AWS IAM vs Identity Center vs. Cognito

Identity and Access Management on AWS

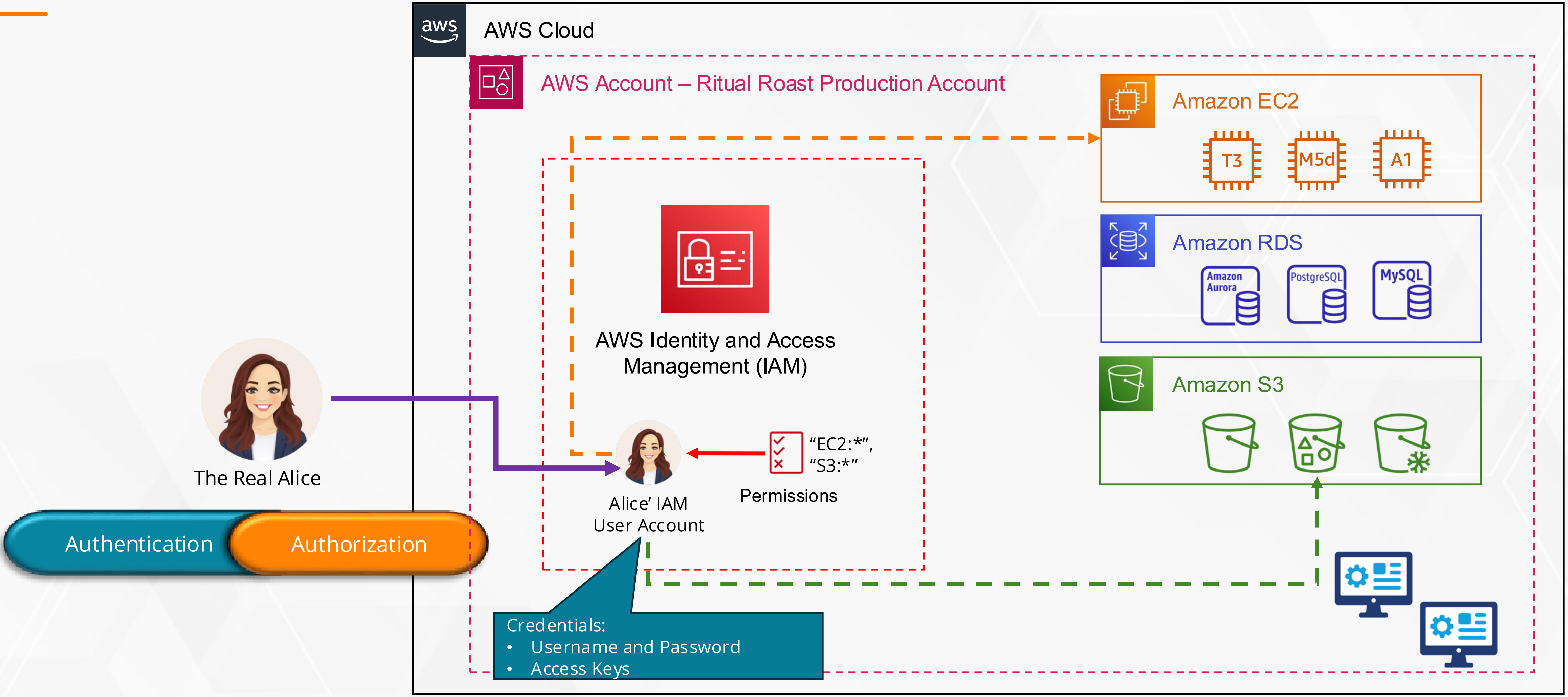
Amazon Cognito

AWS Identity and
Access Management

AWS Identity Center

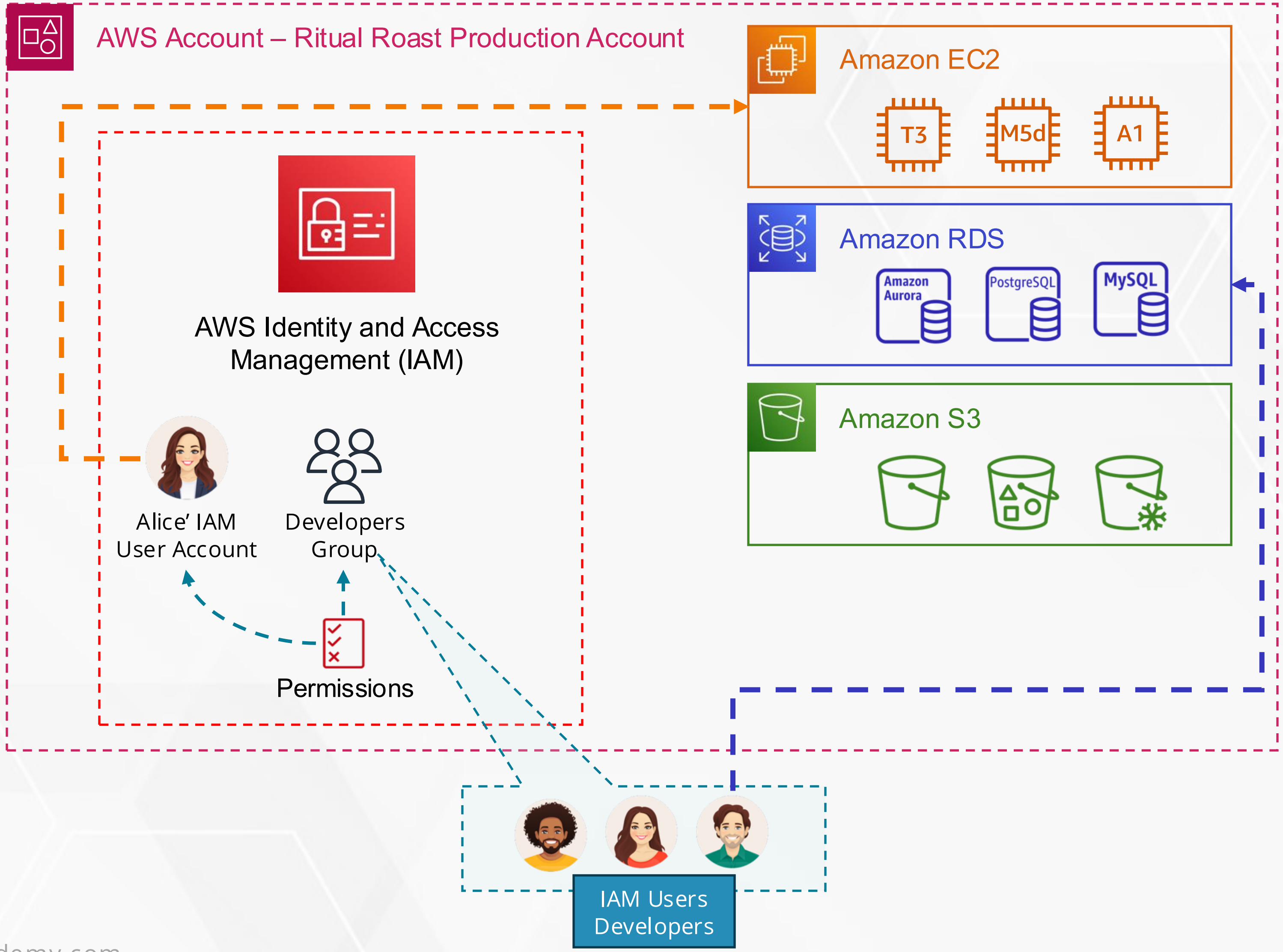
