...	Account: 332207979596
challenge2-regions	This roles is used to demonstrate creating resources in unapprov...	Account: 332207979596
challenge3-permission-boundar...	This role is used to demonstrate creating roles with permission bo...	Account: 332207979596
challenge4-tag-permissions	This role is used to demonstrate using tags to control access.	Account: 332207979596
challenge5-match-tags	This role will be used to demonstrate how you can use matching ta...	Account: 332207979596

https://console.aws.amazon.com/iam/home?region=eu-west-2#roles\$new

IAM Management Console

Services Resource Groups

Create role

Select type of trusted entity

1 2 3 4

AWS service EC2, Lambda and others

Another AWS account Belonging to you or 3rd party

Web identity Cognito or any OpenID provider

SAML 2.0 federation Your corporate directory

Allows AWS services to perform actions on your behalf. Learn more

Choose the service that will use this role

EC2 Allows EC2 instances to call AWS services on your behalf.

Lambda Allows Lambda functions to call AWS services on your behalf.

API Gateway	CodeBuild	EC2 - Fleet	Inspector	Redshift
AWS Support	CodeDeploy	EKS	IoT	Rekognition
AppSync	Config	EMR	Kinesis	S3
Application Auto Scaling	Connect	ElastiCache	Lambda	SMS

* Required Next: Permissions

Feedback English (US)

IAM Management Console

Services Resource Groups

Choose one or more policies to attach to your new role.

Create policy

Filter policies Q s3

Showing 7 results

Policy name	Used as	Description
AmazonDMSRedshiftS3Role	None	Provides access to manage S3 settings f...
AmazonS3FullAccess	None	Provides full access to all buckets via th...
AmazonS3ReadOnlyAccess	None	Provides read only access to all buckets...
QuickSightAccessForS3StorageManagement...	None	Policy used by QuickSight team to acces...
retrieve-s3-and-secrets	Permissions policy (2)	This policy grants access to retrieve sec...
unicorns-s3-read	None	This policy grants access to read S3 in o...
unicorns-s3-read-write	Permissions policy (3)	This enables S3 read and write access f...

Set permissions boundary

* Required Cancel Previous Next: Tags

IAM Management Console

Services Resource Groups challenge3 Global Support

Choose one or more policies to attach to your new role.

Create policy

Filter policies s3 Showing 7 results

Policy name	Used as	Description
This enables S3 read and write access for my lambda function		

Policy summary (JSON Edit policy

```
{ "Version": "2012-10-17", "Statement": [ { "Sid": "VisualEditor0", "Effect": "Allow", "Action": [ "s3:PutObject", "s3:GetObject", "s3>ListAllMyBuckets", ] } ] }
```

Set permissions boundary

* Required Cancel Previous Next: Tags

This is a very general S3 policy that has no region restriction on it.

IAM Management Console

Services Resource Groups challenge3 Global Support

Create policy

Filter policies s3 Showing 7 results

Policy name	Used as	Description
AmazonDMSRedshiftS3Role	None	Provides access to manage S3 settings ...
AmazonS3FullAccess	None	Provides full access to all buckets via th...
AmazonS3ReadOnlyAccess	None	Provides read only access to all buckets...
QuickSightAccessForS3StorageManagement...	None	Policy used by QuickSight team to acces...
retrieve-s3-and-secrets	Permissions policy (2)	This policy grants access to retrieve sec...
unicorns-s3-read	None	This policy grants access to read S3 in o...
unicorns-s3-read-write	Permissions policy (3)	This enables S3 read and write access t...

Set permissions boundary

* Required Cancel Previous Next: Tags

IAM Management Console

Services Resource Groups challenge3 Global Support

Create role

Add tags (optional)

IAM tags are key-value pairs you can add to your role. Tags can include user information, such as an email address, or can be descriptive, such as a job title. You can use the tags to organize, track, or control access for this role. Learn more

Key	Value (optional)	Remove
Add new key		

You can add 50 more tags.

Cancel Previous Next: Review

IAM Management Console challenge3 Global Support

[Services](#) [Resource Groups](#)

Create role

Review

Provide the required information below and review this role before you create it.

Role name*	unicorns-reinvent-tuesday
Use alphanumeric and '-_,@_-' characters. Maximum 64 characters.	
Role description	Creating a role to pass to my Lambda functions for reinvent-challenge!
Maximum 1000 characters. Use alphanumeric and '-_,@_-' characters.	
Trusted entities	AWS service: lambda.amazonaws.com
Policies	unicorns-s3-read-write

Permissions boundary: Permissions boundary is not set

* Required

[Cancel](#) [Previous](#) [Create role](#)

[Feedback](#) [English \(US\)](#)

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)

IAM Management Console challenge3 Global Support

[Services](#) [Resource Groups](#)

Review

Provide the required information below and review this role before you create it.

1 You need permissions
You do not have the permission required to perform this operation. Ask your administrator to add permissions. [Learn more](#)

User: arn:aws:sts::128609111811:assumed-role/challenge3-permission-boundaries/CrazyCasey is not authorized to perform: iam:CreateRole on resource: arn:aws:iam::128609111811:role/unicorns-reinvent-tuesday

Role name*	unicorns-reinvent-tuesday
Use alphanumeric and '-_,@_-' characters. Maximum 64 characters.	
Role description	Creating a role to pass to my Lambda functions for reinvent-challenge!
Maximum 1000 characters. Use alphanumeric and '-_,@_-' characters.	
Trusted entities	AWS service: lambda.amazonaws.com

* Required

[Cancel](#) [Previous](#) [Create role](#)

[Feedback](#) [English \(US\)](#)

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)

The create role is denied because the developer does not have access to create roles in this region

IAM Management Console challenge3 Global Support

[Services](#) [Resource Groups](#)

Create policy

Filter policies

Policy name	Used as	Description
AmazonDMSRedshiftS3Role	None	Provides access to manage S3 settings for...
AmazonS3FullAccess	None	Provides full access to all buckets via th...
AmazonS3ReadOnlyAccess	None	Provides read only access to all buckets...
QuickSightAccessForS3StorageManagement...	None	Policy used by QuickSight team to acces...
retrieve-s3-and-secrets	Permissions policy (2)	This policy grants access to retrieve sec...
unicorns-s3-read	None	This policy grants access to read S3 in o...
unicorns-s3-read-write	Permissions policy (3)	This enables S3 read and write access f...

