



# SUMMIT

ANAHEIM | AUGUST 18, 2022

SEC301

# Data perimeter workshop

Swara Gandhi

Solutions Architect, AWS Identity  
Amazon Web Services

Liam Wadman

Solutions Architect, AWS Identity  
Amazon Web Services



# Agenda

- Data perimeters on AWS
- Hands-on lab

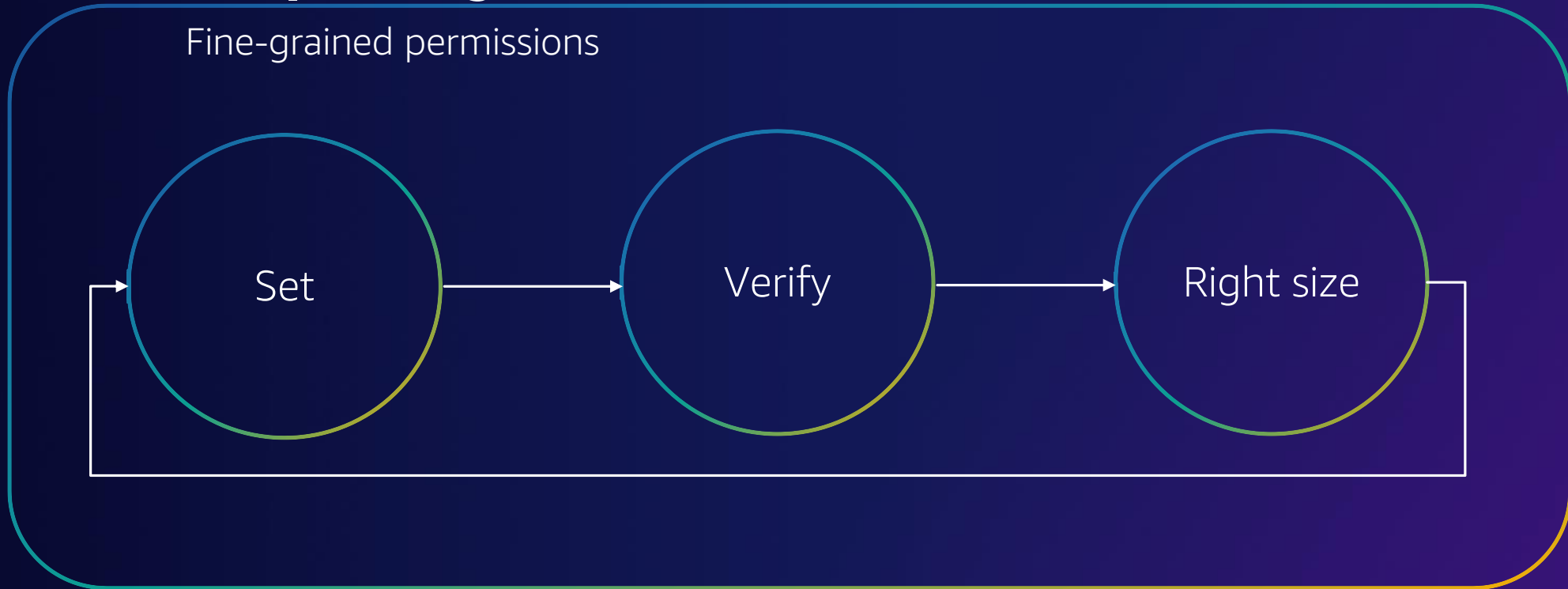
# Access management

## Data perimeter

Coarse-grained controls

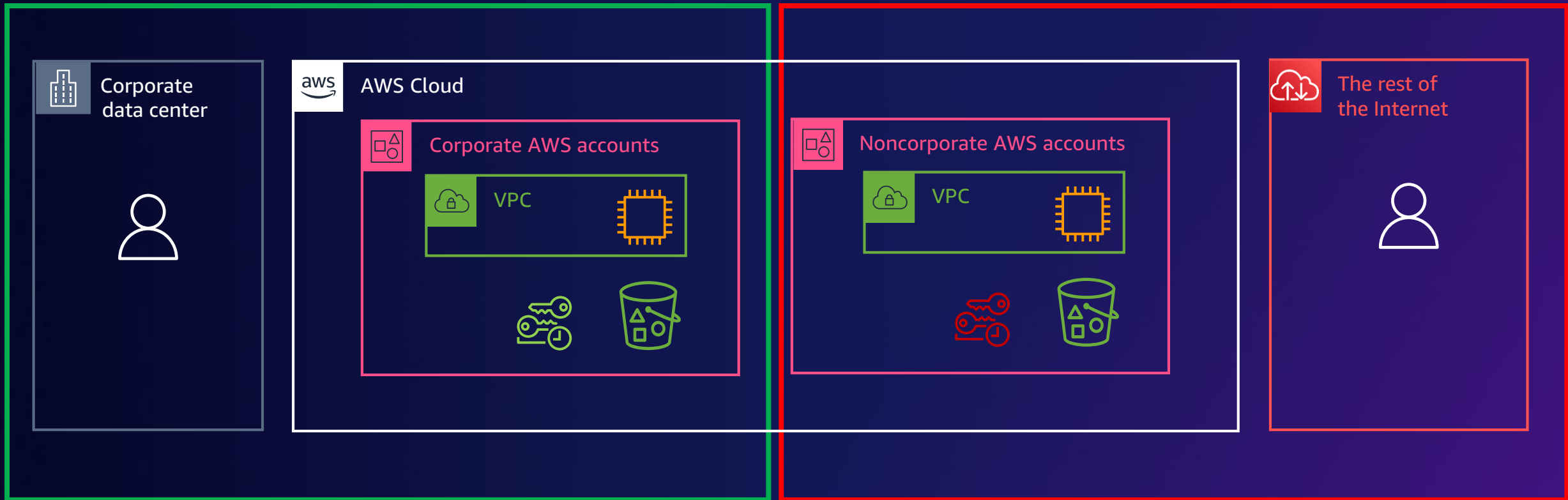
### Least privilege

Fine-grained permissions



# What is a data perimeter?

A set of preventive guardrails in your AWS environment that help ensure only your **trusted identities** are accessing **trusted resources** from **expected networks**



# What is a data perimeter?



## Trusted identities

Principals within  
your AWS accounts  
or AWS services  
acting on your behalf



## Trusted resources

Resources owned by  
your AWS accounts  
or by AWS services  
acting on your behalf



## Expected networks

Your on-premises data centers  
and virtual private clouds (VPCs)  
or networks of AWS services  
acting on your behalf

# Data perimeter controls

| Perimeter | Intent/control objective                                       |
|-----------|----------------------------------------------------------------|
| Identity  | Only trusted identities can access my resources                |
|           | Only trusted identities are allowed from my network            |
| Resource  | My identities can access only trusted resources                |
|           | Only trusted resources can be accessed from my network         |
| Network   | My identities can access resources only from expected networks |
|           | My resources can only be accessed from expected networks       |

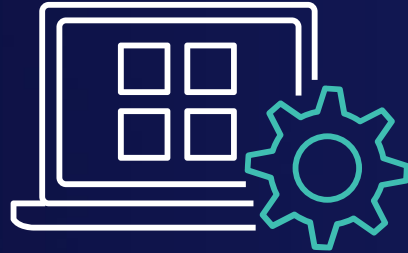
# Benefits



## Build corporate boundary



Restrict access to sensitive data based on expected access patterns



## Provide developer freedom



Allow developers to innovate and move fast



## Meet compliance requirements



Implement guardrails to meet compliance requirements at scale



## Defense in depth



Central control over security invariants

# Establishing a data perimeter

# Primary tools for your data perimeter

1

## Service control policies (SCPs)

Your identities can access only trusted resources from expected networks

2

## VPC endpoint policies

Your network can be used only by trusted identities to access only trusted resources

3

## Resource-based policies

Your resources can be accessed only by trusted identities and from expected networks