AWS Identity Tools

What is the AWS IAM service



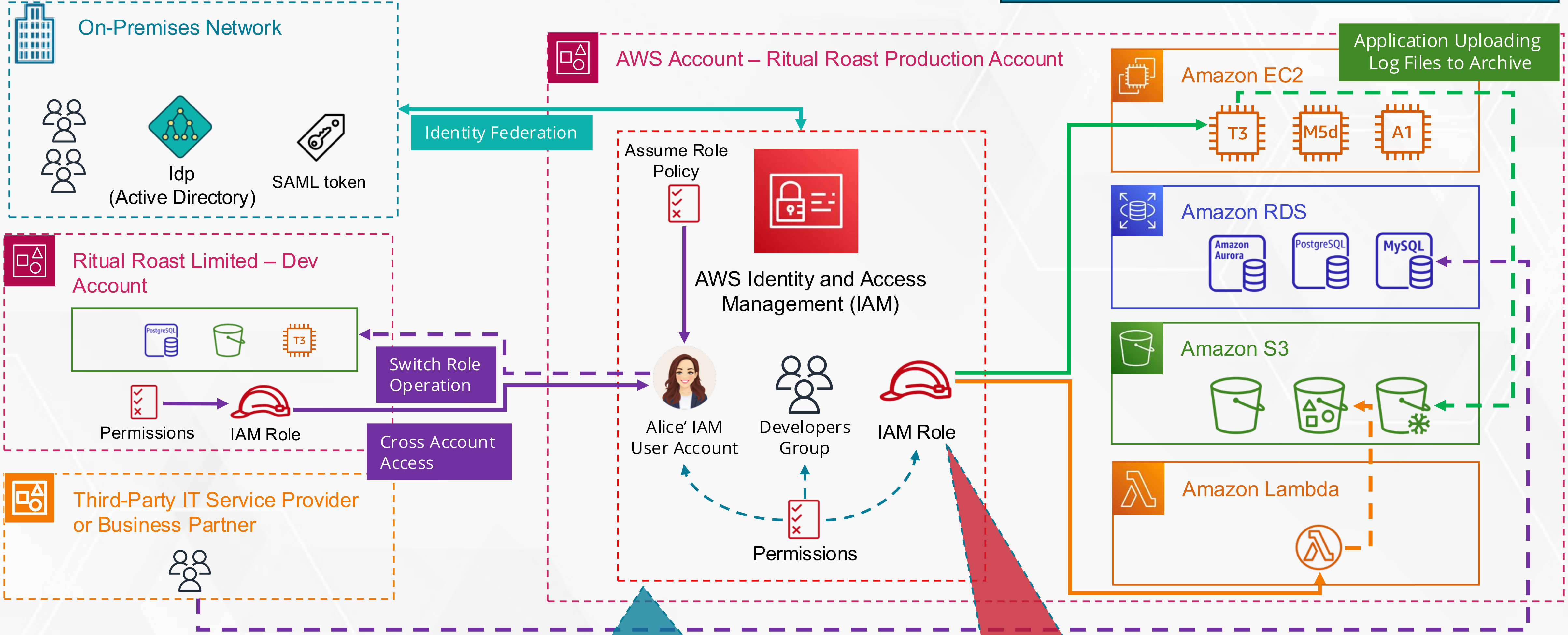
Credentials:
- Username and Password
- Access Keys

AWS IAM – Identity and Access Management



AWS IAM

An independent identity that can be assumed by any entity with the permissions to do so. An IAM role grants temporary credentials to grant access to services and resources based on a set of predefined policies.



Drawbacks:

- Long term credentials
- Cross Account Access Complexity

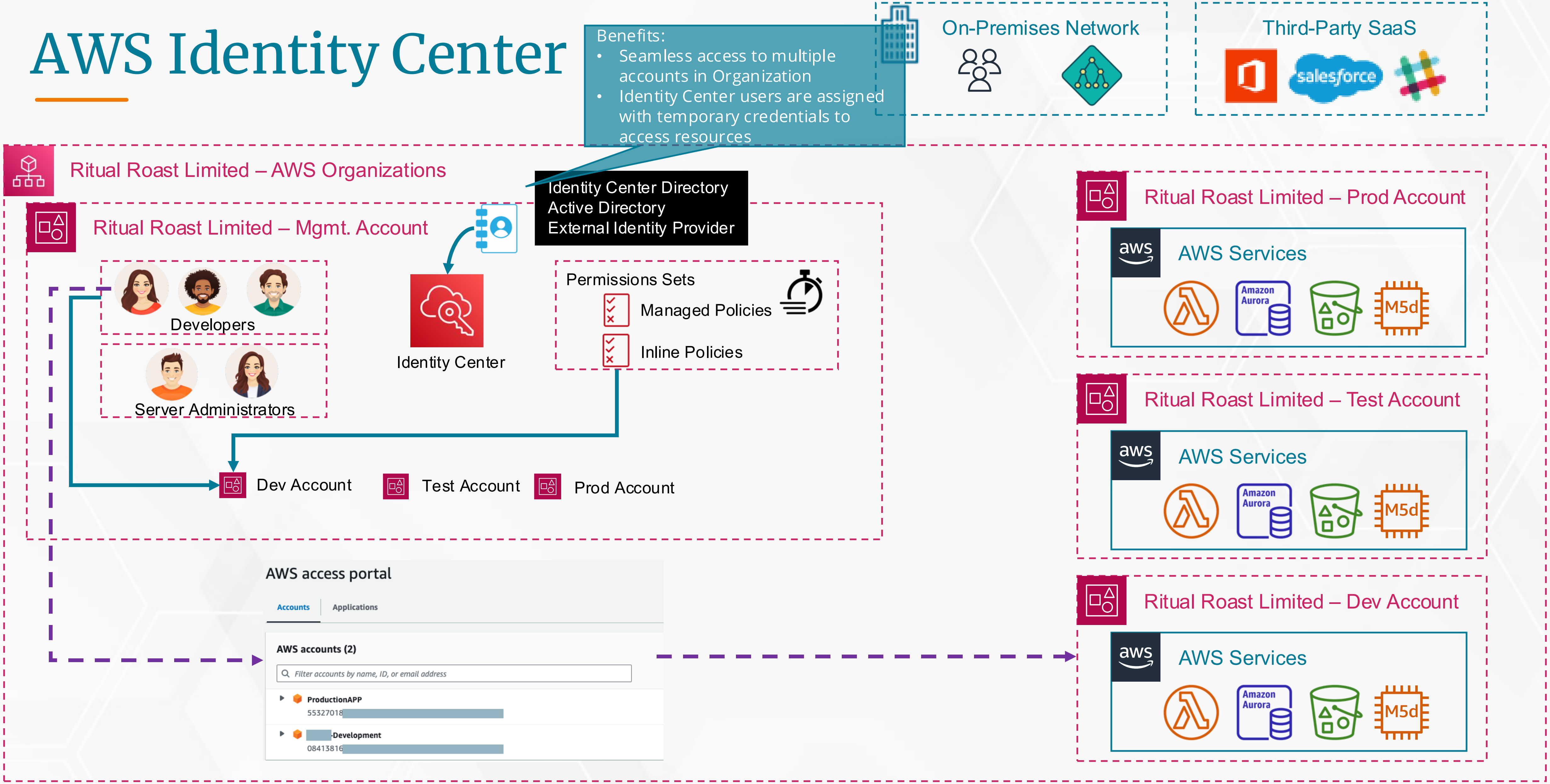
Benefits:

- IAM Roles offers temporary credentials

AWS Identity Center

Benefits:

- Seamless access to multiple accounts in Organization
- Identity Center users are assigned with temporary credentials to access resources