Set permissions boundary

Set a permissions boundary to control the maximum permissions this role can have. This is an advanced feature used to delegate permission management to others. [Learn more](#)

* Required

[Cancel](#) [Previous](#) [Next: Tags](#)

[Feedback](#) [English \(US\)](#)

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)

Set permissions boundary

Set a permissions boundary to control the maximum permissions this role can have. This is an advanced feature used to delegate permission management to others. [Learn more](#)

Create role without a permissions boundary
 Use a permissions boundary to control the maximum role permissions

Select policy to set the permissions boundary

[Create policy](#)

Filter policies		
Policy name	Used as	Description
create-role-with-region-boundary	Permissions policy (1)	This allows developers to create roles w...
region-restriction	Permissions policy (2), Boundary (3)	This policy restricts the regions users ca...
WAFRegionalLoggingServiceRolePolicy	None	Creating SLR to write customer's logs to ...

* Required [Cancel](#) [Previous](#) [Next: Tags](#)

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Create role

Review

Provide the required information below and review this role before you create it.

Role name* unicorns-reinvent-tuesday
Use alphanumeric and +, -, @, _ characters. Maximum 64 characters.

Role description Creating a role to pass to my Lambda functions for reinvent-challenge!

Maximum 1000 characters. Use alphanumeric and +, -, @, _ characters.

Trusted entities AWS service: lambda.amazonaws.com

Policies unicorns-s3-read-write

Permissions boundary region-restriction

* Required [Cancel](#) [Previous](#) [Create role](#)

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

The role unicorns-reinvent-tuesday has been created.

Create role [Delete role](#)

Role name	Description	Trusted entities
aws-reinvent-2018-unicorns-dev		Account: 727820809195
AwsSecurityAudit		Account: 877377650033
AwsSecurityNacundaAudit		Account: 350429083849
AWSServiceRoleForIsengardCo...	This role will allow Isengard to manage and audit your account. Th...	Account: 727820809195 (Service-Linked role)
AWSServiceRoleForOrganizations	Service-linked role used by AWS Organizations to enable integrati...	AWS service: organizations (Service-Linked ...)
AWSServiceRoleForSupport	Enables resource access for AWS to provide billing, administrative...	AWS service: support (Service-Linked role)
AWSServiceRoleForTrustedAdv...	Access for the AWS Trusted Advisor Service to help reduce cost, i...	AWS service: trustedadvisor (Service-Linked ...)
challenge1-admin	This role is used to test challenge one for SCP restrictions with ad...	Account: 332207979596
challenge2-regions	This roles is used to demonstrate creating resources in unapprov...	Account: 332207979596
challenge3-permission-boundar...	This role is used to demonstrate creating roles with permission bo...	Account: 332207979596

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Now when we use the correct permission boundary SCP, the developer can now create the role successfully

IAM Management Console https://console.aws.amazon.com/iam/home?region=us-west-2#roles

AWS Identity - Home Jade Sapphire Mong PmStuff AWS reInvent 2018 Audit Report (from A Travel Consistent Authoriza Salesforce - Unlimite

aws-reinvent-2018-unicorns-d... Global Support

Search IAM Create role Delete role

Showing 6 results

Role name	Description	Trusted entities
challenge1-admin	This role is used to test challenge one for SCP restrictions with admin...	Account: 332207979596
challenge2-regions	This roles is used to demonstrate creating resources in unapproved r...	Account: 332207979596
challenge3-permission-boundaries	This role is used to demonstrate creating roles with permission boun...	Account: 332207979596
challenge4-tag-permissions	This role is used to demonstrate using tags to control access.	Account: 332207979596
challenge5-match-tags	This role will be used to demonstrate how you can use matching tags...	Account: 332207979596
unicorns-challenge3-with-boundary	This role is used to demonstrate permission boundaries	Account: 332207979596

IAM Management Console https://console.aws.amazon.com/iam/home?region=us-west-2#roles/unicorns-challenge3-with-boundary

AWS Identity - Home Jade Sapphire Mong PmStuff AWS reInvent 2018 Audit Report (from A Travel Consistent Authoriza Salesforce - Unlimite

aws-reinvent-2018-unicorns-d... Global Support

Search IAM Roles > unicorns-challenge3-with-boundary Summary Delete role

Role ARN am.aws.iam.: 128609111811.role/unicorns-challenge3-with-boundary

Role description This role is used to demonstrate permission boundaries | Edit

Instance Profile ARNs

Path /

Creation time 2018-11-21 11:49 PST

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=unicorns-challenge3-with-boundary&account=128609111811

Permissions Trust relationships Tags Access Advisor Revoke sessions

Permissions policies (1 policy applied)

Attach policies Add inline policy

Policy name Managed policy

Policy type

unicorn-s3-read-write

IAM Management Console https://console.aws.amazon.com/iam/home?region=us-west-2#roles/unicorns-challenge3-with-boundary

AWS Identity - Home Jade Sapphire Mong PmStuff AWS reInvent 2018 Audit Report (from A Travel Consistent Authoriza Salesforce - Unlimite

aws-reinvent-2018-unicorns-d... Global Support

Search IAM 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=unicorns-challenge3-with-boundary&account=128609111811

Permissions Trust relationships Tags Access Advisor Revoke sessions

Permissions policies (1 policy applied)

Attach policies Add inline policy

Policy name Managed policy

Policy type

unicorn-s3-read-write

Permissions boundary (set)

Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting but can be used to delegate permission management to others. Learn more

Change boundary Remove boundary

region-restriction (Managed policy)

We are using the region-restriction policy and the general s3-read-write policy.

The screenshot shows the AWS S3 Management Console interface. At the top, there are tabs for IAM Management Console, S3 Management Console, and others. The main header reads "Stream Video to AWS for Analytics—Easily capture, process, and store video streams for analytics and machine learning. Learn More > Documentation". On the left, a sidebar titled "Amazon S3" has a "Buckets" section. It says "Public access settings for this account:" and lists four buckets: "cloudtrail-awslogs-128609111811-jxsnm11h-isengard-do-not-del...", "do-not-delete-gatedgarden-audit-128609111811", "reinvent-challenge3-fail-demo-2018", and "reinvent-challenge3-success-demo-2018". Below the sidebar is a search bar and a dropdown menu for "All access types". A table displays the bucket details:

Bucket name	Access	Region	Date created
cloudtrail-awslogs-128609111811-jxsnm11h-isengard-do-not-del...	Objects can be public	US East (N. Virginia)	Nov 18, 2018 6:06:41 PM GMT-0800
do-not-delete-gatedgarden-audit-128609111811	Objects can be public	US West (Oregon)	Nov 18, 2018 6:21:32 PM GMT-0800
reinvent-challenge3-fail-demo-2018	Bucket and objects not public	EU (London)	Nov 20, 2018 12:19:22 PM GMT-0800
reinvent-challenge3-success-demo-2018	Bucket and objects not public	US West (Oregon)	Nov 20, 2018 12:18:56 PM GMT-0800

At the bottom, there are links for "Feedback", "English (US)", and copyright information: "© 2006–2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use".