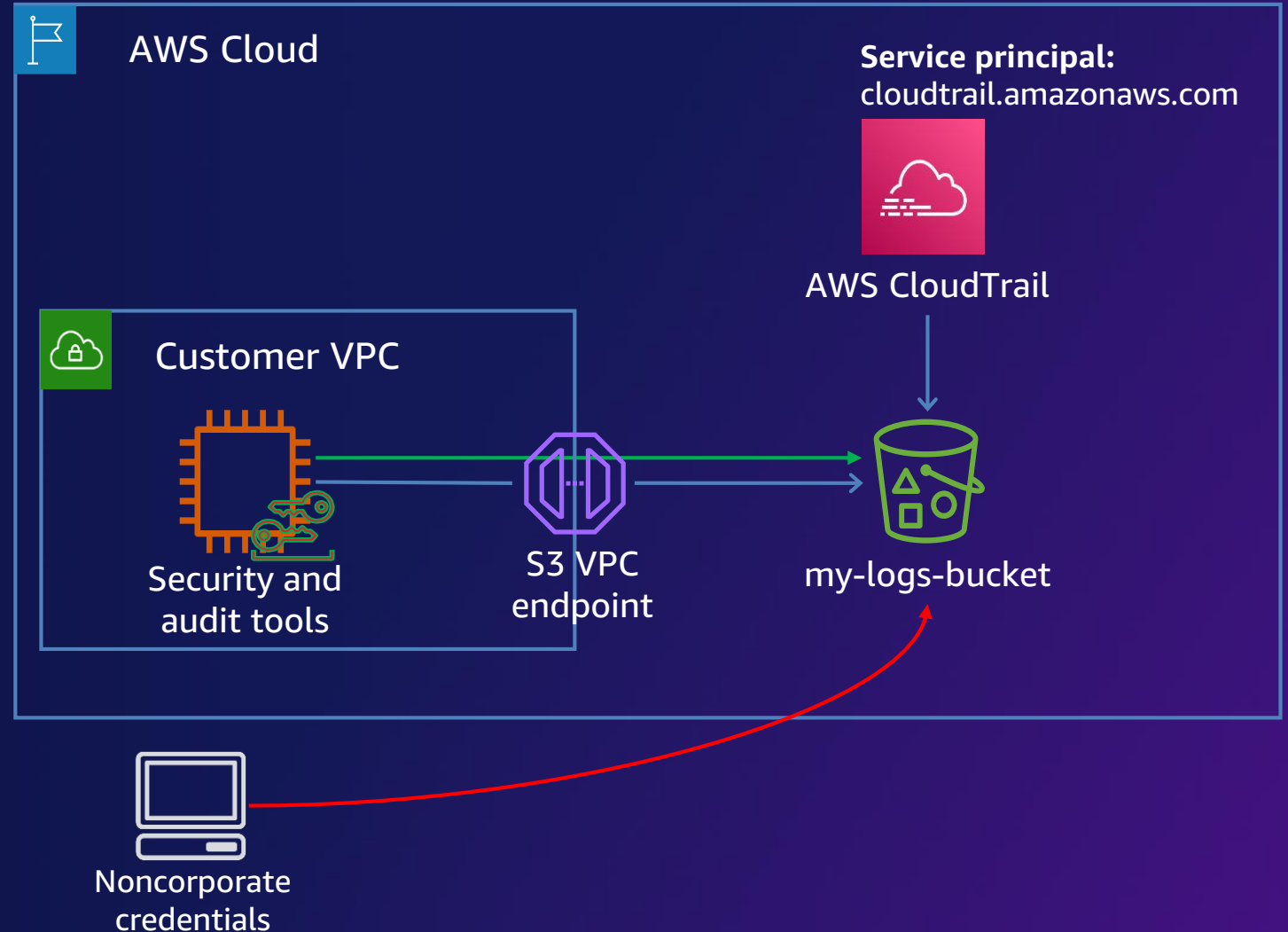
# Data perimeter controls

| Perimeter | Intent/control objective                                       | Applied on | Using                 |
|-----------|----------------------------------------------------------------|------------|-----------------------|
| Identity  | Only trusted identities can access my resources                | Resources  | Resource-based policy |
|           | Only trusted identities are allowed from my network            | Network    | VPC endpoint policy   |
| Resource  | My identities can access only trusted resources                | Identities | SCP                   |
|           | Only trusted resources can be accessed from my network         | Network    | VPC endpoint policy   |
