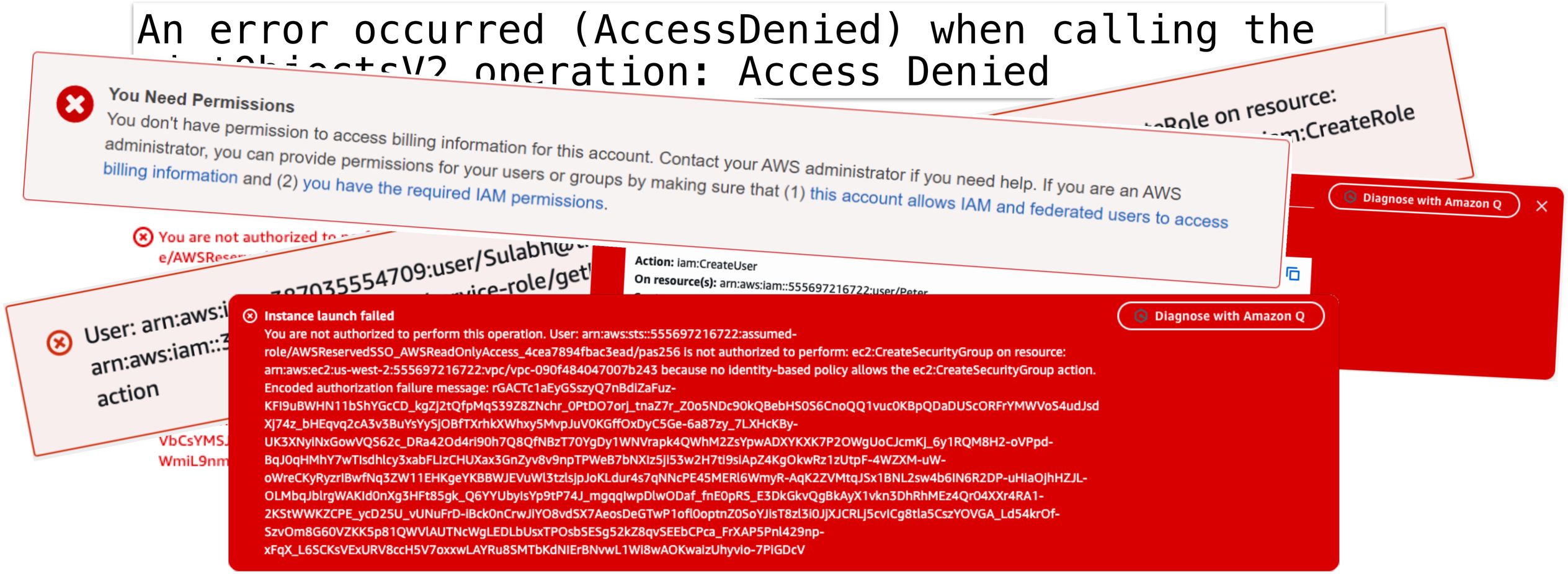
aws

COMMUNITY DAY

Access Denied: Understanding & Debugging AWS IAM

Peter Sankauskas | @pas256 | June 2025
he/him

Typical AWS usage



Replace this talk with Amazon Q

You don’t have sufficient permissions to Diagnose with Amazon Q.
You or your AWS administrator can grant access by adding the `AmazonQDeveloperAccess` policy to your IAM identity. For more information, see the AWS managed policies for Amazon Q .

Standard IAM error message format

```
User: arn:aws:sts::12345678901:sux/to-be-you
  is not authorized to perform
aws:SimpleAction
  on resource
arn:aws:ec2:us-tirefire-1:12345678901:service/abcd1234
  because
I haven't had my coffee yet.
```

Agenda

- How is IAM is designed?
- How does it evaluate policies?
- What are the different types of policies?
- When is each one useful?
- What is this error trying to tell me?
- What techniques can I use to debug permission errors?

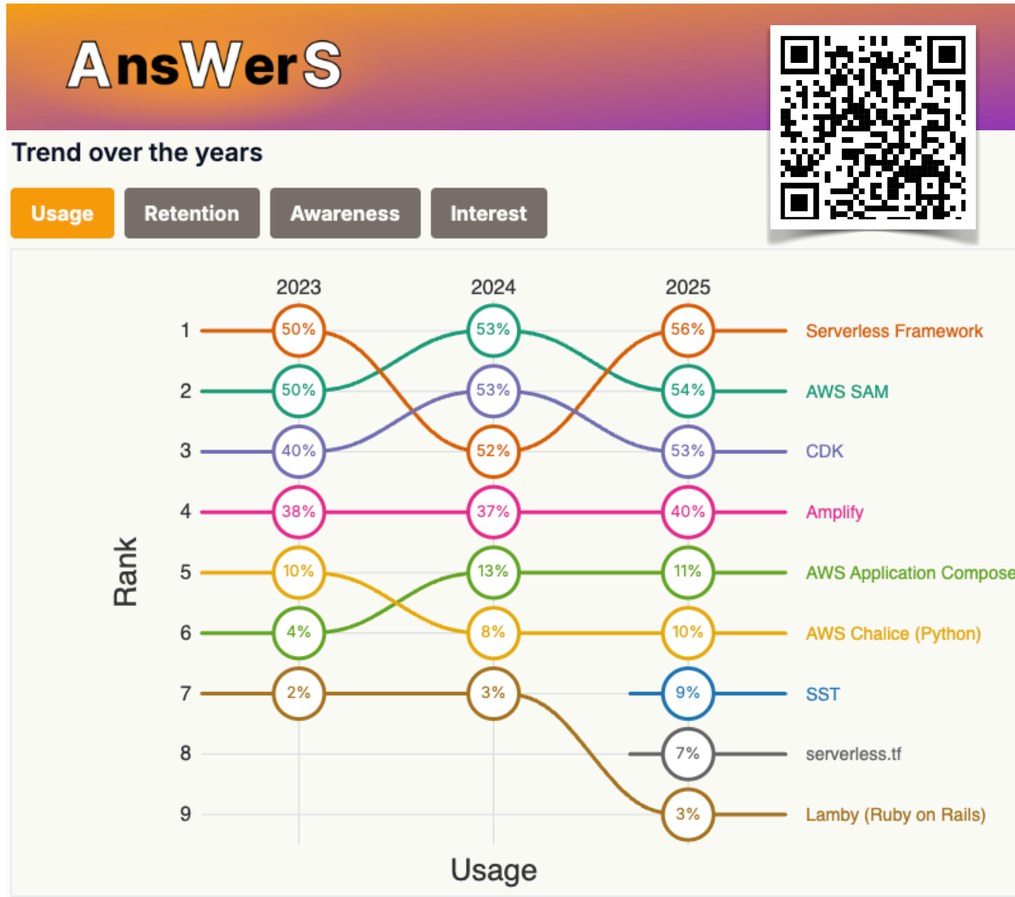
Peter Sankauskas

Strategic Engineering Hiring Partner

@ PAS Ventures

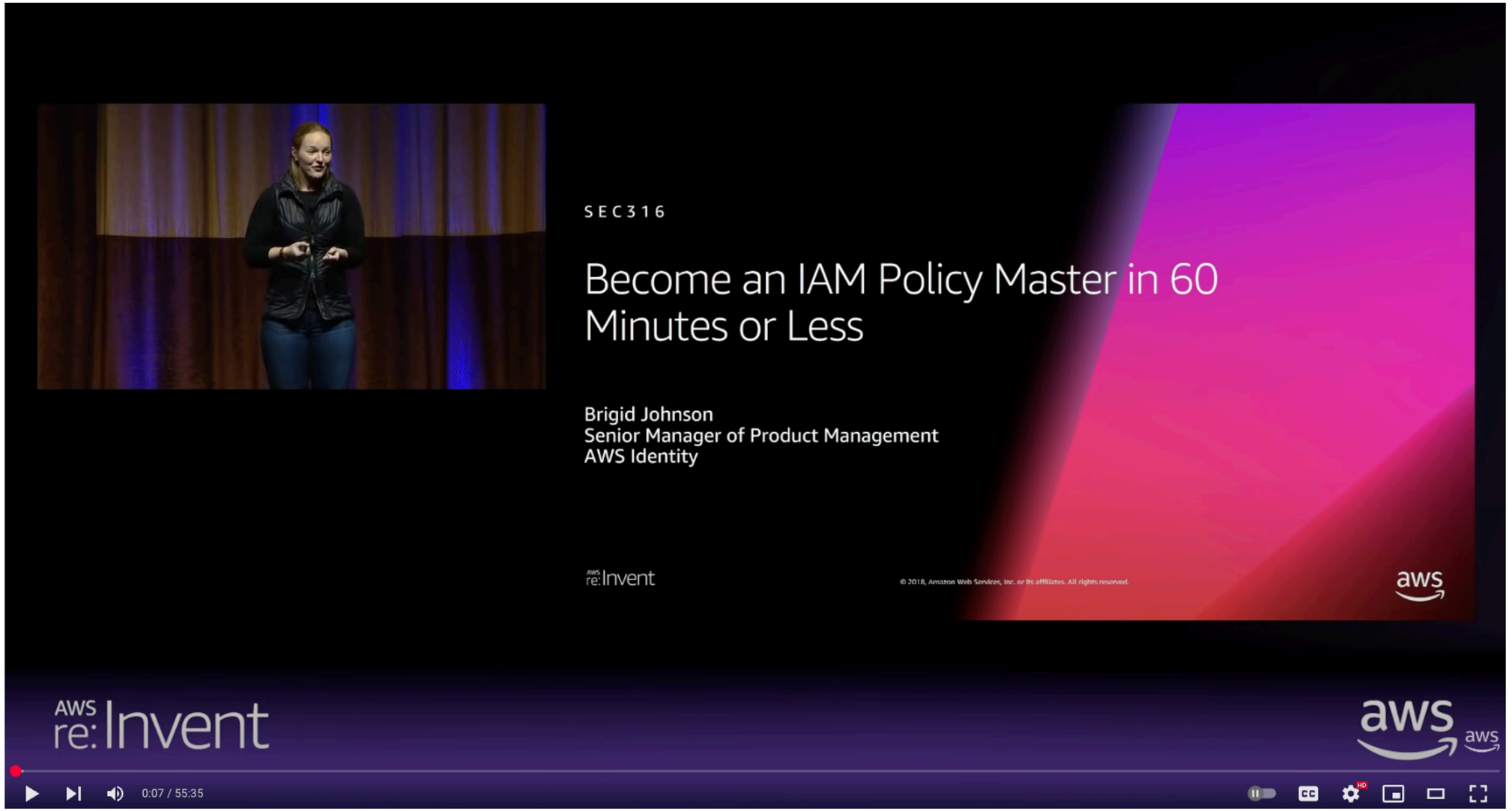
A former CTO, now helping you hire great engineers!