We also have 2 S3 buckets for the fail and the success scenarios

```
C:\Users\brigidj\Desktop\reInvent2018>aws s3api put-object --bucket reinvent-challenge3-success-demo-2018 --key ./pickles.jpg --body pickles.jpg --profile challenge2-regions
{
    "ETag": "\"d45a3ae9dc9ac57a5471281d4d4080d9\""
}
```

We will use the CLI to PUT object in the success object in the west region, it worked.

```
C:\Users\brigidj\Desktop\reInvent2018>aws s3api put-object --bucket reinvent-challenge3-fail-demo-2018 --key ./pickles.jpg --body pickles.jpg --profile challenge2-regions
An error occurred (AccessDenied) when calling the PutObject operation: Access Denied
C:\Users\brigidj\Desktop\reInvent2018>
```

When we try to PUT into the fail bucket in the London region, it fails because of the permission boundary is preventing us from calling outside of the US west region.

Let's see permission boundaries in action

- Using the developer role, create a role with a permission boundary
- Use a role with a permission boundary to put data in S3 for approved regions and for unapproved regions.

Challenge #4

Enable developers working on the Dorky project and the Sneaky project to manage their own resources without also managing the other project's.

Pro Tip: Carefully consider the tag keys you want to use for authorization

We are still inside the same account but the unicorn team is broken into 2 different teams for the Dorky and Sneaky projects that need to manage their own resources and not share resources. We are going to **use tag-based access control** in this case to scale our permissions management. You need to control your tags management.

Three parts required for tag-based access control

 Allow users to create tags when creating resources, but require specific tags when users create resources

`RequestTag` condition to require specific tag value during create actions

 Control which existing resources and values developers can tag

Use a combination of `RequestTag` and `ResourceTag` control access

 Control resources users can manage based on tag values

`ResourceTag` to control access to resources based on a tag that exists on a resource

Policy for challenge #4

```
"Effect": "Allow",
  "Action": [
    "ec2:RunInstances"
  ],
  "Resource": [
    "arn:aws:ec2:*:*:subnet/*",
    "arn:aws:ec2:*:*:key-pair/*",
    "arn:aws:ec2:*:*:snapshot/*",
    "arn:aws:ec2:*:*:launch-template/*",
    "arn:aws:ec2:*:*:volume/*",
    "arn:aws:ec2:*:*:security-group/*",
    "arn:aws:ec2:*:*:placement-group/*",
    "arn:aws:ec2:*:*:network-interface/*",
    "arn:aws:ec2:*:*:image/*"
  ],
  "Condition": {
    "StringEquals": {
      "aws:RequestedRegion": ["us-west-1", "us-west-2"]
    }
  }
```

Policy for challenge #4

Allow for creation of tags when creating new resources, but...

```
"Effect": "Allow",
"Action": "ec2:CreateTags", —————— Allows creation of tags
"Resource": "*",
"Condition": {
    "StringEquals": {
        "ec2:CreateAction": "RunInstances" —————— But only during
    } RunInstances calls
}
}
```

Policy for challenge #4

...require specific tags when users create new resources

```
"Effect": "Allow",
>Action": [
    "ec2:RunInstances"
],
"Resource": [
    "arn:aws:ec2:*:*:instance/*"],
"Condition": {
    "ForAllValues:StringEquals": {
        "aws:TagKeys": ["project", "name"] —————— Allows project and/or
    } name, but nothing else
    "StringEquals": {
        "aws:RequestTag/project": ["dorky"], —————— Requires project tag
        "aws:RequestedRegion": ["us-west-1", "us-west-2"] —————— and must be this value
    }
}
}
```

Policy for challenge #4

Control which existing resources and values developers can tag

```
"Effect": "Allow",
"Action": "ec2:CreateTags",
"Resource": "*",
"Condition": {
    "StringEquals": {
        "ec2:ResourceTag/project": ["dorky"] —————— Only tag resources with
    } these tags
    "ForAllValues:StringEquals": {
        "aws:TagKeys": ["project", "name"] —————— Tag with either of these
    } keys
    "StringEqualsIfExists": {
        "aws:RequestTag/project": ["dorky"]
    }
}
}
```

Policy for challenge #4

Control resources users can manage based on tag values

```
"Effect": "Allow",
"Action": [
    "ec2:StartInstances",
    "ec2:StopInstances"],
"Resource": "*",
"Condition": {
    "StringEquals": {
        "ec2:ResourceTag/project": "dorky"
    }
}
```

Only manage resources with these tags

Let's see tag-based access control in action

- Launch instances for project dorky
- Try to launch instances for project sneaky
- Modify tags on existing instances (dorky and sneaky)
- Manage existing instances

The screenshot shows the AWS IAM Management Console with a search bar at the top containing the text 'challenge'. Below the search bar, there is a table listing six roles:

Role name	Description	Trusted entities
challenge1-admin	This role is used to test challenge one for SCP restrictions with admin...	Account: 332207979596
challenge2-regions	This role is used to demonstrate creating resources in unapproved r...	Account: 332207979596
challenge3-permission-boundaries	This role is used to demonstrate creating roles with permission boun...	Account: 332207979596
challenge4-tag-permissions	This role is used to demonstrate using tags to control access.	Account: 332207979596
challenge5-match-tags	This role will be used to demonstrate how you can use matching tags ...	Account: 332207979596
unicorns-challenge3-with-boundary	This role is used to demonstrate permission boundaries	Account: 332207979596

The screenshot shows the AWS IAM Management Console with the URL <https://console.aws.amazon.com/iam/home?region=us-west-2#/roles/challenge4-tag-permissions>. The page displays the details of the 'challenge4-tag-permissions' role. Key information includes:

- Role ARN:** arn:aws:iam::128609111811:role/challenge4-tag-permissions
- Role description:** This role is used to demonstrate using tags to control access.
- Instance Profile ARNs:** None
- Path:** /
- Creation time:** 2018-11-20 10:55 PST
- Maximum CLI/API session duration:** 1 hour
- Give this link to users who can switch roles in the console:** <https://signin.aws.amazon.com/switchrole?roleName=challenge4-tag-permissions&account=128609111811>

The 'Permissions' tab is selected, showing three policies applied:

- dorky-unicorn-project-access (Managed policy)
- AmazonEC2ReadOnlyAccess (AWS managed policy)
- passRole-permissions (Managed policy)

This screenshot shows the same IAM role summary page, but with a different set of attached policies:

- dorky-unicorn-project-access (Managed policy)
- AmazonEC2ReadOnlyAccess (AWS managed policy)
- passRole-permissions (Managed policy)

The 'Permissions boundary (not set)' section is visible at the bottom of the list.

We now have the **dorky-unicorn-project-access** policy added to allow just the Dorky developers do things in their account only. We then also gave the general **AmazonEC2ReadOnlyAccess** policy that allows to do EC2 related things like list or runInstances from the EC2 console.