| Network   | My identities can access resources only from expected networks | Identities | SCP                   |
|           | My resources can only be accessed from expected networks       | Resources  | Resource-based policy |

# Identity perimeter on resources

ONLY TRUSTED IDENTITIES CAN ACCESS MY RESOURCES

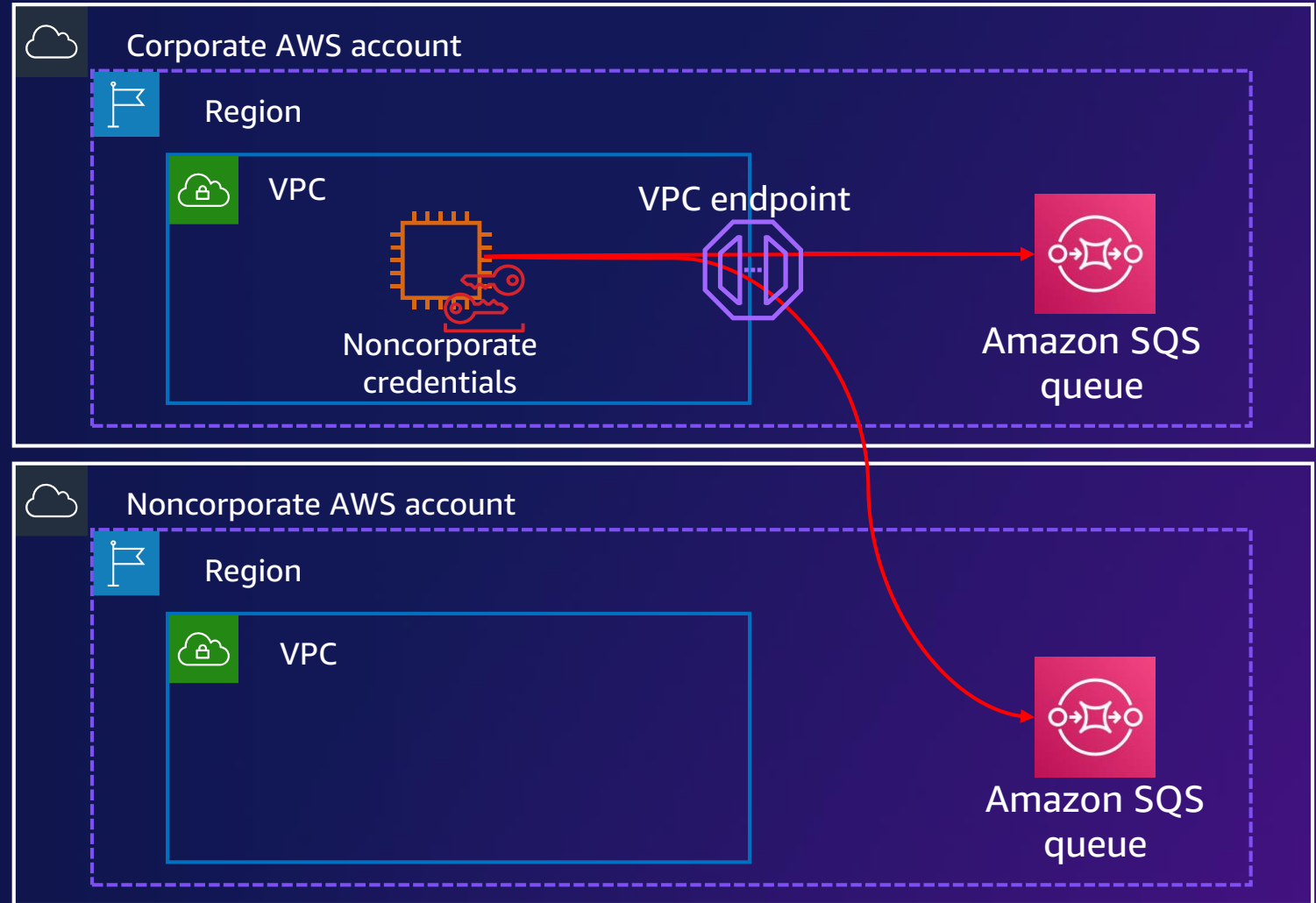
```
.....  
{  
  "Effect": "Deny",  
  "Principal": "*",  
  "Action": [  
    "s3:ListBucket",  
    "s3:PutObject*",  
    "s3:GetObject*"  
  ],  
  "Resource": [  
    "arn:aws:s3:::<my-data-bucket>",  
    "arn:aws:s3:::<my-data-bucket>/*"  
  ],  
  "Condition": {  
    "StringNotEqualsIfExists": {  
      "aws:PrincipalOrgID": "<o-xxxxxxx>"  
    },  
    "BoolIfExists": {  
      "aws:PrincipalIsAWSservice": "false"  
    }  
  }  
}  
.....
```