AWS access portal

Accounts | Applications

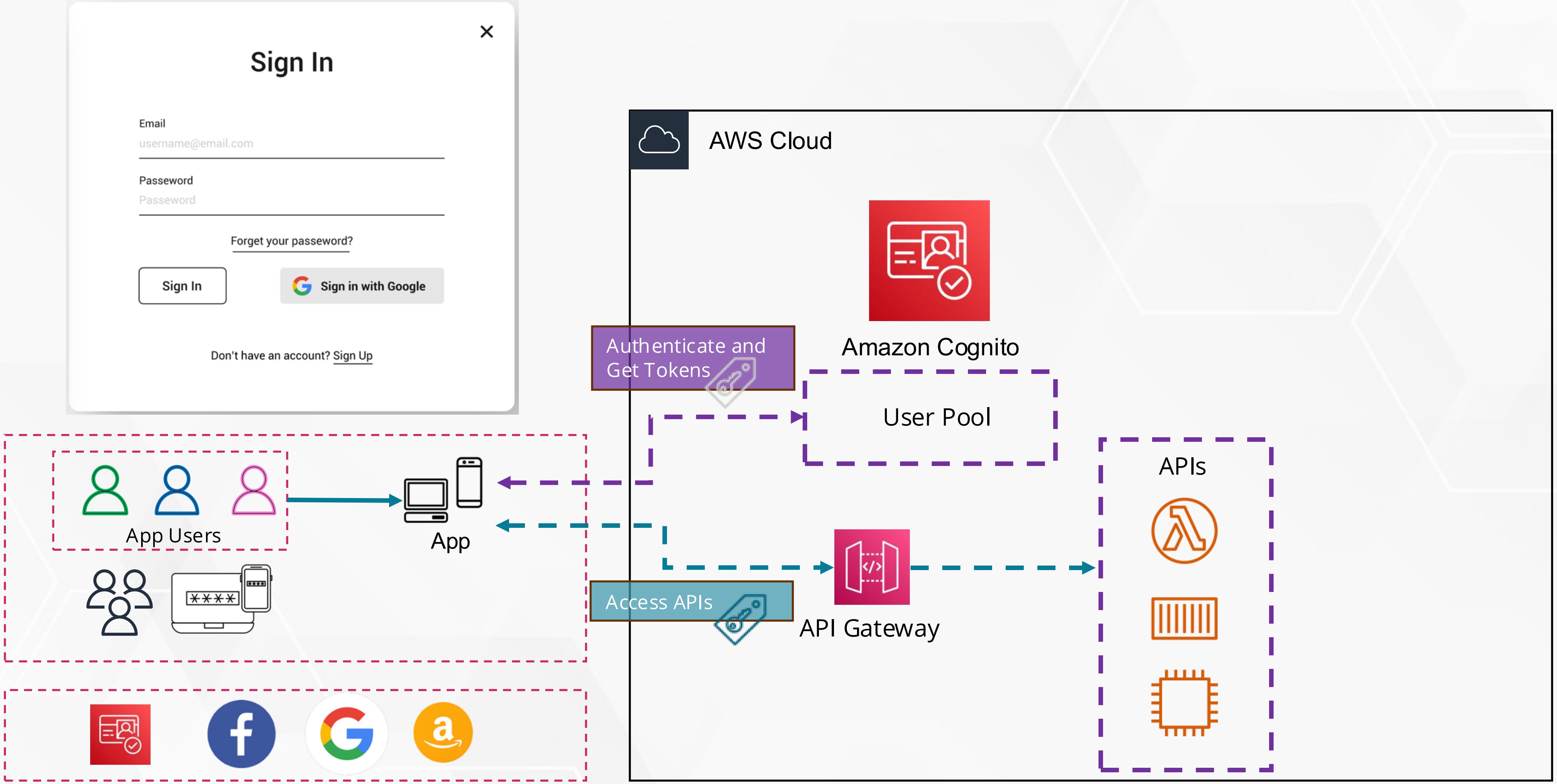
AWS accounts (2)

Filter accounts by name, ID, or email address

▸ ProductionAPP
55327018[redacted]

▸ [redacted]-Development
08413816[redacted]