- AWS Community Hero since 2014
- Advanced AWS Meetup in SF
- Answers for AWS community survey



Inspiration

a.k.a great artist steel



What are IAM Policies?

1) **Specification**

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

2) **Enforcement**

   IAM enforces this access by ***evaluating*** the AWS request with the policies you defined and returns either allow or deny.

IAM Policy Language - PARC model

```
{
  "Statement":[{
    "Sid": "unique",
    "Effect": "effect",
    "Principal": "principal",
    "Action": "action",
    "Resource": "resource",
    "Condition": {
      "operator": { "key": "value" }
    }
  }]
}
```

Principal - *The entity that is allowed or denied access*

```
arn:aws:iam::123456789012:user/you
```

Action - *Type of access that is allowed or denied*

```
s3:PutObject
```

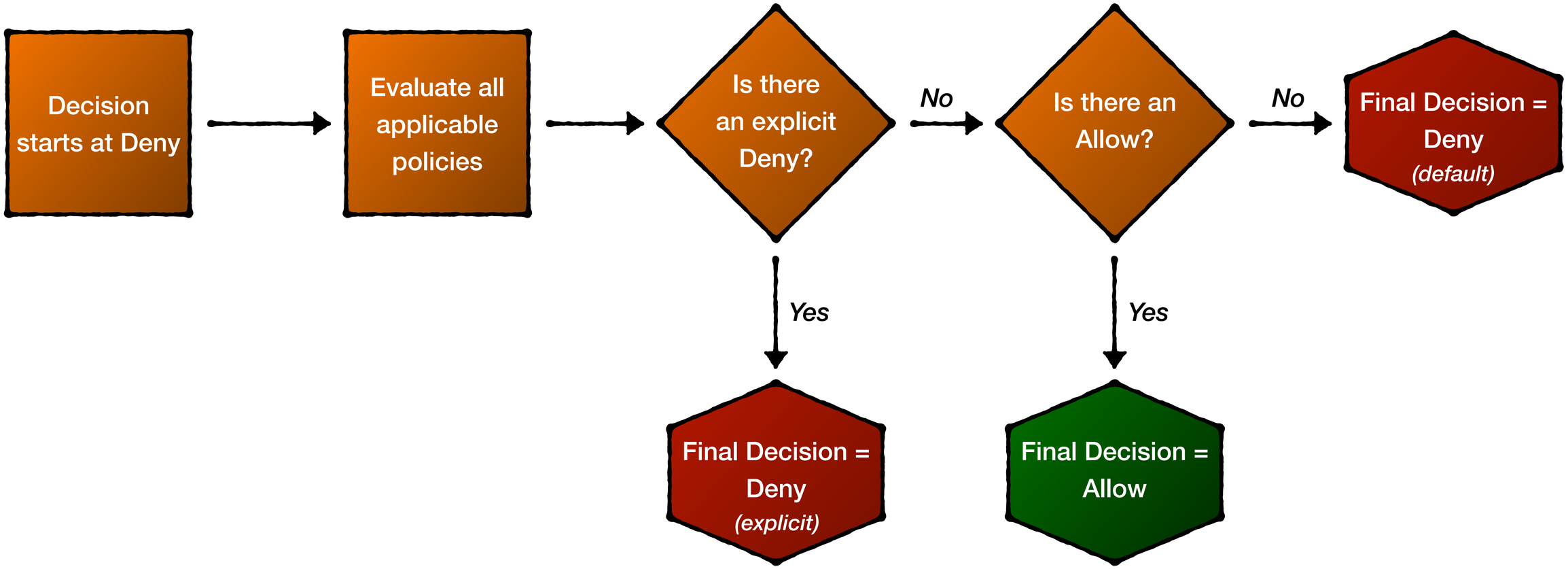
Resource - *The AWS resource(s) the action will act on*

```
arn:aws:s3:::my-bucket
```

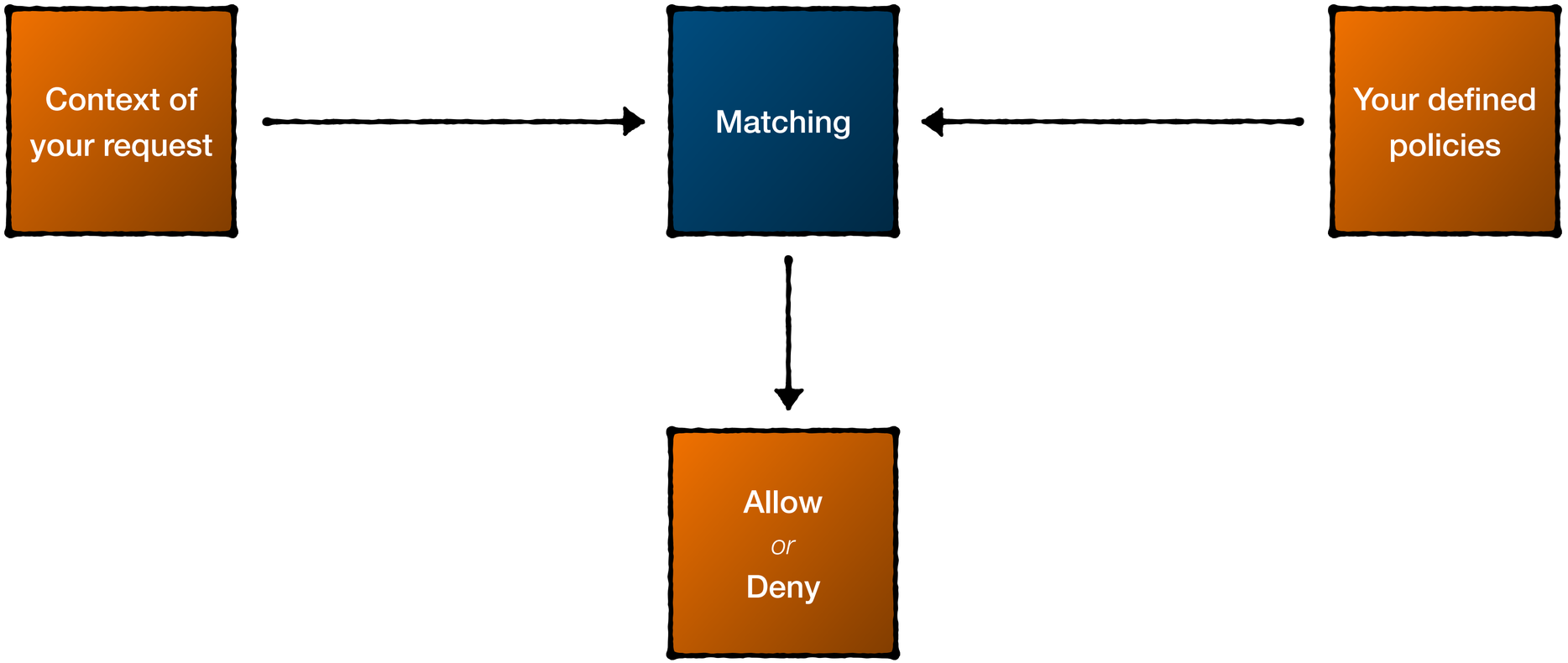
Condition - *But only under these conditions*

```
aws:MultiFactorAuthPresent = true
```

IAM Policy Evaluation



What's not matching?