# Identity perimeter on network

ONLY TRUSTED IDENTITIES ARE ALLOWED FROM MY NETWORK

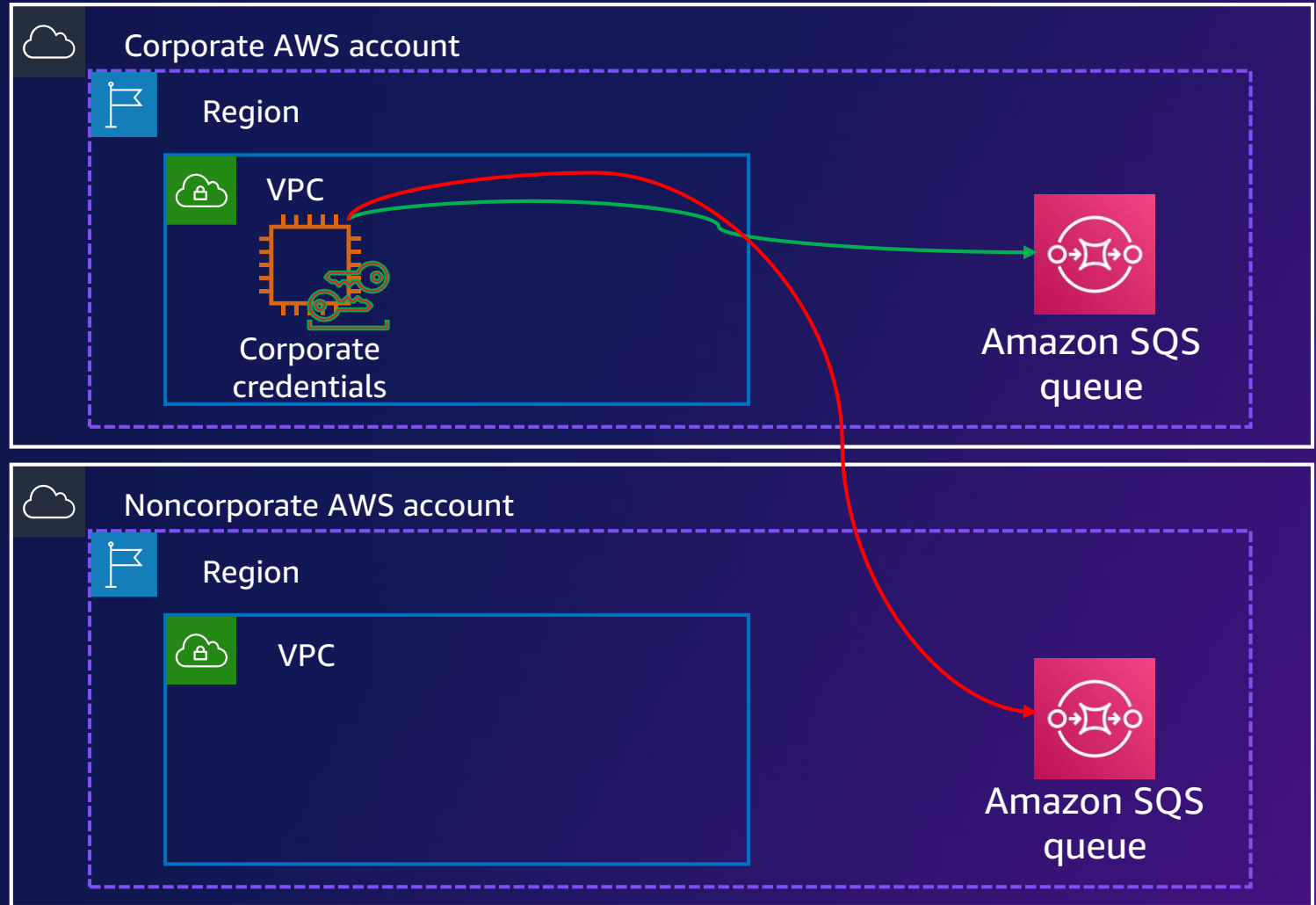
```
{
  "Statement": [
    {
      "Action": "*",
      "Effect": "Allow",
      "Resource": "*",
      "Principal": {
        "AWS": "*"
      },
      "Condition": {
        "StringEquals": {
          "aws:PrincipalOrgID":
            "o-xxxxxxxx"
        }
      }
    }
  ]
}
```