IAM Management Console S3 Management Console

https://console.aws.amazon.com/iam/home?region=us-west-2#roles/challenge4-tag-permissions

AWS Identity - Home Jade Sapphire Mong PmStuff AWS re:Invent 2018 Audit Report (from A P Travel Consistent Authoriza Salesforce - Unlim...

Services Resource Groups

aws-reinvent-2018-unicorns-d... Global Support

Search IAM

Attach policies

Add inline policy

Policy name: dorky-unicorn-project-access Policy type: Managed policy

Policy summary JSON Edit policy Simulate policy

```
1: {
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:RunInstances"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:subnet/*",
        "arn:aws:ec2:*:*:key-pair/*",
        "arn:aws:ec2:*:*:snapshot/*",
        "arn:aws:ec2:*:*:launch-template/*",
        "arn:aws:ec2:*:*:volume/*",
        "arn:aws:ec2:*:*:security-group/*",
        "arn:aws:ec2:*:*:placement-group/*",
        "arn:aws:ec2:*:*:volume/*"
      ]
    }
  ]
}
```

IAM Management Console S3 Management Console

https://console.aws.amazon.com/iam/home?region=us-west-2#roles/challenge4-tag-permissions

AWS Identity - Home Jade Sapphire Mong PmStuff AWS re:Invent 2018 Audit Report (from A P Travel Consistent Authoriza Salesforce - Unlim...

Services Resource Groups

aws-reinvent-2018-unicorns-d... Global Support

Search IAM

Attach policies

Add inline policy

Policy name: dorky-unicorn-project-access Policy type: Managed policy

Policy summary JSON Edit policy Simulate policy

```
1: {
  "Version": "2012-10-17",
  "Statement": [
    {
      "Condition": {
        "StringEquals": {
          "aws:RequestedRegion": [
            "us-west-1",
            "us-west-2"
          ]
        }
      },
      "Effect": "Allow",
      "Action": [
        "ec2:RunInstances"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:instance/*"
      ]
    }
  ]
}
```

IAM Management Console S3 Management Console

https://console.aws.amazon.com/iam/home?region=us-west-2#roles/challenge4-tag-permissions

AWS Identity - Home Jade Sapphire Mong PmStuff AWS re:Invent 2018 Audit Report (from A P Travel Consistent Authoriza Salesforce - Unlim...

Services Resource Groups

aws-reinvent-2018-unicorns-d... Global Support

Search IAM

Attach policies

Add inline policy

Policy name: dorky-unicorn-project-access Policy type: Managed policy

Policy summary JSON Edit policy Simulate policy

```
1: {
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:RunInstances"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:instance/*"
      ],
      "Condition": {
        "ForAllValues:StringEquals": {
          "aws:TagKeys": [
            "project",
            "name"
          ]
        }
      },
      "StringEquals": {
        "aws:TagKey": "name"
      }
    }
  ]
}
```

The screenshot shows the AWS EC2 Management Console interface. The left sidebar lists navigation options like EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, Images, AMIs, Bundle Tasks, and Elastic Block Store Volumes. The main content area displays a table of instances. The first instance, i-030e2230b260639d3, is selected. Below it, the instance details are shown in a modal window.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
	i-030e2230b260639d3	t2.micro	us-west-2a	running	2/2 checks ...	None	ec2-35-160-249-22.us...
	i-09e66bd1d80906ea5	t2.micro	us-west-2a	running	2/2 checks ...	None	ec2-35-160-249-22.us...

Instance: i-030e2230b260639d3 Private IP: 172.31.43.54

Description	Status Checks	Monitoring	Tags
Instance ID: i-030e2230b260639d3	Instance state: running	Instance type: t2.micro	Elastic IPs: -
Availability zone: us-west-2a	Security groups: default, view inbound rules, view outbound	Public DNS (IPv4): -	IPv4 Public IP: -
		IPv6 IPs: -	Private DNS: ip-172-31-43-54.us-west-2.compute.internal
		Private IPs: 172.31.43.54	Secondary private IPs: -

The screenshot shows the AWS EC2 Management Console interface. The left sidebar includes links for EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, and Images. The main content area displays a table of instances. The table has columns for Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, and Public DNS (IPv4). Two instances are listed:

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
	i-030e2230b260639d3	t2.micro	us-west-2a	running	2/2 checks ...	None	ec2-35-160-249-22.us...
	i-09e66bd1d80906ea5	t2.micro	us-west-2a	running	2/2 checks ...	None	ec2-35-160-249-22.us...

Below the table, details for the first instance are shown: Instance ID i-030e2230b260639d3, Private IP 172.31.43.54. A tab bar below the instance details shows Description, Status Checks, Monitoring, and Tags, with Tags selected. A "Add/Edit Tags" button is present. A table for tags shows one entry: Key project and Value dorky. Navigation icons for back, forward, and search are at the top right.

Step 7: Review Instance Launch

Please review your instance launch details. You can always change them later.

AMI Details

Amazon Linux 2 AMI 2.0.20181114 x86_64 (HVM, SSD Volume Type)

Instance Type

Instance Type	ECUs	VCPUs	Memory (GiB)	Storage (GiB)	Network (Mbps)	Local NVMe (GiB)	Local SSD (GiB)	Virtualization Type
t2.micro	Variable	1	1.0	10	100	0	0	None

Security Groups

Security Group ID: sg-e97c8f9d

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI

Choose an existing key pair: Select a key pair: ec2-reinvent-demo

I acknowledge that I have access to the selected private key file (ec2-reinvent-demo.pem), and that without this file, I won't be able to log into my instance.

Launch Instances

EC2 Management Console

Create billing alerts: to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

How to connect to your instances

Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

Click [View Instances](#) to monitor your instances' status. Once your instances are in the **running** state, you can [connect](#) to them from the instances screen. [Find out](#) how to connect to your instances.