Policy types

Identity-Based Policies

Trust Policies

VPC Endpoint Policies

Resource Control Policies

Service Control Policies

Resource-Based Policies

Permission Boundaries

Session Policies

Policy types

Identity-Based Policies

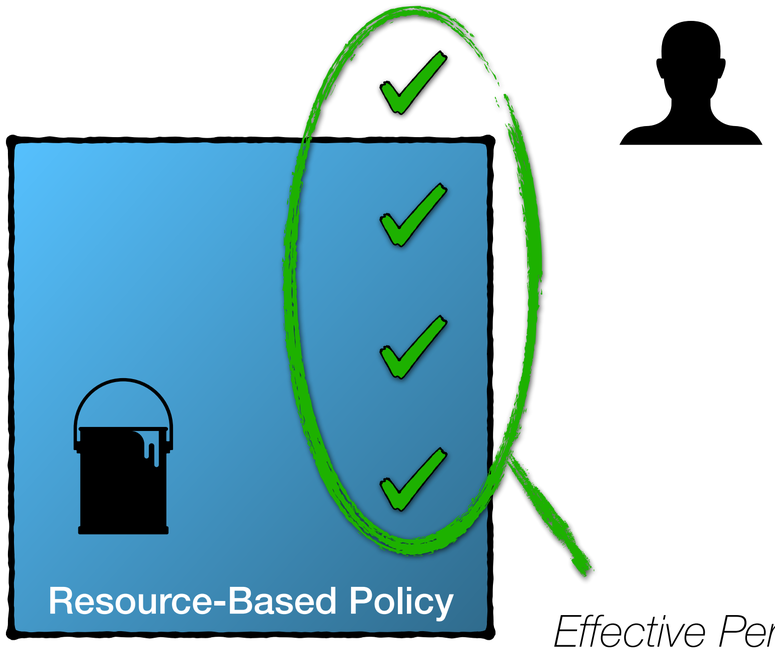
Attach to a Principal

e.g. User, Group, Role, Session

Resource-Based Policies

Attach to a Resource

e.g. S3 Bucket, VPC Endpoint



Effective Permissions
= UNION

**within an AWS Account*

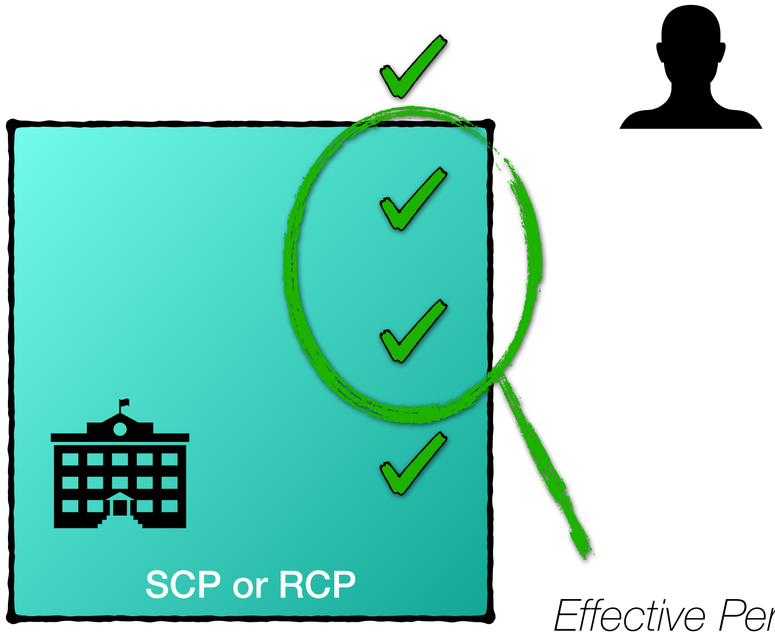
Organization policy types

Service Control Policies (SCP)

- Getting more common
- Set at the Organization level
- Limit permissions granted to Identities

Resource Control Policies (RCPs)

- New - released Dec 2024
- Set at the Organization level
- Limit permissions granted on Resources

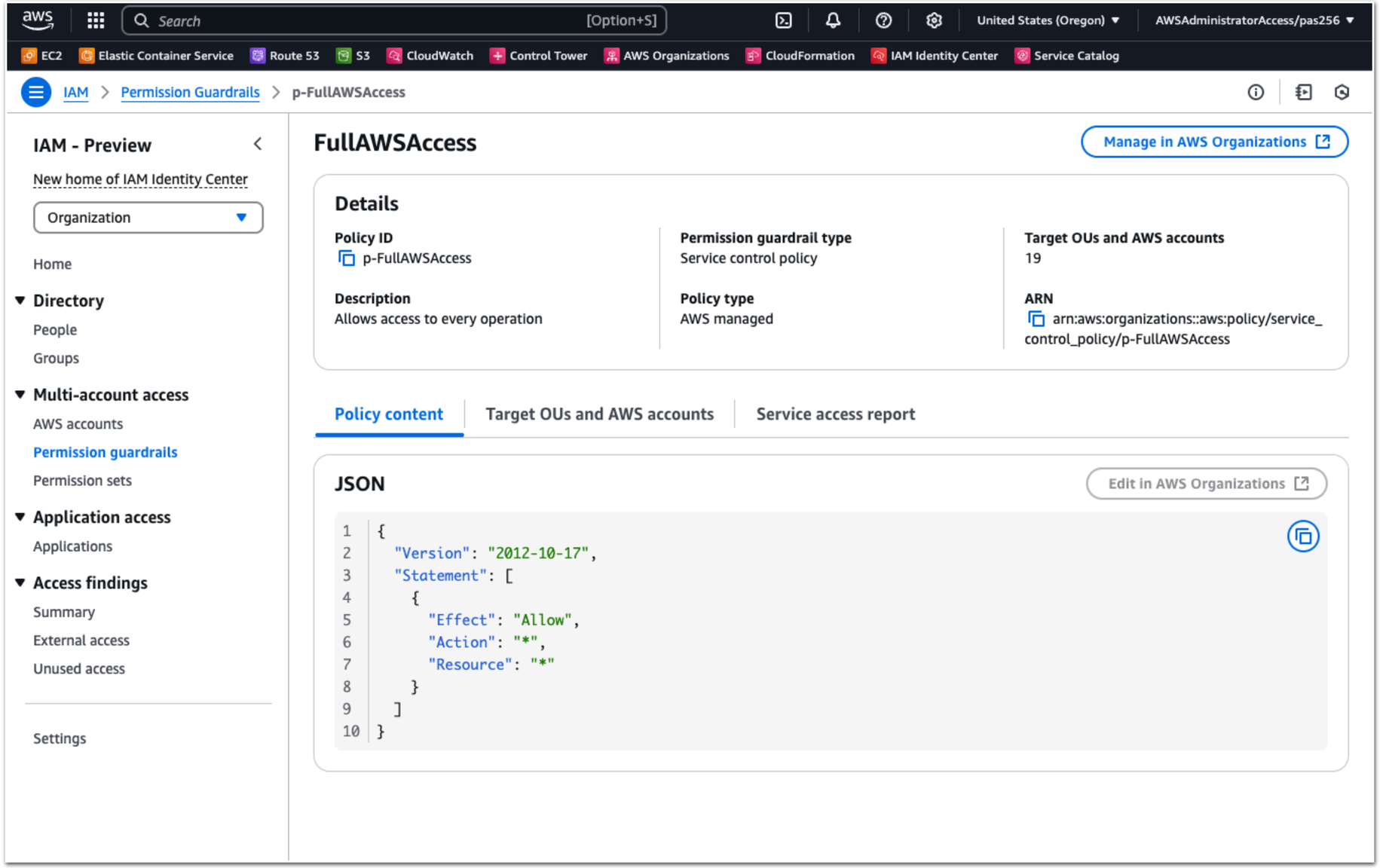


Effective Permissions
= INTERSECTION

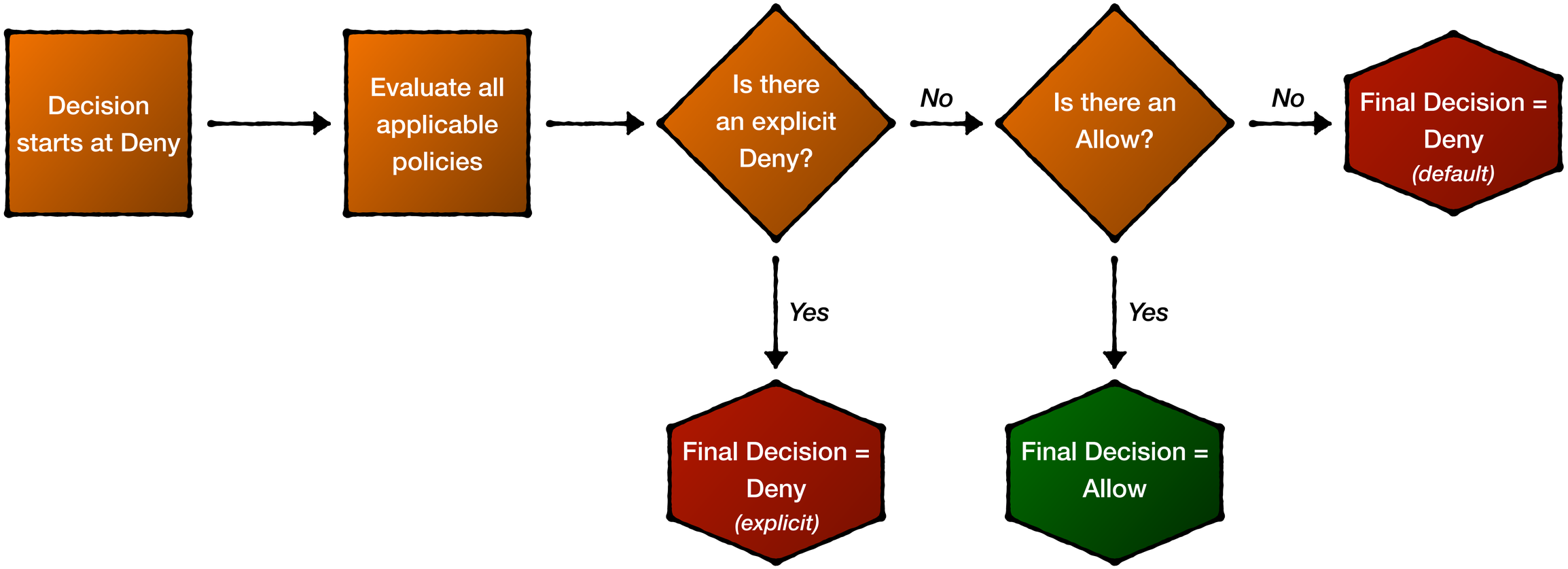
Policy type matrix

Policy type	Act on	Set at	Used to	Example
Identity-based	Principals	Account level	Grant permissions	Grant granular permissions for Users, Groups and Roles.
Resource-based	Resources	Account level	Grant permissions	Grant cross-account access to a Resource. Control access from a Resource.
Service Control (SCP)	Principals	Organization level	Deny permissions	Disable access to services by Users, Groups or Roles.
Resource Control (RCP)	Resources	Organization level	Deny permissions	Disable access to Resources.