# Resource perimeter on identity

MY IDENTITIES CAN ACCESS ONLY TRUSTED RESOURCES

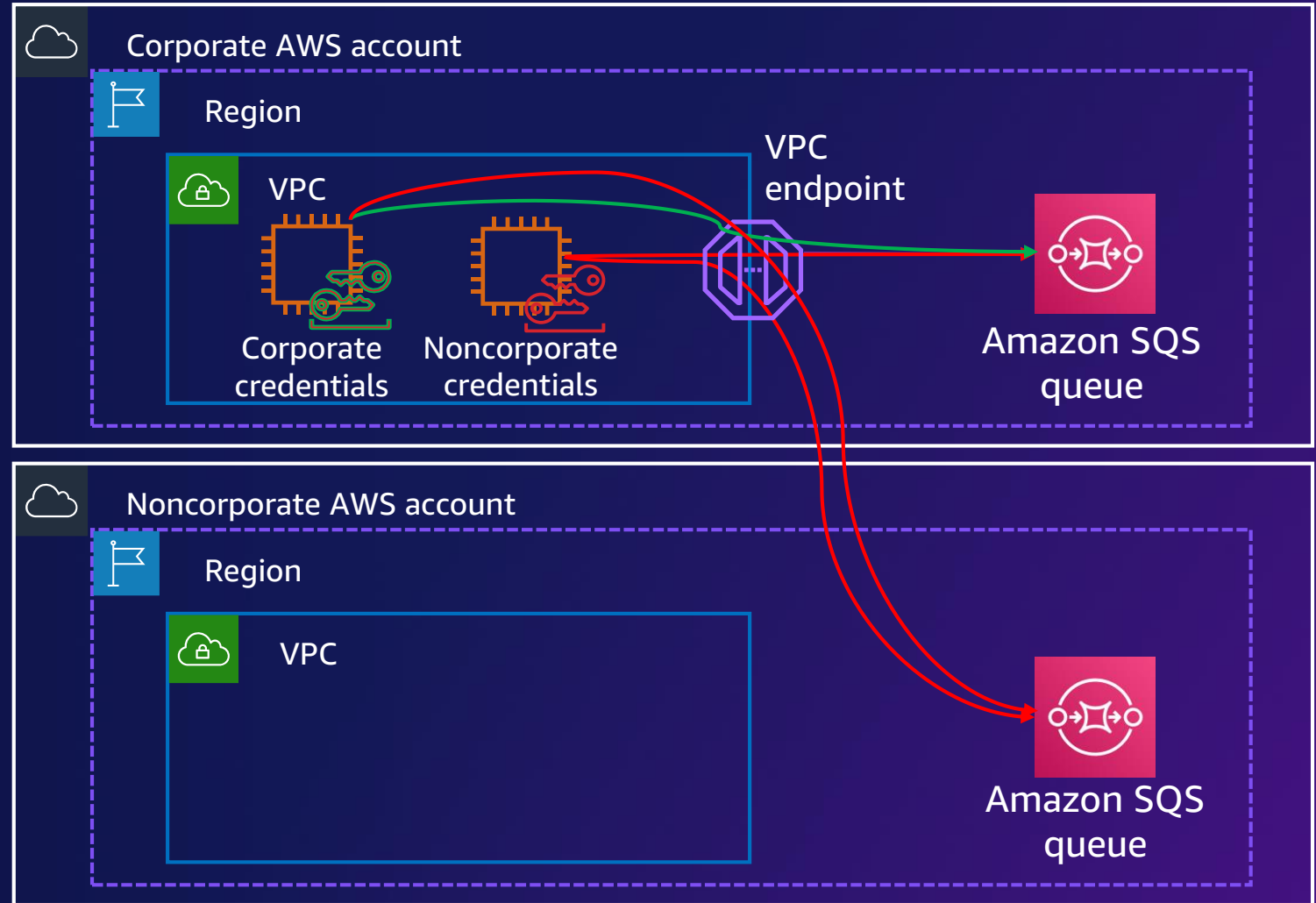
```
.....  
{  
  "Effect": "Deny",  
  "Action": [  
    "sqs:SendMessage"  
  ],  
  "Resource": "*",  
  "Condition": {  
    "StringNotEqualsIfExists ": {  
      "aws:ResourceOrgID ": "o-  
xxxxxxxxxxxx"  
    }  
  }  
}  
.....
```