Here are some helpful resources to get you started

- How to connect to your Linux instance
- Amazon EC2: User Guide
- Learn about AWS Free Usage Tier
- Amazon EC2: Discussion Forum

While your instances are launching you can also

- Create status check alarms to be notified when these instances fail status checks. (Additional charges may apply)
- Create and attach additional EBS volumes (Additional charges may apply)
- Manage security groups

[View Instances](#)

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

EC2 Management Console

Launch Instance Connect Actions

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4
i-030e2230b260639d3	i2.micro	us-west-2a	running	2/2 checks ...	None	green	-	-
i-086c5194146718a29	i2.micro	us-west-2a	pending	initializing	None	yellow	-	-
i-09e6bd1d80906ea5	i2.micro	us-west-2a	running	2/2 checks ...	None	green	ec2-35-160-249-22.us... 35.160.249.22	35.160.249.22

Instance: i-030e2230b260639d3 Private IP: 172.31.43.54

Description Status Checks Monitoring Tags

Add/Edit Tags

Key	Value
project	dorky

Show Column

We are allowed to launch instances within the US-WEST region specified in the policy. We then try to launch another instance with the tag **sneaky** as above

EC2 Management Console

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

AMI Details

amzn2-ami-hvm-2.0.20181114-x86_64-gp2 - ami-01bbe152bf19d0289

Amazon Linux 2 AMI 2.0.20181114 x86_64 HVM gp2

Root Device Type: sda Virtualization type: hvm

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
i2.micro	Variable	1	1	EBS only	-	Low to Moderate

Security Groups

Security Group ID	Name	Description
sg-e97c8f9d	default	default VPC security group

Cancel Previous **Launch** Define key pair and launch

Launch Failed

You are not authorized to perform this operation. Encoded authorization failure message: PUXODerRuh4uvuzWEbVLMqWlsLH3PSBYayp98cYsx50KEvKscIN-Sv1stcWQmWhz2yFCtMVFCHU7Oj6_fuyWij0JUlipmrhOrJZgluYiwYLix21kx0f-lqEMWn1u5yjA2qzlKx20VVdu12wVtcAyUCp4-pfIClAEoy3y09MDDVQuGVsUsstaZOWellQsUfCgCGP3_g5G3-atVtK_OdcFBk3mBj3-Xp4l6Emig26n0f0DhyJ0tK0tsuvQ79C0_5G771KD8DTezhphsMUrJsd3r1eFmV2kQoQ9rxXfzlzDpfbu7MdPTWLJf7UgARN86mgrSFMNFGz7mlUj2ga-eRy2k8ghlyK9gNojTmeATS-m2vSGEYV0VvWHPjubBPbjarG71WAwVHeUtygGMkkDDeFPM82F3u5_cDzOIQS3Pb7z9PmbBkr8Aaad-3cgMo_09PNU3CPvZlFSzT09ysGNqd9N7eyGK0cv9xyMG7wEC5C-rWKg46k0Hd9hr-nHnHu-gz4YfUpXRCCRJUow7_MkEqZ93USNCQaygbHZW9bQTzWYmx40F4sSRNks6q7mNec80JOF06GEd5EqjYwmPdvuAuyDjmBX3QKzaygSPkzQcvzDix_MdkUv7Hw-XOpZPZwdIM_0T2BK0hThwUBUC063e_cmlJ_U1vhfIUZeePauB5pw65lSzKgfZ9V25K42lScGHQbOrU-8

[Hide launch log](#)

Initiating launches Failure [Retry](#)

[Cancel](#) [Back to Review Screen](#) [Retry Failed Tasks](#)

It failed because the Dorky team is not allowed to launch instances with tags starting with 'sneaky'.

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.

A copy of a tag can be applied to volumes, instances or both.

Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key	(127 characters maximum)	Value	(255 characters maximum)	Instances	Volumes
project	dorky			<input checked="" type="checkbox"/>	<input type="checkbox"/>
brngid	Isawesome			<input checked="" type="checkbox"/>	<input type="checkbox"/>

[Add another tag](#) (Up to 50 tags maximum)

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Security Group](#)

We try to launch another instance with the above tags again

Launch Failed

You are not authorized to perform this operation. Encoded authorization failure message:

```
TnW1NFQKyGBV0GDIhNkjl8g2y675kgPU7ejal4vV2FX3Mcu1VbQIPUcf4Dk0t2OsXrVmEvCtyLs9CqGp9kSVjZn3lUh0qmOhxbGEctCMoZeK1L3PpVTwRS7JYveAia8t9s9uYbigW67sphWKe3eQOL3laCWSmXm6D6gYaAAxsC22r3BaIYYVm0z0Rxw7m0gEMMnsa7kYcgvhnZPykdrCmCSL-D62BDYmDxWWLcIT8Ldtnc981go8CNT4l197Kqdadj_xSrqt77fIsdeJxJ.AeqLhsPK90yPQHQzPmQX68edFnA6FBZel7-CXkIWb912KUSH_kgfwdDheJMC8Q7FICPEbrGSpGBnS0x2JgVT0WcQ7-GQ7sAgjBnkSAc3lpBok8nHux822FcOpCAP4jCEGydgQRCrre8jRv-a6MQ3du8Avm1_e9ya3z4pvqUzjyYD_B3TVhkrhdj3L3L_d_lzCuyGu5z51GPI4cgDT6bGqpkwQGzaG7uC4BVztl_kj01YPD5dJErslRaw1BJDFIA7Pxj9leVtzG9GKTzA4Uj1-FbvIqKCnvU4vHunfyy-wDpYmBml0ltcs5rPlWCxWVGSyS.SobYJxeiKL3SDwH8Q4VU_s9GEr7hJu_A77Azg1Btqyfjxt12nxC5XVb2nwrxKLDYkcSe0Xai7HrnGEYwF7BOrh3GWpN23mtzEm8q6-kdzFPrdjs
```

[Hide launch log](#)

Initiating launches Failure [Retry](#)

[Cancel](#) [Back to Review Screen](#) [Retry Failed Tasks](#)

This failed because we can only have **project** and **name** tags and the project tag must have the value of **dorky** only.

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.

A copy of a tag can be applied to volumes, instances or both.

Tags will be applied to all instances and volumes. Learn more about tagging your Amazon EC2 resources.

Key	(127 characters maximum)	Value	(255 characters maximum)	Instances	Volumes
project	dorky	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
name	isawesome	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	

Add another tag (Up to 50 tags maximum)

Cancel **Previous** **Review and Launch** **Next: Configure Security Group**

Feedback **English (US)** © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. **Privacy Policy** **Terms of Use**

Launch Status

Your instances are now launching. The following instance launches have been initiated: i-0854bbc7f8708edf4 [View launch log](#)

Get notified of estimated charges Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

How to connect to your instances

Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

Click [View Instances](#) to monitor your instances' status. Once your instances are in the **running** state, you can [connect](#) to them from the Instances screen. Find out how to connect to your instances.

Here are some helpful resources to get you started

- How to connect to your Linux instance
- Learn about AWS Free Usage Tier
- Amazon EC2 User Guide
- Amazon EC2 Discussion Forum

While your instances are launching you can also

Feedback **English (US)** © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. **Privacy Policy** **Terms of Use**

It worked.