SCP default policy



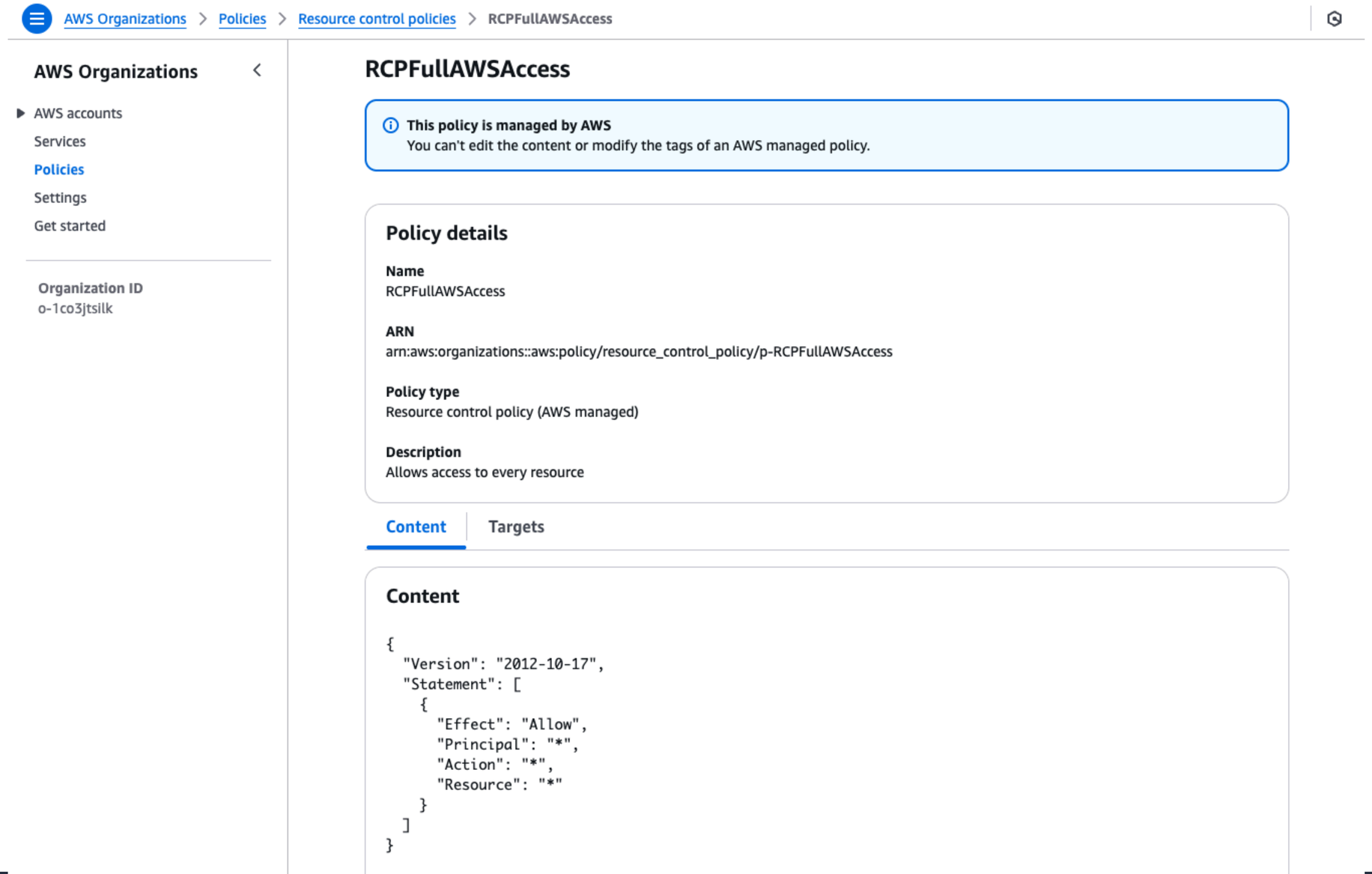
IAM Policy Evaluation



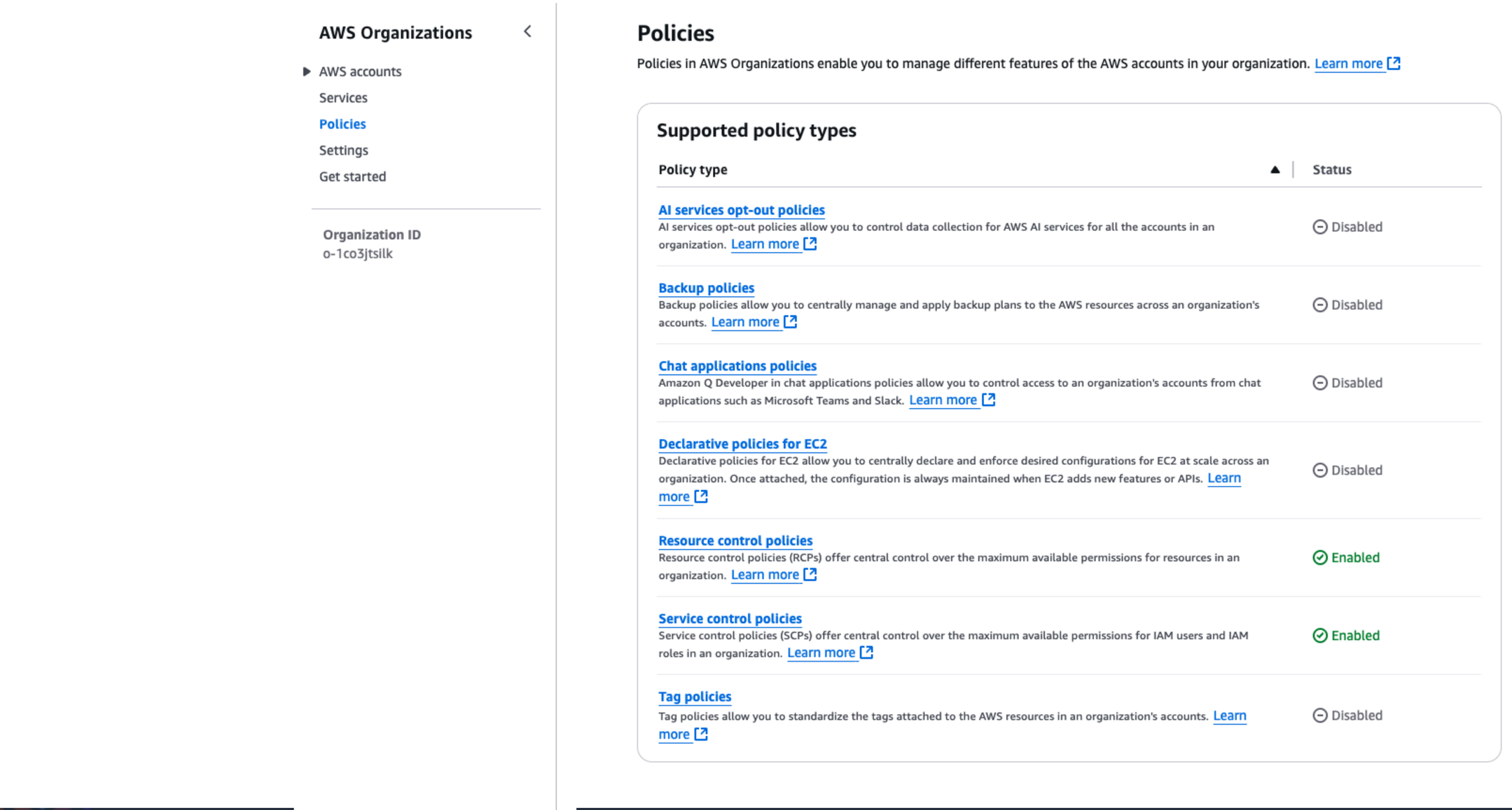
Example SCP policy - Guardrail

```
{
  "Statement":[{
    "Effect": "Deny",
    "Action": [
      "config:DeleteConfigRule",
      "config:DeleteConfigurationAggregator",
      "config:DeleteEvaluationResults",
      "config:PutConfigRule",
      "config:PutConfigurationAggregator"
    ],
    "Resource": "*",
    "Condition": {
      "ArnNotLike": { "aws:PrincipalARN": "arn:aws:iam::*:role/AWSControlTowerExecution" }
    }
  }]
}
```

RCP default policy



Organization Policies



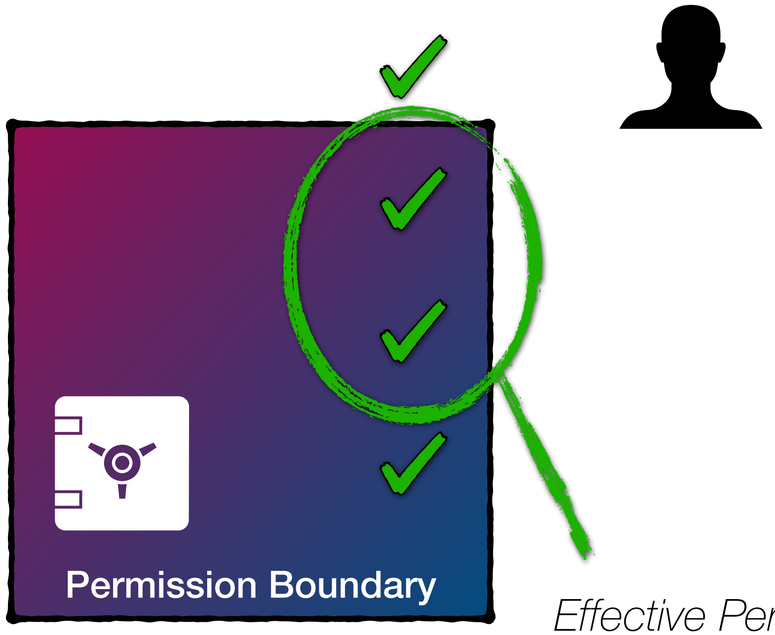
Some additional policy types

Permission Boundaries

Powerful way to limit privilege escalation, but not widely adopted outside of larger enterprises