# Resource perimeter on network

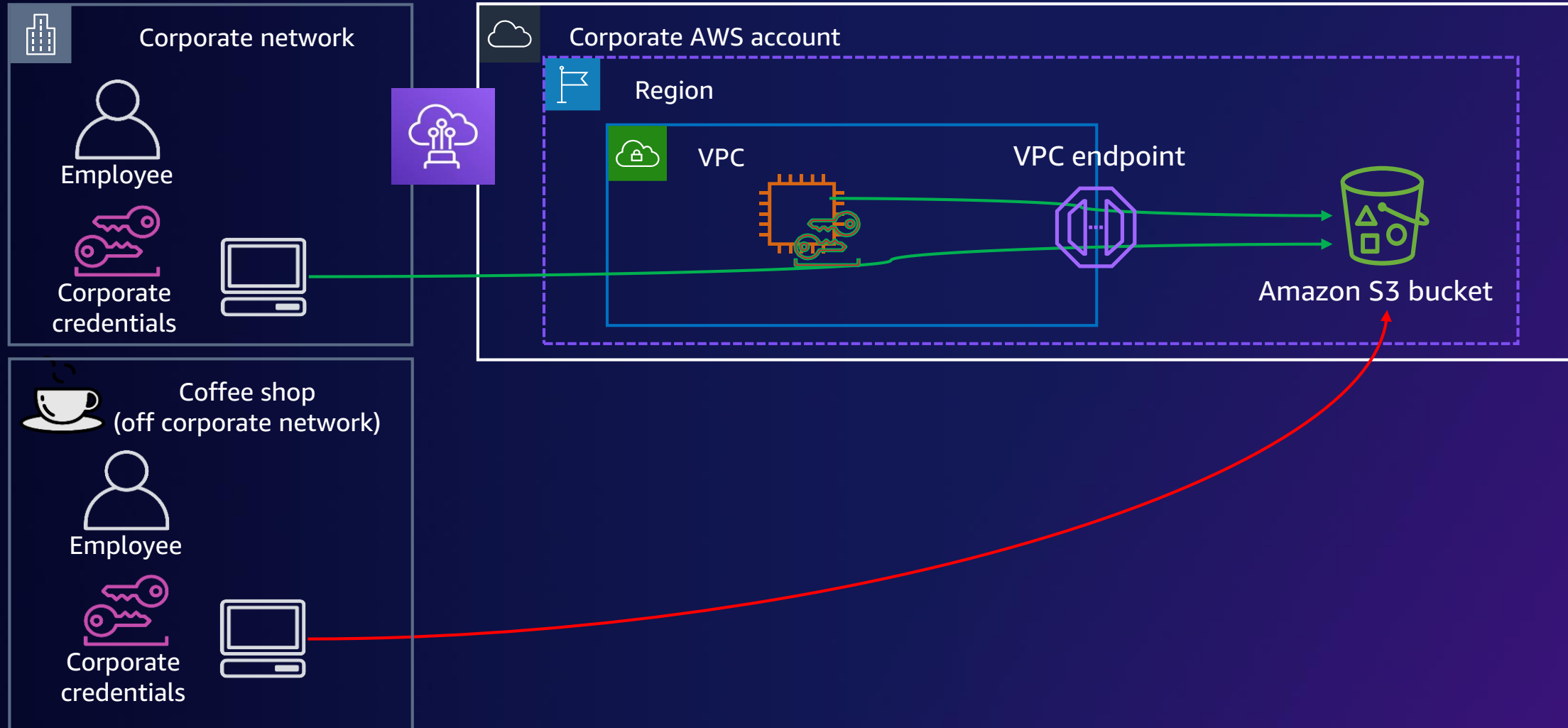
ONLY TRUSTED RESOURCES CAN BE ACCESSED FROM MY NETWORK

```
{
  "Statement": [
    {
      "Action": "*",
      "Effect": "Allow",
      "Resource": "*",
      "Principal": {
        "AWS": "*"
      },
      "Condition": {
        "StringEquals": {
          "aws:PrincipalOrgID": "o-xxxxxxxxxx",
          "aws:ResourceOrgID": "o-xxxxxxxxxx"
        }
      }
    }
  ]
}
```