EC2 Management Console [Launch Instance](#) [Connect](#) [Actions](#)

Events
Tags
Reports
Limits
INSTANCES
Instances
Launch Templates
Spot Requests
Reserved Instances
Dedicated Hosts
Scheduled Instances
Capacity Reservations
IMAGES
AMIs
Bundle Tasks
ELASTIC BLOCK STORE
Volumes

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4
i-030e2230b260639d3	t2.micro	us-west-2a	running	2/2 checks ...	None		-	-
i-0854bbc7f8708edf4	t2.micro	us-west-2a	pending	Initializing	None		ec2-34-223-2-200.us-west-2...	34.2
i-086c5194146718a29	t2.micro	us-west-2a	running	Initializing	None		-	-
i-09e66bd1d80906ea5	t2.micro	us-west-2a	running	2/2 checks ...	None		ec2-35-160-249-22.us-west-2...	35.1

Instance: i-09e66bd1d80906ea5 Public DNS: ec2-35-160-249-22.us-west-2.compute.amazonaws.com

[Description](#) [Status Checks](#) [Monitoring](#) **Tags**

Add/Edit Tags

Key	Value
project	sneaky

Show Column

EC2 Management Console

Services Resource Groups

EC2 Dashboard Events Tags Reports Limits

INSTANCES Instances Launch Templates Spot Requests Reserved Instances Dedicated Hosts Scheduled Instances Capacity Reservations

IMAGES AMIs Bundle Tasks

ELASTIC BLOCK STORE Volumes

Launch Instance Connect Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4
i-030e2230b260639d3	t2.micro	us-west-2a	running	2/2 checks ...	None	None	-	-
i-0854bcb7f708ed4	t2.micro	us-west-2a	pending	Initializing	None	None	ec2-34-223-2-200.us-w...	34.2...
i-086c5194146718a29	t2.micro	us-west-2a	running	2/2 checks ...	None	None	-	-
i-09e66bd1d80906ea5	t2.micro	us-west-2a	running	2/2 checks ...	None	None	ec2-35-160-249-22.us...	35.1...

Connect Get Windows Password Create Template From Instance Launch More Like This

Description Status Checks Monitoring Tags

Add/Edit Tags

Key project Value sneaky

1 to 4 of 4

Feedback English (US)

© 2008–2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Let us try and stop an instance that belongs to the Sneaky team

EC2 Management Console

Services Resource Groups

EC2 Dashboard Events Tags Reports Limits

INSTANCES Instances Launch Templates Spot Requests Reserved Instances Dedicated Hosts Scheduled Instances Capacity Reservations

IMAGES AMIs Bundle Tasks

ELASTIC BLOCK STORE Volumes

Feedback English (US)

Stop Instances

Are you sure you want to stop these instances?

i-09e66bd1d80906ea5

Note that when your instances are stopped:
Any data on the ephemeral storage of your instances will be lost.

Error stopping instances

You are not authorized to perform this operation. Encoded authorization failure message: sY3VzeIDWemppIAQHsu7cV6XUpjeJYz82X8WMlkl-6UQc-ue8C0CC0j6i3RQPORoCU-IRgsGUEzr1WhZ-TNwaz05D-pY7Es8gJus68UA779AMqvhIr35G_o4MAdKbcP4E6JF9QEmcD0oQ4_Twhu8e1MuNxE2he7TpJqJNVnZziyycCnQ8XEHnTgl800690KgdhB0s6egxGzKtR_qkV19pChIayzvzbhTDaDE4djhAsuCX130k092S0uw3PumCb5tvJz5dloGez4TZY-LffId_ezeVkbAxSiKG0dmgf87-EjT-ur4mdPKVYET4SWBpzIPVnEoasD4tg63mVrC4T77Cx-L7k-Qr-FnICe_NV6kWjky_3GT8VP2jBS_Loj8i42UqO02xltbWS0mchku5Ecwvn6VUsfDjAYEPVfQ6srwrtTuzkLuhfmxXvhJvHQQle-b-ses69ALkjSodO4lkBF7GMgb7csnfSRNj20lg2Nz-jmrO1rFDePfskWDM7hjGSNIofEPID7ABVfkq9fbz-d9f2d2c3telfcdBdZltfGQhelylkTQZZL-14ZBllnVF0ZzVzbhkn3W2ChBnb14khWJWfURRrre6C1thzn7B51N3u7PPPwsNiRAQWfjwmlRV6Sa-9WFCoargFvnYR-k0lf42pDTgRbbuHYYlqsmYAeGnGrnchJLrmMgrbU EgmqAvxtLxdnTcmrwRoxzCYSuLU8n5QKVQfkqCIBwN1jk-

Cancel Yes, stop

1 to 4 of 4

Privacy Policy Terms of Use

We cannot stop instances that do not belong to the Dorky team that always has the project equals dorky tags.

EC2 Management Console

Services Resource Groups

EC2 Dashboard Events Tags Reports Limits

INSTANCES Instances Launch Templates Spot Requests Reserved Instances Dedicated Hosts Scheduled Instances Capacity Reservations

IMAGES AMIs Bundle Tasks

ELASTIC BLOCK STORE Volumes

Launch Instance Connect Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4
i-030e2230b260639d3	t2.micro	us-west-2a	running	2/2 checks ...	None	None	-	-
i-0854bcb7f708ed4	t2.micro	us-west-2a	pending	Initializing	None	None	ec2-34-223-2-200.us-w...	34.2...
i-086c5194146718a29	t2.micro	us-west-2a	running	2/2 checks ...	None	None	-	-
i-09e66bd1d80906ea5	t2.micro	us-west-2a	running	2/2 checks ...	None	None	ec2-35-160-249-22.us...	35.1...

Connect Get Windows Password Create Template From Instance Launch More Like This

Description Status Checks Monitoring Tags

Add/Edit Tags

Key project Value sneaky

Create Tag Cancel Save

Key project Value sneaky

1 to 4 of 4

Feedback English (US)

© 2008–2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

The screenshot shows the AWS EC2 Management Console interface. On the left, there's a sidebar with various navigation links like EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, AMIs, Bundle Tasks, and EBS Volumes. The main area shows a list of instances with columns for Name, Alarm Status, Public DNS (IPv4), and IPv4. One instance, 'i-09e86bd1', is selected. A modal window titled 'Add/Edit Tags' is open over the list. Inside the modal, there's a table with 'Key' and 'Value' columns. Two rows are present: one for 'project' with value 'sneaky' and another for 'name' with value 'brigid'. At the bottom of the modal are 'Create Tag', 'Cancel', and 'Save' buttons, with 'Save' being the active one.