Scoped-down policies

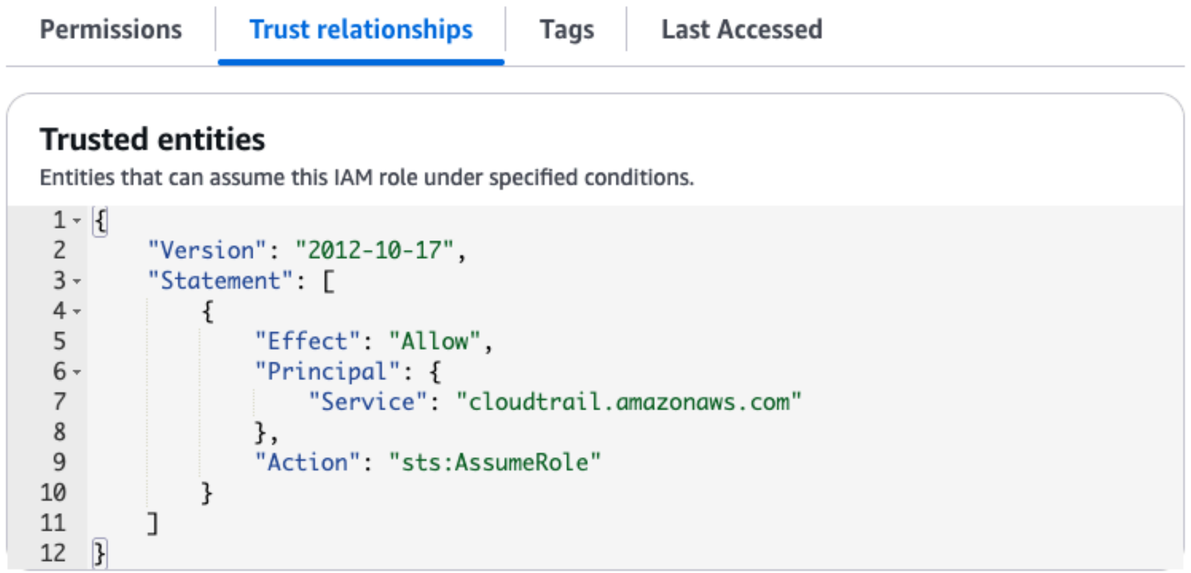
Assume a role (via STS) with a minimum set of permissions



Effective Permissions
= INTERSECTION

IAM Role Trust Policy

- Allows you to define which Principals you trust to assume a role.
- Not to be confused with what the role can do



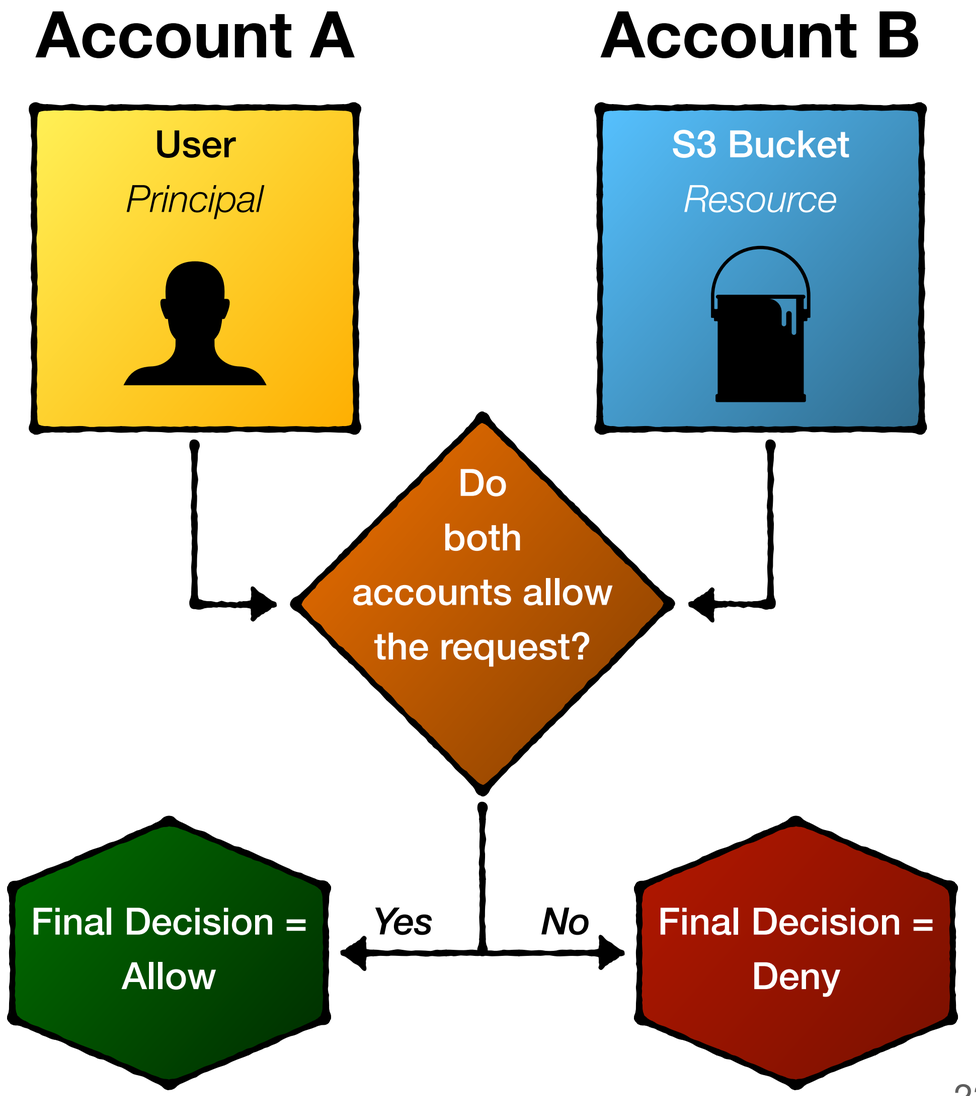
Permission across accounts

Within an AWS account

- Identity *UNION* Resource

Across multiple AWS accounts

- Identity *INTERSECT* Resource
- Need explicit Allows on both sides
- Create a "trust hug"




Pop Quiz

How policies work together *within an account*

Account A

IAM User: pas256

```
{
  "Statement":[{
    "Effect": "Allow",
    "Action": "s3:PutObject"
    "Resource": "*"
  }]
}
```

OR

S3 Bucket: my-bucket

```
{

        no policy defined


}
```

```
aws s3 cp my-file.txt s3://my-bucket/
```

How policies work together *within an account*

Account A

IAM User: pas256

```
{

        no policy defined



}
```

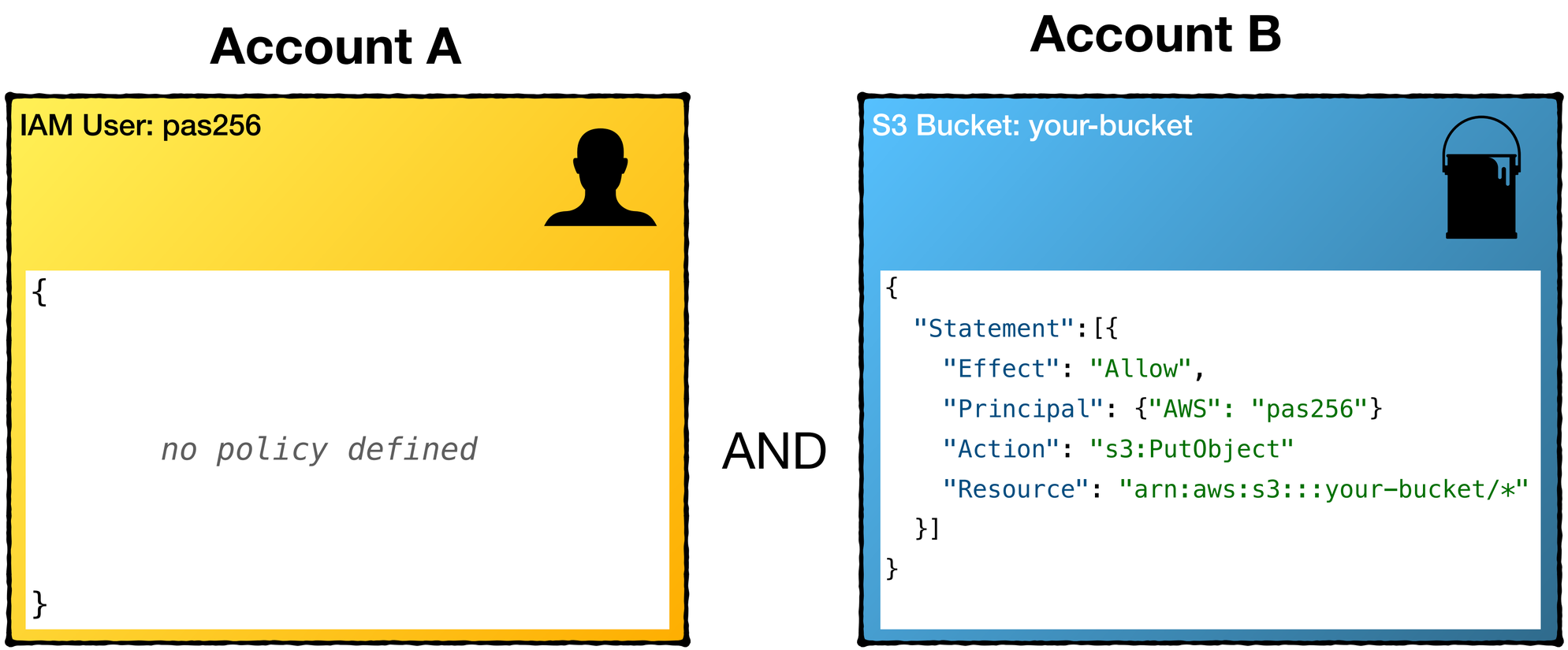
OR

S3 Bucket: my-bucket

```
{
  "Statement":[{
    "Effect": "Allow",
    "Principal": {"AWS": "pas256"}
    "Action": "s3:PutObject"
    "Resource": "arn:aws:s3:::my-bucket/*"
  }]
}
```

```
aws s3 cp my-file.txt s3://my-bucket/
```