# Network perimeter on resource

MY RESOURCES CAN ONLY BE ACCESSED FROM EXPECTED NETWORKS



# Network perimeter on resource

MY RESOURCES CAN ONLY BE ACCESSED FROM EXPECTED NETWORKS

```
.....
{
  "Effect": "Deny",
  "Principal": "*",
  "Action": "s3:*",
  "Resource": [
    "arn:aws:s3:::my-data-bucket",
    "arn:aws:s3:::my-data-bucket/*"
  ],
  "Condition": {
    "NotIpAddressIfExists": {
      "aws:SourceIp": "<cidr>"
    },
    "StringNotEqualsIfExists": {
      "aws:SourceVpc": "<vpc-xxxxxxxx>"
    },
    "BoolIfExists": {
      "aws:PrincipalIsAWSService": "false",
      "aws:ViaAWSService": "false"
    },
    "ArnNotLikeIfExists": {
      "aws:PrincipalArn": "arn:aws:iam::<AccountNumber>:role/aws-service-role/*"
    }
  }
}
.....
```

# Data perimeter controls

| Perimeter | Intent/control objective                                       | Applied on | Using                 | Primary AWS Identity and Access Management (IAM) feature                                   |
|-----------|----------------------------------------------------------------|------------|-----------------------|--------------------------------------------------------------------------------------------|
| Identity  | Only trusted identities can access my resources                | Resources  | Resource-based policy | aws:PrincipalOrgID<br>aws:PrincipalIsAWSService                                            |
|           | Only trusted identities are allowed from my network            | Network    | VPC endpoint policy   | aws:PrincipalOrgID<br>aws:PrincipalIsAWSService                                            |
| Resource  | My identities can access only trusted resources                | Identities | SCP                   | aws:ResourceOrgID                                                                          |
|           | Only trusted resources can be accessed from my network         | Network    | VPC endpoint policy   | aws:ResourceOrgID                                                                          |
| Network   | My identities can access resources only from expected networks | Identities | SCP                   | aws:SourceIp<br>aws:SourceVpc/SourceVpce<br>aws:ViaAWSService                              |
|           | My resources can only be accessed from expected networks       | Resources  | Resource-based policy | aws:SourceIp<br>aws:SourceVpc/SourceVpce<br>aws:ViaAWSService<br>aws:PrincipalIsAWSService |