This screenshot is similar to the previous one but shows a different outcome. The 'Add/Edit Tags' dialog box is still open, but the 'Save' button is now disabled and grayed out. A large, detailed error message is displayed in a separate box below the dialog. The message is a long string of encoded characters, starting with '7KKZCN0hrmqEcDmogYeYWYo36nAl1hmVUAsWYs5GN4L-HC1cpAozlkTKhf_pCT-ipz8S1yLy7aNhFIPM1X2YK0w8TkH0ncIKlinccGpl7xELHuOKKL6BQqaIdQgJLEBWriZ0z0tVe-Ifs4Apd_Ze2DQR4decUJlCeaxwCsqDWK1eiEhzj52bfEo4_VuQF0gs7RvhznRsg6bc5KdAxqA2N1OxpbgHwtk0KBfm51ObW6sLgeKChQtpwYCjU9Pm83blduteGx0aJpAfFzy1phfQfWIM4QvA2ao2B21GRVxEcg'. It includes information about the service (AmazonEC2), status code (403), error code (UnauthorizedOperation), and request ID (ed5aea20-7f6e-40cb-ab91).

We also cannot rename or add tags to an instance that does not belong to our Dorky team.

Let's see tag-based access control in action

- Launch instances for project dorky
- Try to launch instances for project sneaky
- Modify tags on existing instances (dorky and sneaky)
- Manage existing instances

Bonus challenge

New! You can tag IAM users and roles

Create a general policy that allows read access to secrets

tagged with a role tag.

Pro Tip: Any condition key can also be used as a variable as a condition value (the right-hand side) for string operators

`["${aws:PrincipalTag/project}"]`

This means that you can use the tag on your user like dorky and sneaky to do things in the policy.

Policy for challenge #5

...require specific tags when users create new resources

```
"Effect": "Allow",
"Action": [
    "ec2:RunInstances"
],
"Resource": [
    "arn:aws:ec2:*:*:instance/*"],
"Condition": {
    "ForAllValues:StringEquals": {
        "aws:TagKeys": ["project", "name"]
    },
    "StringEquals": {
        "aws:RequestTag/project": ["${aws:PrincipalTag/project}"],
        "aws:RequestedRegion": ["us-west-1", "us-west-2"]
    }
}
```

Allows project and/or name, but nothing else

Requires project tag and must be my project tag

Requires instance to be in approved region

We are using the same previous policy except that we are now requiring that the project tag (dorky or sneaky) must be passed into the policy using `${aws:PrincipalTag/project}`

Policy for challenge #5

Control which existing resources and values developers can tag

```
"Effect": "Allow",
"Action": "ec2:createTags",
"Resource": "*",
"Condition": {
    "StringEquals": {
        "ec2:ResourceTag/project": ["${aws:PrincipalTag/project}"]
    },
    "ForAllValues:StringEquals": {
        "aws:TagKeys": ["project", "name"]
    },
    "StringEqualsIfExists": {
        "aws:RequestTag/project": ["${aws:PrincipalTag/project}"]
    }
}
```

Only tag resources with my project ta

Tag with either of these keys

For project, you specify your project tag

You can also now create tags only on existing resources that have your project tag only, if you pass in a project tag, it must also be your project tag only.

Policy for challenge #5

Control resources users can manage based on tag values

```
"Effect": "Allow",
"Action": [
    "ec2:StartInstances",
    "ec2:StopInstances"],
"Resource": "*",
"Condition": {
    "StringEquals": {
        "ec2:ResourceTag/project": "${aws:PrincipalTag/project}"
    }
}
}
```

Only manage resources
with my project tag

The screenshot shows the AWS IAM Management Console with the search bar containing 'challenge'. Below the search bar, there is a table with columns: Role name, Description, and Trusted entities. The table lists several roles:

Role name	Description	Trusted entities
challenge1-admin	This role is used to test challenge one for SCP restrictions with admin...	Account: 332207979596
challenge2-regions	This role is used to demonstrate creating resources in unapproved r...	Account: 332207979596
challenge3-permission-boundaries	This role is used to demonstrate creating roles with permission boun...	Account: 332207979596
challenge4-tag-permissions	This role is used to demonstrate using tags to control access.	Account: 332207979596
challenge5-match-tags	This role will be used to demonstrate how you can use matching tags...	Account: 332207979596
unicorns-challenge3-with-boundary	This role is used to demonstrate permission boundaries	Account: 332207979596

The screenshot shows the AWS IAM Management Console with the URL <https://console.aws.amazon.com/iam/home?region=us-west-2#roles/challenge5-match-tags>. The left sidebar is visible with the 'Roles' option selected. The main area shows the 'Summary' tab for the 'challenge5-match-tags' role. The role ARN is listed as `arn:aws:iam::128609111811:role/challenge5-match-tags`. The role description states: 'This role will be used to demonstrate how you can use matching tags to control access to resources.' The instance profile ARNs section is empty. The creation time is listed as 2018-11-20 10:57 PST. The maximum CLI/API session duration is set to 1 hour. A link to switch roles is provided: <https://signin.aws.amazon.com/switchrole?roleName=challenge5-match-tags&account=128609111811>.

Path: /

Creation time: 2018-11-20 10:57 PST

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=challenge5-match-tags&account=128609111811

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

Permissions policies (3 policies applied)

Attach policies **Add inline policy**

Policy name	Policy type
AmazonEC2ReadOnlyAccess	AWS managed policy
passRole-permissions	Managed policy
role-tag-project-access	Managed policy

▶ Permissions boundary (not set)

We now have this as the **role-tag-project-access** policy

```

1 {
2     "Version": "2012-10-17",
3     "Statement": [
4         {
5             "Effect": "Allow",
6             "Action": [
7                 "ec2:RunInstances"
8             ],
9             "Resource": [
10                 "arn:aws:ec2:::subnet/*",
11                 "arn:aws:ec2:::key-pair/*",
12                 "arn:aws:ec2:::snapshot/*",
13                 "arn:aws:ec2:::launch-template/*",
14                 "arn:aws:ec2:::volume/*",
15                 "arn:aws:ec2:::security-group/*",
16                 "arn:aws:ec2:::placement-group/*",
17                 "arn:aws:ec2:::transit-gateway/*"
18             ]
19         }
20     ]
21 }
  
```

▶ Permissions boundary (not set)

```

1 {
2     "Resource": [
3         "arn:aws:ec2:::instance/*"
4     ],
5     "Condition": {
6         "ForAllValues:StringEquals": {
7             "aws:TagKeys": [
8                 "project",
9                 "name"
10             ]
11         },
12         "StringEquals": {
13             "aws:RequestTag/project": [
14                 "${aws:PrincipalTag/project}"
15             ],
16             "aws:RequestedRegion": [
17                 "us-west-1"
18             ]
19         }
20     }
21 }
  
```

▶ Permissions boundary (not set)

The screenshot shows the AWS IAM Policy Editor interface. On the left, a sidebar lists navigation options: Dashboard, Groups, Users, Roles (selected), Policies, Identity providers, Account settings, Credential report, and Encryption keys. The main area displays a policy named 'role-tag-project-access' under the 'Managed policy' section. It includes tabs for 'Policy summary' (selected), 'JSON', and 'Edit policy'. A large code editor window shows the JSON policy document:

```
    ],
    },
    "ForAllValues:StringEquals": {
        "aws:TagKeys": [
            "project",
            "name"
        ]
    },
    "StringEqualsIfExists": {
        "aws:RequestTag/project": [
            "${aws:PrincipalTag/project}"
        ]
    }
}
},
{
    "Effect": "Allow"
}
```

Below the code editor, a note states 'Permissions boundary (not set)'. At the bottom, there are links for 'Feedback', 'English (US)', and legal notices.