How policies work together *across accounts*




```
aws s3 cp my-file.txt s3://your-bucket/
```

How policies work together *across accounts*

Account A

IAM User: pas256

```
{
  "Statement":[{
    "Effect": "Allow",
    "Action": "s3:PutObject"
    "Resource": "*"
  }]
}
```

AND

Account B

S3 Bucket: your-bucket

```
{
  "Statement":[{
    "Effect": "Allow",
    "Principal": {"AWS": "pas256"}
    "Action": "s3:PutObject"
    "Resource": "arn:aws:s3:::your-bucket/*"
  }]
}
```

```
aws s3 cp my-file.txt s3://your-bucket/
```

How policies work together *across accounts*

Account A

SCP

```
{
  "Statement":[{
    "Sid": "FullAWSAccess",
    "Effect": "Allow",
    "Action": "*"
    "Resource": "*"
  }]
}
```

AND

IAM User: pas256

```
{

    no policy defined


}
```

Account B

AND

S3 Bucket: your-bucket

```
{

    no policy defined


}
```

```
aws s3 cp my-file.txt s3://your-bucket/
```

How policies work together *across accounts*

Account A

SCP

```
{
  "Statement":[{
    "Sid": "FullAWSAccess",
    "Effect": "Allow",
    "Action": "*"
    "Resource": "*"
  }]
}
```

AND

IAM User: pas256

```
{

        no policy defined


}
```

AND

Account B

S3 Bucket: your-bucket

```
{
  "Statement":[{
    "Effect": "Allow",
    "Principal": {"AWS": "pas256"}
    "Action": "s3:PutObject"
    "Resource": "arn:aws:s3:::your-bucket/*"
  }]
}
```

```
aws s3 cp my-file.txt s3://your-bucket/
```

How policies work together *across accounts*

Account A

SCP

```
{
  "Statement":[{
    "Sid": "FullAWSAccess",
    "Effect": "Allow",
    "Action": "*"
    "Resource": "*"
  }]
}
```

AND

IAM User: pas256

```
{
  "Statement":[{
    "Effect": "Allow",
    "Action": "s3:PutObject"
    "Resource": "*"
  }]
}
```

AND

Account B

S3 Bucket: your-bucket

```
{
  "Statement":[{
    "Effect": "Allow",
    "Principal": {"AWS": "pas256"}
    "Action": "s3:PutObject"
    "Resource": "arn:aws:s3:::your-bucket/*"
  }]
}
```

```
aws s3 cp my-file.txt s3://your-bucket/
```

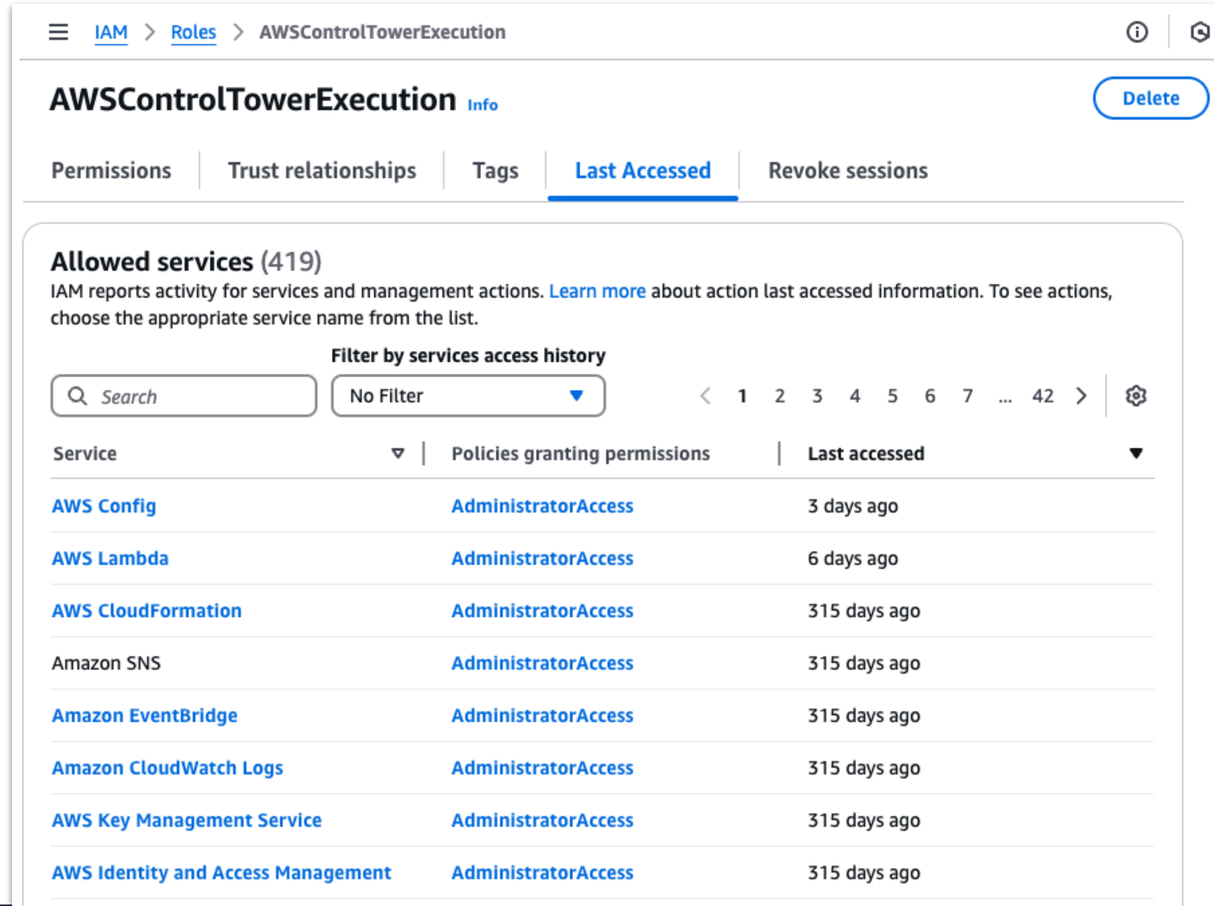
Troubleshooting

Verify Last Accessed

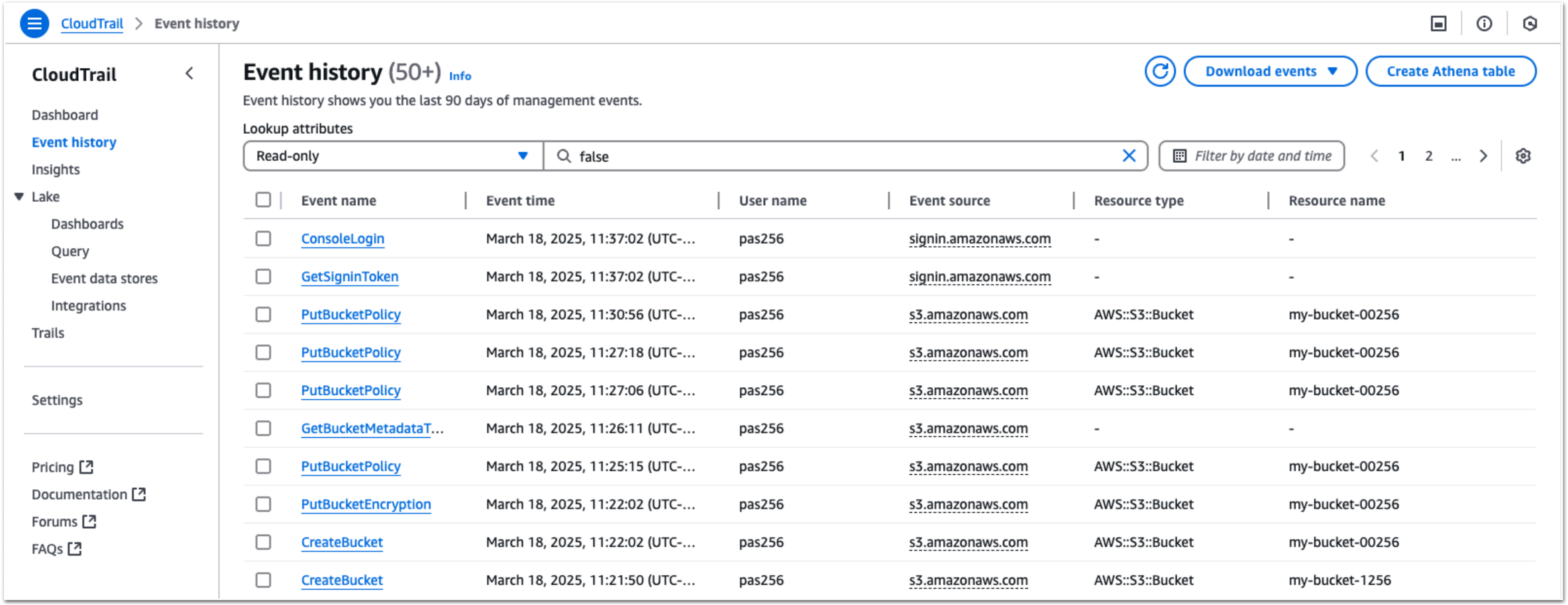
First, make sure the Principal you think you're using is indeed being used!