Amazon Cognito –Application Users - User Pools



Sign In

Email

username@email.com

Password

Password

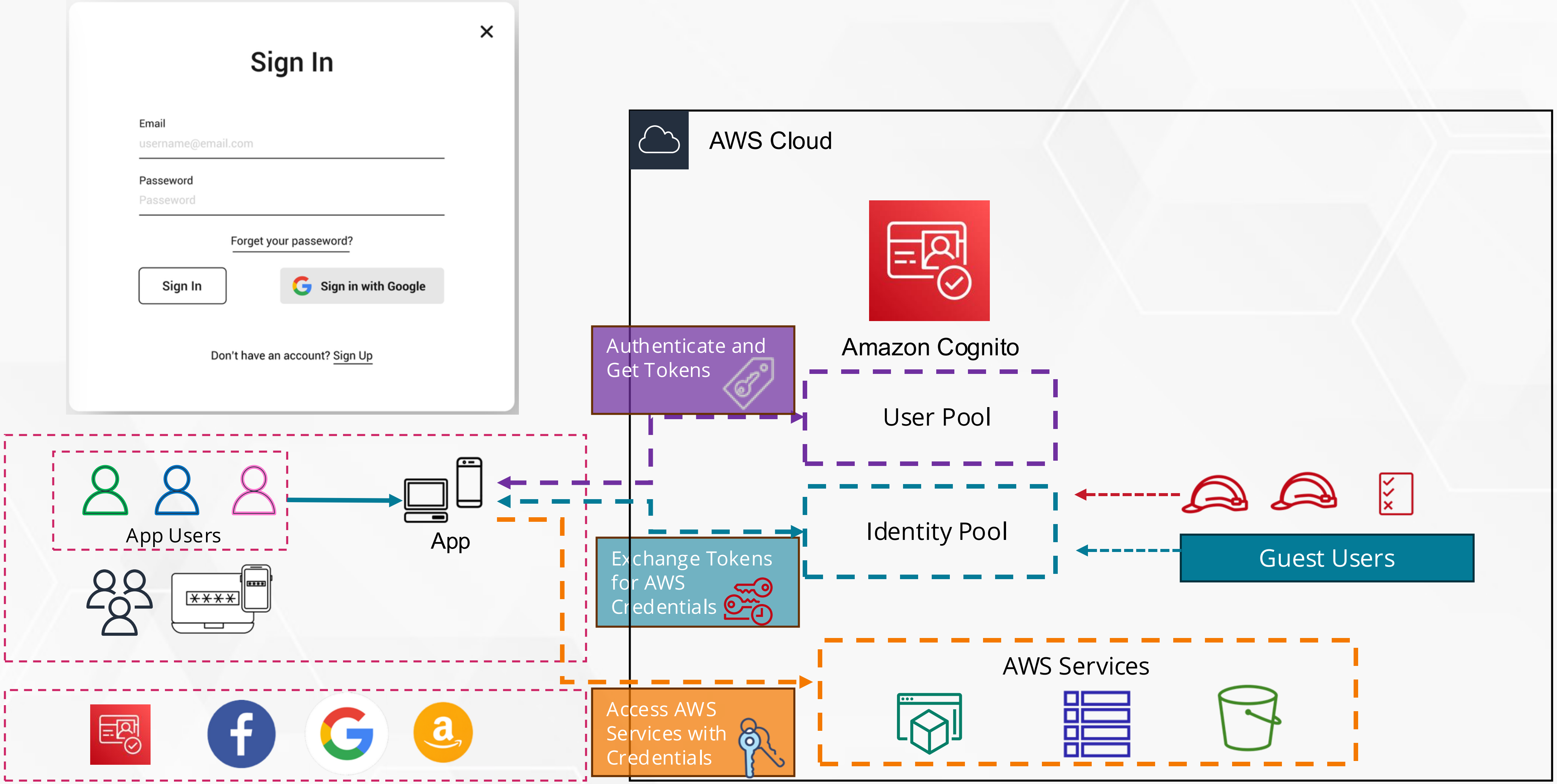
Forget your passeword?

Sign In

Sign in with Google

Don't have an account? Sign Up

Amazon Cognito – Application Users - Identity Pools



Sign In

Email
username@email.com

Password
Password

Forget your password?