The screenshot shows the AWS IAM Management Console with two tabs open: 'IAM Management Console' and 'SS Management Console'. The main view is for the role 'challenge5-match-tags' in the 'us-west-2' region. The role has the ARN arn:aws:iam::128609111811:role/challenge5-match-tags. It includes a description: 'This role will be used to demonstrate how you can use matching tags to control access to resources.', and a path of '/'. The creation time is 2018-11-20 10:57 PST, and the maximum CLI/API session duration is 1 hour. A note says 'Give this link to users who can switch roles in the console' with a link to https://signin.aws.amazon.com/switchrole?roleName=challenge5-match-tags&account=128609111811. The 'Tags' tab is selected, showing one tag: 'project' with the value 'dorky'. There are tabs for 'Permissions', 'Trust relationships', 'Access Advisor', and 'Revoke sessions'. A search bar at the bottom left says 'Search tags by key or value (supports regex)'. The bottom navigation bar includes links for Feedback, English (US), Privacy Policy, and Terms of Use.

But we have added a tag for the project value to be dorky for this policy.

The screenshot shows the AWS EC2 Management Console interface. The left sidebar lists navigation options like EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, and Images. The Instances section is currently selected. The main content area displays a table of instances with columns for Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, Public DNS (IPv4), and IPv6. One instance row is selected, and a context menu is open over it. The menu items are: Connect, Get Windows Password, Create Template From Instance, Launch More Like This (which is highlighted in yellow), Instance State, Instance Settings, Image, Networking, and CloudWatch Monitoring.

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

AMI Details

Amazon Linux 2 AMI 2.0.20181114.x86_64 gp2	Root Device Type: ebs. Virtualization type: hvm
--------------------------------------------	-------------------------------------------------

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

Security Groups

Security Group ID	Name	Description
sg-e97c8f9d	default	default VPC security group

Launch

Feedback English (US)

Launch Status

Your instances are now launching. The following instance launches have been initiated: i-0000317fd830df662 [View launch log](#)

Get notified of estimated charges. Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

How to connect to your instances. Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances. Click [View Instances](#) to monitor your instances' status. Once your instances are in the **running** state, you can [connect](#) to them from the Instances screen. Find out how to connect to your instances.

Here are some helpful resources to get you started:

- How to connect to your Linux instance
- Amazon EC2 User Guide
- Learn about AWS Free Usage Tier
- Amazon EC2 Discussion Forum

While your instances are launching you can also

We can successfully launch instances with the dorky project tag.

EC2 Management Console

Launch Instance

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4
i-0000317fd830df662	i-0000317fd830df662	t2.micro	us-west-2a	pending	Initializing	None	-	-
i-030e2230b260639d3	i-030e2230b260639d3	t2.micro	us-west-2a	running	2/2 checks	None	-	-
i-0854bc7f708ed4	i-0854bc7f708ed4	t2.micro	us-west-2a	running	2/2 checks	None	ec2-34-223-2-200.us-west-2	34.2
i-086c5194146718a29	i-086c5194146718a29	t2.micro	us-west-2a	running	2/2 checks	None	-	-
i-09e66bd1d80906ea5	i-09e66bd1d80906ea5	t2.micro	us-west-2a	running	2/2 checks	None	ec2-35-180-249-22.us-west-2	35.1

Instance: i-09e66bd1d80906ea5 Public DNS: ec2-35-180-249-22.us-west-2

Tags

Add/Edit Tags

Key: project

Launch Now Like This

- Connect
- Get Windows Password
- Create Template From Instance
- Instance State
- Instance Settings
- Image
- Networking
- CloudWatch Monitoring

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

AMI Details

amzn2-ami-hvm-2.0.20181114-x86_64-gp2 - ami-01bbe152bf19d0289

Amazon Linux 2 AMI 2.0.20181114 x86_64 HVM gp2

Root Device Type: ebs Virtualization type: hvm

Edit AMI

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

Edit instance type

Security Groups

Security Group ID	Name	Description
sg-e97c8f9d	default	default VPC security group

Edit security groups

Launch

Feedback English (US)

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Launch Status

1 Launch Failed

You are not authorized to perform this operation. Encoded authorization failure message: UACV2UVEWCTGpZEGVZriukkoRqJ—mEovFdMn9AG4UPA3QNiCMw0B6Tfn8lMDDW45PKLk7btIDCbBio6mJMpjh8Pwgq4bQnoYYCa28o8j-shl.hrn6MxkrnX5TEwin4DR2AdxYsrO5dh5oprOoWou89/G4IZsvUJddL-YxIR5jSp_WK6j7LOfH-udtLz-k5l2SUINABYfk2iWFGeMklnDZEPJ70HR0_dPzWV1vDuEe9qa7HWWeH4En1x-S44DAAAZE8KwvPv3X0i2gyIGYsoVqz2lqrmwgGORH-Z3tluwMsveYzcgkC37U6q9PTXvU7wEnR3oZ3ShxEazjEnVOZCL1QDJ1alFb5l52nB2Nk4r4wSdAKnxJ3QS6lllVMjXNvUehRsDwoiKdbXPu8CuPOMhkTDvng3O-PDoxuvGrCnwvDFAuJnrgXk76R_nGi-qMnM.Ga932FpN9NxUd7dHR-CRjSD52M8rlrlyEoeuOpRnhZQtpzCPU2v4cJpGUf4ip85YmigkGoFBaqarPWGbVbK3PUM2X7Sk0l5Abx5an7oSDcMKLw84rtK5S7|RvGJMozk9puluwSpHG9LX5DHkms5JGku1qkp2-jtGZWH2G8NNzWYZ5hVMBhHRTq-80tbhXN5aokAdz2gikasRan3QN4prQ04VxeC-RM50n6fPHJUSKf_4P8ImuZHz-o2HMX9TE90-JoK0IC

Hide launch log

Initiating launches Failure **Retry**

Cancel Back to Review Screen Retry Failed Tasks

We cannot launch sneaky instances.

Items I didn't get to that I recommend

PrincipalOrgID

Use in a resource-based policy to only allow IAM principals from your organization to access resources.

Blog: [An easier way to control access to AWS resources by using the AWS organization of IAM principals](#)

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

re:Invent

© 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.