Documentation

- Activity appears within 4 hours
- Tracks 400 days of history
- Tracks attempts, not just success
- Tracks Users and Roles



Inspect using CloudTrail



Inspect using CloudTrail

PutBucketPolicy Info

Details Info

Event time
March 18, 2025, 11:30:56 (UTC-07:00)

User name
pas256

Event name
PutBucketPolicy

Event source
s3.amazonaws.com

AWS access key
ASIA2UC3D7TR5SK2ZPS6

Source IP address
76.247.189.86

Event ID
d4333450-dd1c-4a12-894d-91c10754267c

Request ID
92P16KM607SKHFJD

AWS region
us-west-2

Error code
MalformedPolicy

Read-only
false

Resources referenced (1) Info

Resource type	Resource name	AWS Config resource timeline
AWS::S3::Bucket	my-bucket-00256	View AWS Config resource timeline

Event record Info

Copy

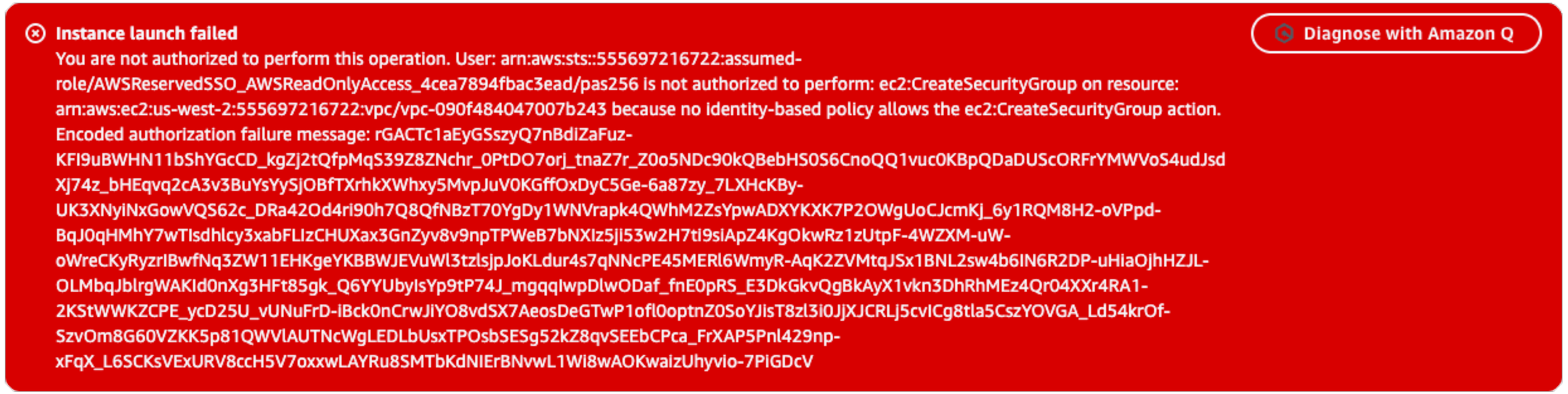
JSON view


```
{
    "eventVersion": "1.11",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROA2UC3D7TR23CMOBIDD:pas256",
        "arn": "arn:aws:sts::730335542499:assumed-role/AWSReservedSSO_AWSAdministratorAccess_aeccfa102a4e673d/pas256",
        "accountId": "730335542499",
```

Inspect using CloudTrail

- View last 90 days in AWS Web Console
- Must be enabled
  - Create one, and **ONLY ONE**, trail (*otherwise $$$!!!*)
- Not 100% coverage by default
  - e.g. No `s3:PutObject` coverage
  - Can enable logging data events to get `s3:PutObject`
    - Careful. This can cost $$$ with high volume projects

Decode the error

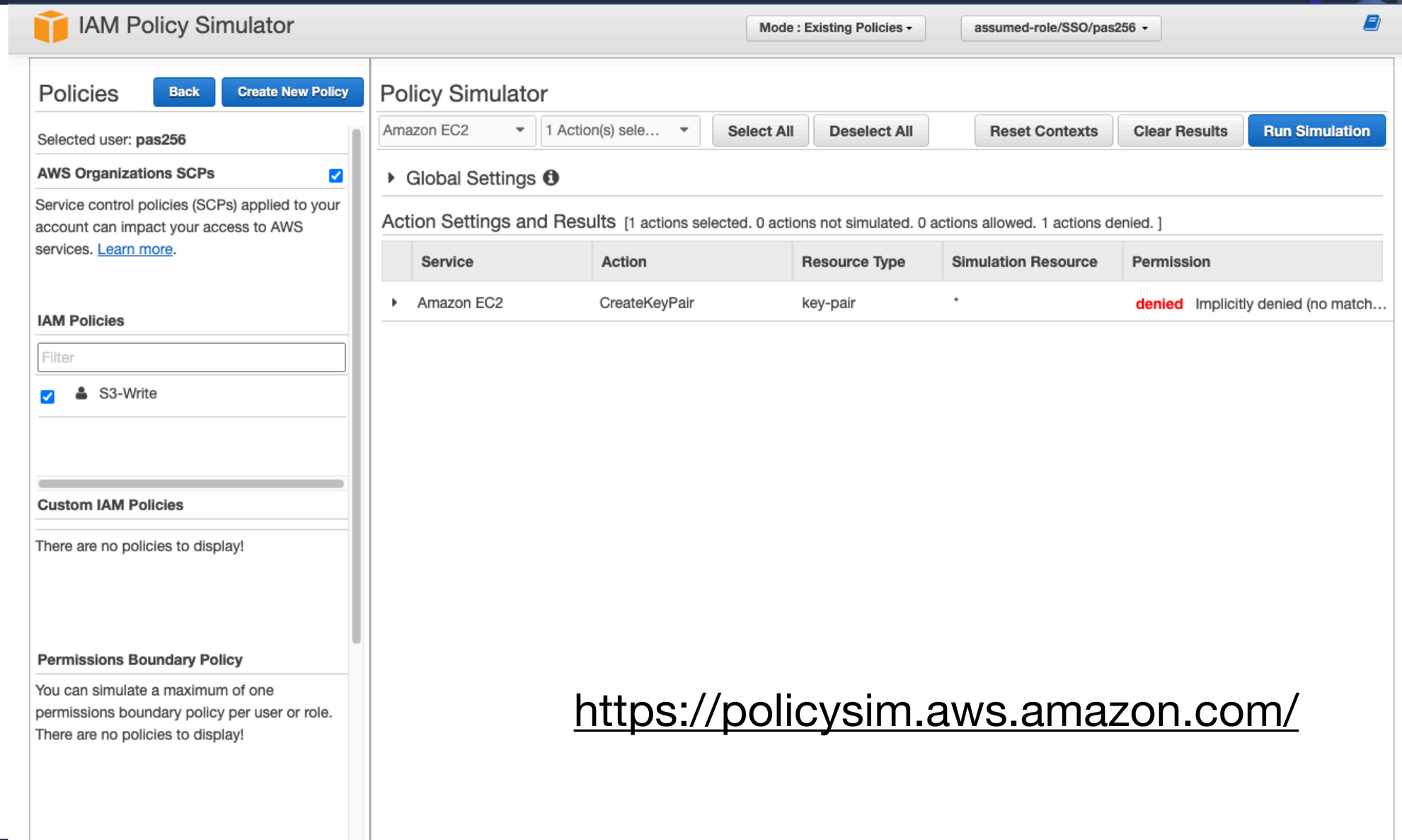



```
aws sts decode-authorization-message
  --encoded-message 'rGACT....GDcV' |
  jq -r '.DecodedMessage' |
  jq
```