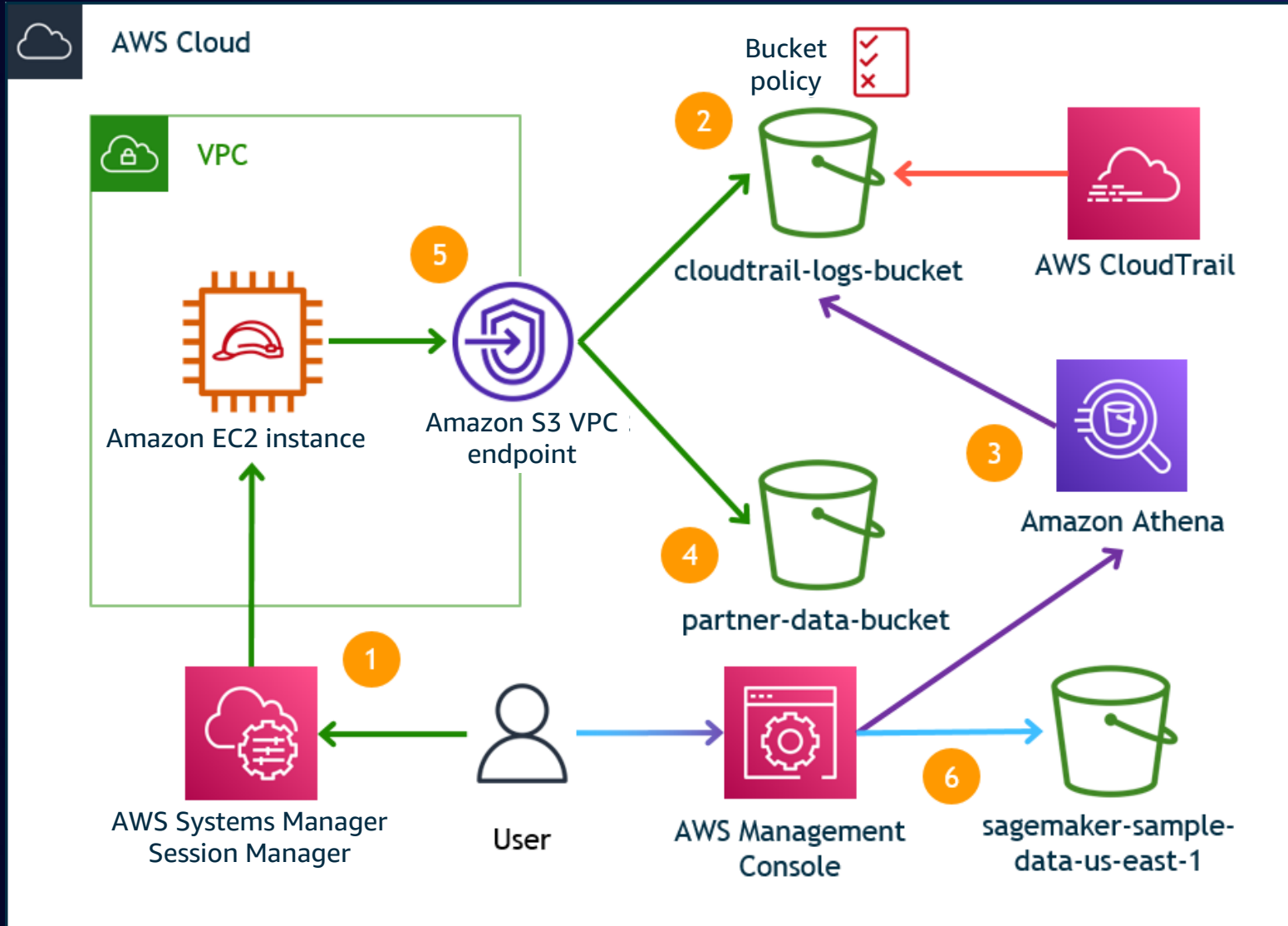
# Hands-on lab



# Lab agenda

- Lab 0: Test environment
- Lab 1: Identity controls for resources
- Lab 2: Network controls for resources
- Lab 3: Identity controls for network
- Lab 4: Resource controls for network
- Lab 5: Resource controls for identities

# Workshop setup



1. You will use AWS Systems Manager Session Manager to access the Amazon EC2 instance in the VPC. The VPC does not have internet connectivity and will connect to all resources through VPC endpoints.
2. In Lab 1, you will access the CloudTrail bucket contents and create a bucket policy that only allows specific principals to access the content; then in Lab 2, you will update that bucket policy to restrict what networks can be used to access the resource.
3. In Lab 2, you will use the AWS Management Console to access Amazon Athena and run a query on the data in the CloudTrail bucket.

The diagram illustrates a multi-account AWS architecture for data sharing and security, featuring the following components and interactions:

- AWS Cloud:** The overall environment containing the VPC and other services.
- VPC (Virtual Private Cloud):** A green-bordered box containing:
  - Amazon EC2 instance:** Represented by a server icon with a red hard hat.
  - Amazon S3 VPC endpoint:** Represented by a purple shield icon with a white arrow.
- External Services and Buckets:**
  - AWS CloudTrail:** A pink square icon with a cloud and lines.
  - cloudtrail-logs-bucket:** A green bucket icon.
  - partner-data-bucket:** A green bucket icon.
  - Amazon Athena:** A purple square icon with a magnifying glass over a document.
  - sagemaker-sample-data-us-east-1:** A green bucket icon.
  - AWS Management Console:** A pink square icon with a gear and a document.
  - AWS Systems Manager Session Manager:** A pink square icon with a cloud and a gear.
- User:** Represented by a person icon.
- Interactions (Numbered 1-6):**
  - 1:** User connects to the AWS Systems Manager Session Manager.
  - 2:** AWS CloudTrail sends data to the cloudtrail-logs-bucket.
  - 3:** Amazon Athena queries data from the partner-data-bucket.
  - 4:** The Amazon S3 VPC endpoint sends data to the partner-data-bucket.
  - 5:** The Amazon EC2 instance sends data to the Amazon S3 VPC endpoint.
  - 6:** The AWS Management Console sends data to the sagemaker-sample-data-us-east-1 bucket.
- Additional Details:**
  - A **Bucket policy** is shown with a red box containing three checkmarks and one 'X', indicating a policy that allows access from the VPC endpoint.
  - The **cloudtrail-logs-bucket** is also connected to the **Amazon S3 VPC endpoint**.

4. In Lab 3, you will access your partner's Amazon S3 data bucket, demonstrating an access pattern you don't want to occur.
5. Continuing in Lab 3, in order to prevent the cross-account access, you will update the Amazon S3 VPC endpoint policy to only allow principals from a specific account to access Amazon S3 resources from the network; then in Lab 4, you will update that policy again to restrict what resources can be accessed from the network.
6. In Lab 5, you will use the AWS Management Console to create an IAM policy that restricts the Amazon S3 resources that can be accessed by your IAM role and test that policy on an AWS-owned public Amazon S3 bucket.

Instructions: <https://data-perimeter.workshop.aws/>

AWS account login: <https://dashboard.eventengine.run>

Event hash: [71ce-133f9c7354-1e](#)

# Learn in-demand AWS Cloud skills



## AWS Skill Builder

Access **500+ free** digital courses and Learning Plans

Explore resources with a variety of skill levels and **16+** languages to meet your learning needs

Deepen your skills with digital learning on demand



Train now



## AWS Certifications

Earn an industry-recognized credential

Receive Foundational, Associate, Professional, and Specialty certifications

Join the **AWS Certified community** and get exclusive benefits



Access **new** exam guides

# Thank you!