Decode the error

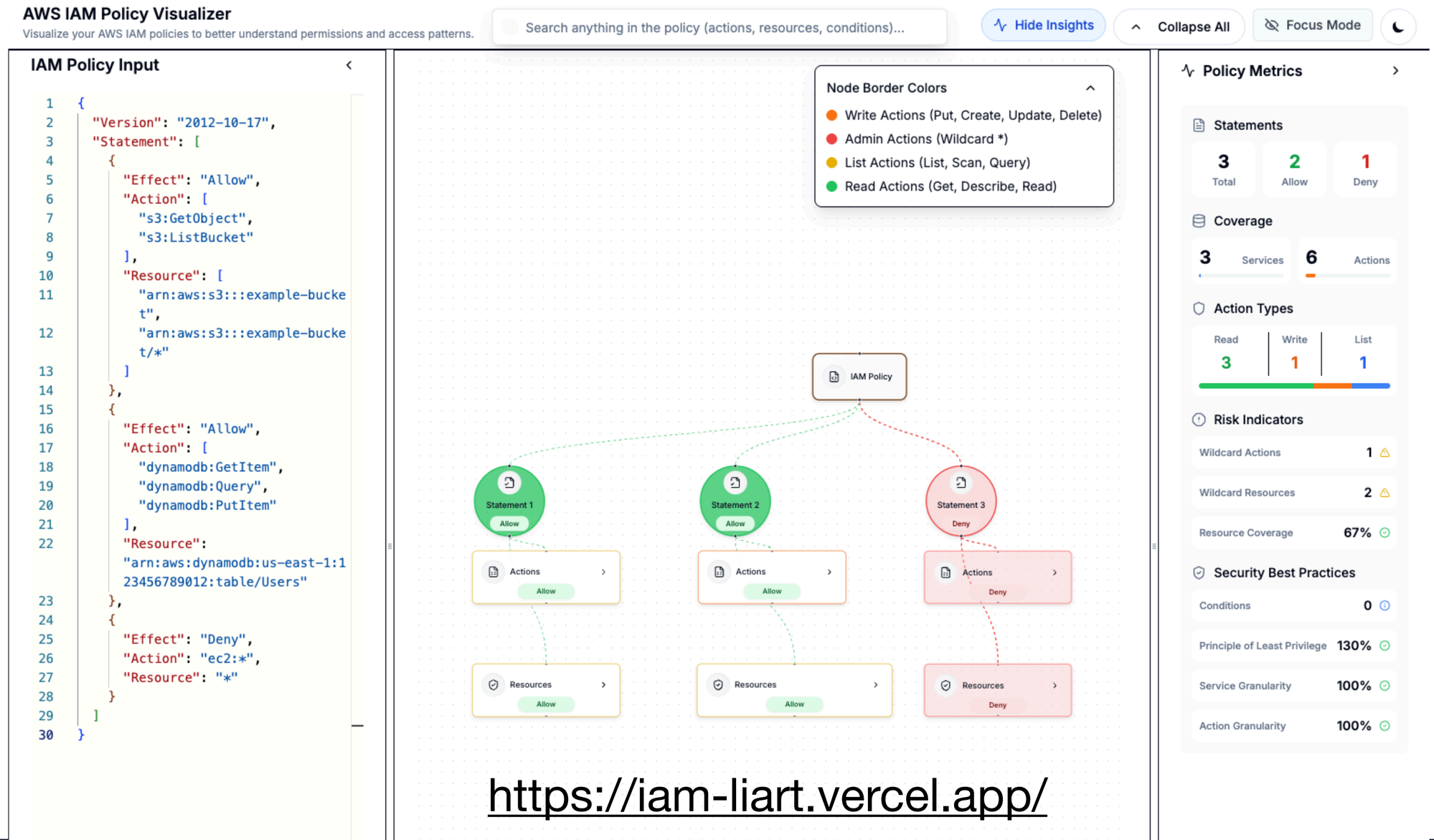

```
{
  "allowed": false,
  "explicitDeny": false,
  "matchedStatements": {
    "items": []
  },
  "failures": {
    "items": []
  },
  "context": {
    "principal": {
      "id": "AROA2UC3D7TR6HATBG7IO:pas256",
      "arn": "arn:aws:sts::730335542499:assumed-role/SSO/pas256"
    },
    "action": "CreateSecurityGroup",
    "resource": "arn:aws:ec2:us-west-2:730335542499:vpc/vpc-079df256ef81d3c40",
    "conditions": {
      "items": [
        {
          "key": "ec2:ResourceTag/ManagedBy",
          "values": {
            "items": [
              {
                "value": "Terraform"
              }
            ]
          }
        },
        {
          "key": "aws:Region",
          "values": {
            "items": [
              {
                "value": "us-west-2"
              }
            ]
          }
        },
        {
          "key": "aws:Resource",
          "values": {
            "items": [
              {
                "value": "vpc/vpc-079df256ef81d3c40"
              }
            ]
          }
        },
        {
          "key": "ec2:VpcID",
          "values": {
            "items": [
              {
                "value": "vpc-079df256ef81d3c40"
              }
            ]
          }
        },
```

IAM Policy Simulator



https://policysim.aws.amazon.com/

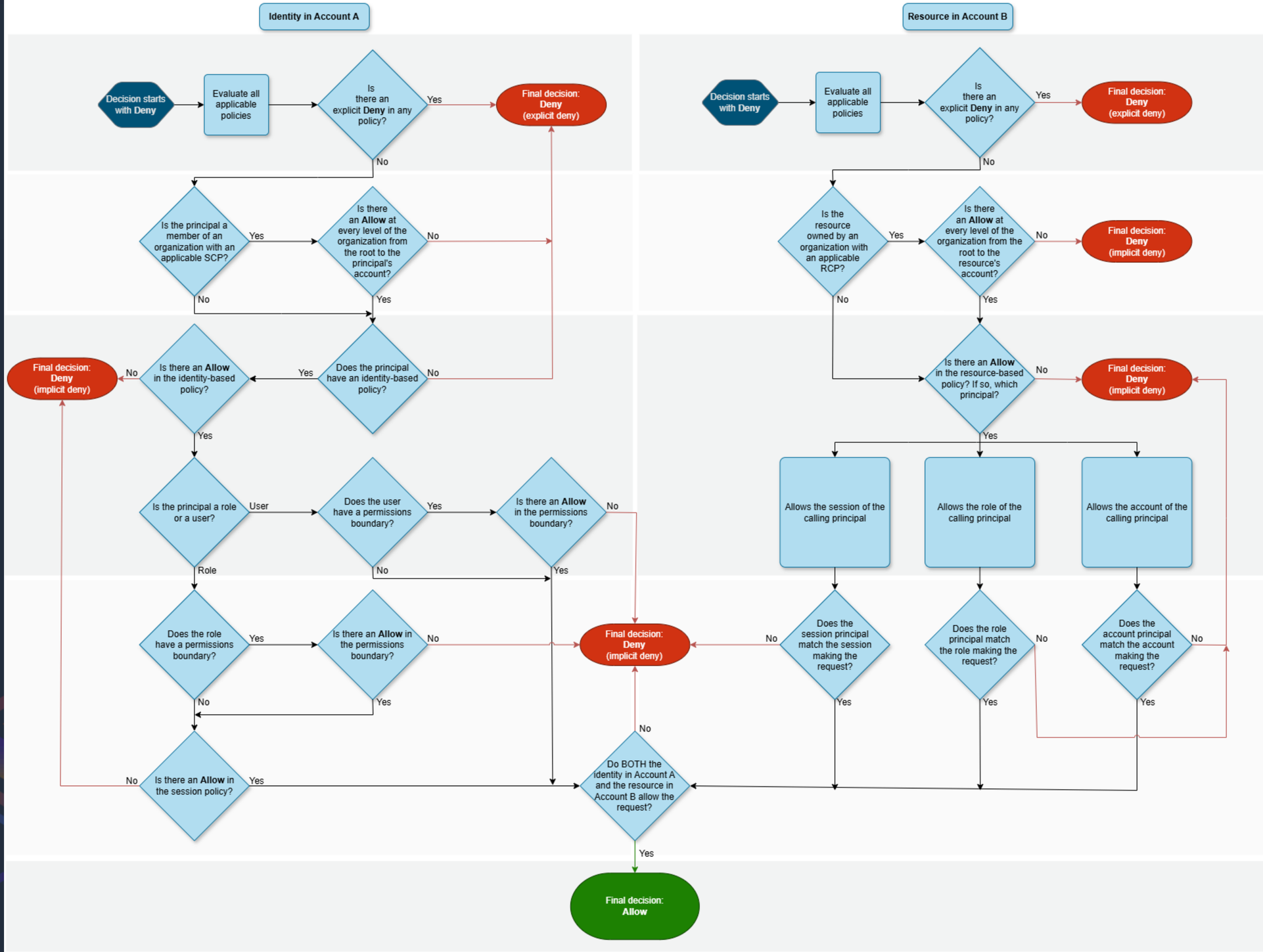
AWS IAM Policy Visualizer



https://iam-liart.vercel.app/

This is the tip of the iceberg

- Experiment
  - Get hands-on
- The IAM docs are actually really good
- Watch Brigid's talk
  - https://www.youtube.com/watch?v=YQsK4MtsELU


Identity in Account A
Resource in Account B
Decision starts with Deny
Evaluate all applicable policies
Is there an explicit Deny in any policy?
Yes
Final decision: Deny (explicit deny)
No
Is the principal a member of an organization with an applicable SCP?
Yes
Is there an Allow at every level of the organization from the root to the principal's account?
No
No
Yes
Final decision: Deny (implicit deny)
No
Is there an Allow in the identity-based policy?
Yes
Does the principal have an identity-based policy?
No
Yes
Is the principal a role or a user?
User
Does the user have a permissions boundary?
Yes
Is there an Allow in the permissions boundary?
No
Role
No
Yes
Does the role have a permissions boundary?
Yes
Is there an Allow in the permissions boundary?
No
No
Yes
Is there an Allow in the session policy?
No
Yes
Final decision: Deny (implicit deny)
Do BOTH the identity in Account A and the resource in Account B allow the request?
No
Yes
Final decision: Allow
Decision starts with Deny
Evaluate all applicable policies
Is there an explicit Deny in any policy?
Yes
Final decision: Deny (explicit deny)
No
Is the resource owned by an organization with an applicable RCP?
Yes
Is there an Allow at every level of the organization from the root to the resource's account?
No
Final decision: Deny (implicit deny)
No
Yes
Is there an Allow in the resource-based policy? If so, which principal?
No
Final decision: Deny (implicit deny)
Yes
Allows the session of the calling principal
Allows the role of the calling principal
Allows the account of the calling principal
Does the session principal match the session making the request?
No
Yes
Does the role principal match the role making the request?
No
Yes
Does the account principal match the account making the request?
No
Yes



Thank you