Sign In | Sign in with Google

Don't have an account? Sign Up

AWS Cloud

Amazon Cognito

Authenticate and Get Tokens

User Pool

Identity Pool

Exchange Tokens for AWS Credentials

Guest Users

App Users

App

Access AWS Services with Credentials

AWS Services

Create IAM Users and Groups

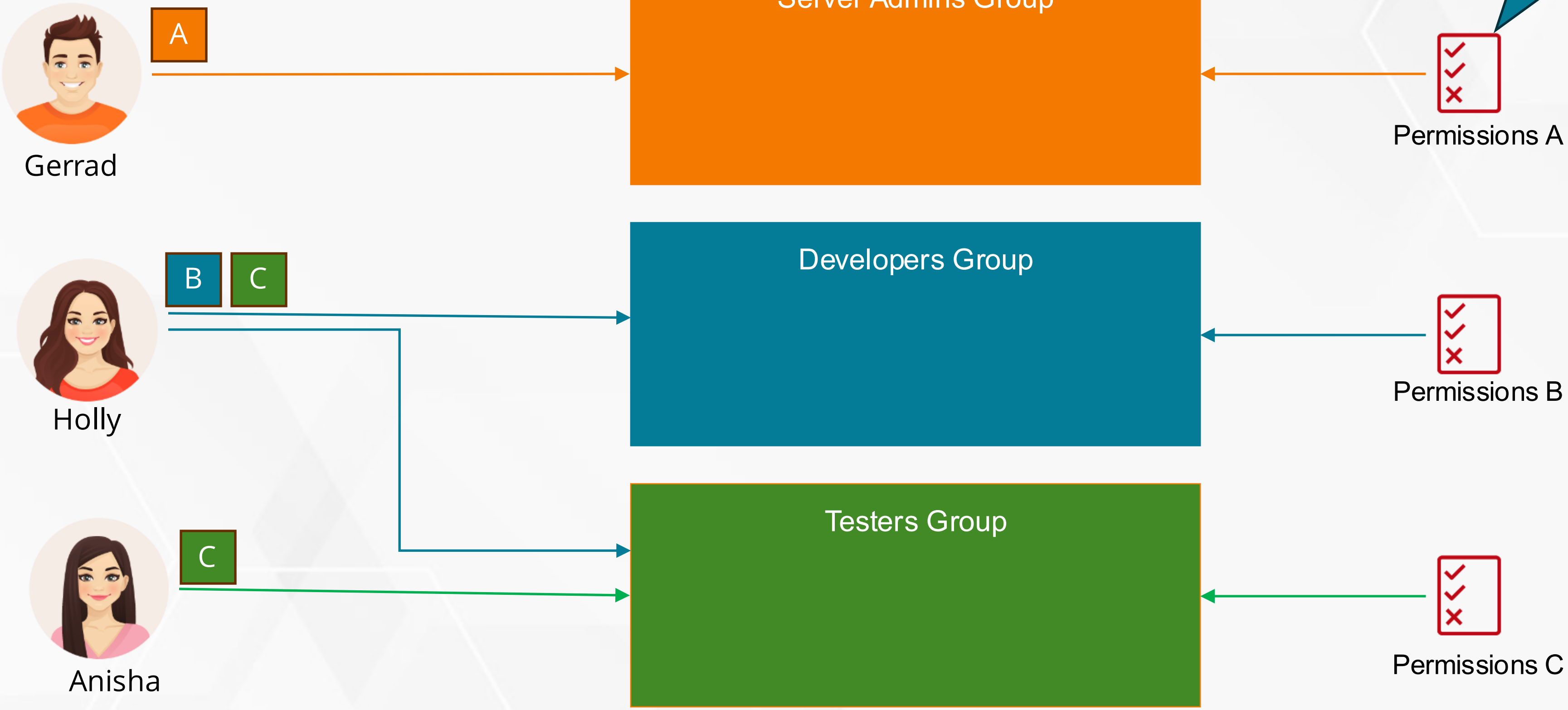
Lab Exercise

IAAS ACADEMY

IAM Policies

Features and format

Policy Application and Types



JavaScript Object Notation (JSON)

AWS Managed Policies

Customer Managed Policies

Inline Policies

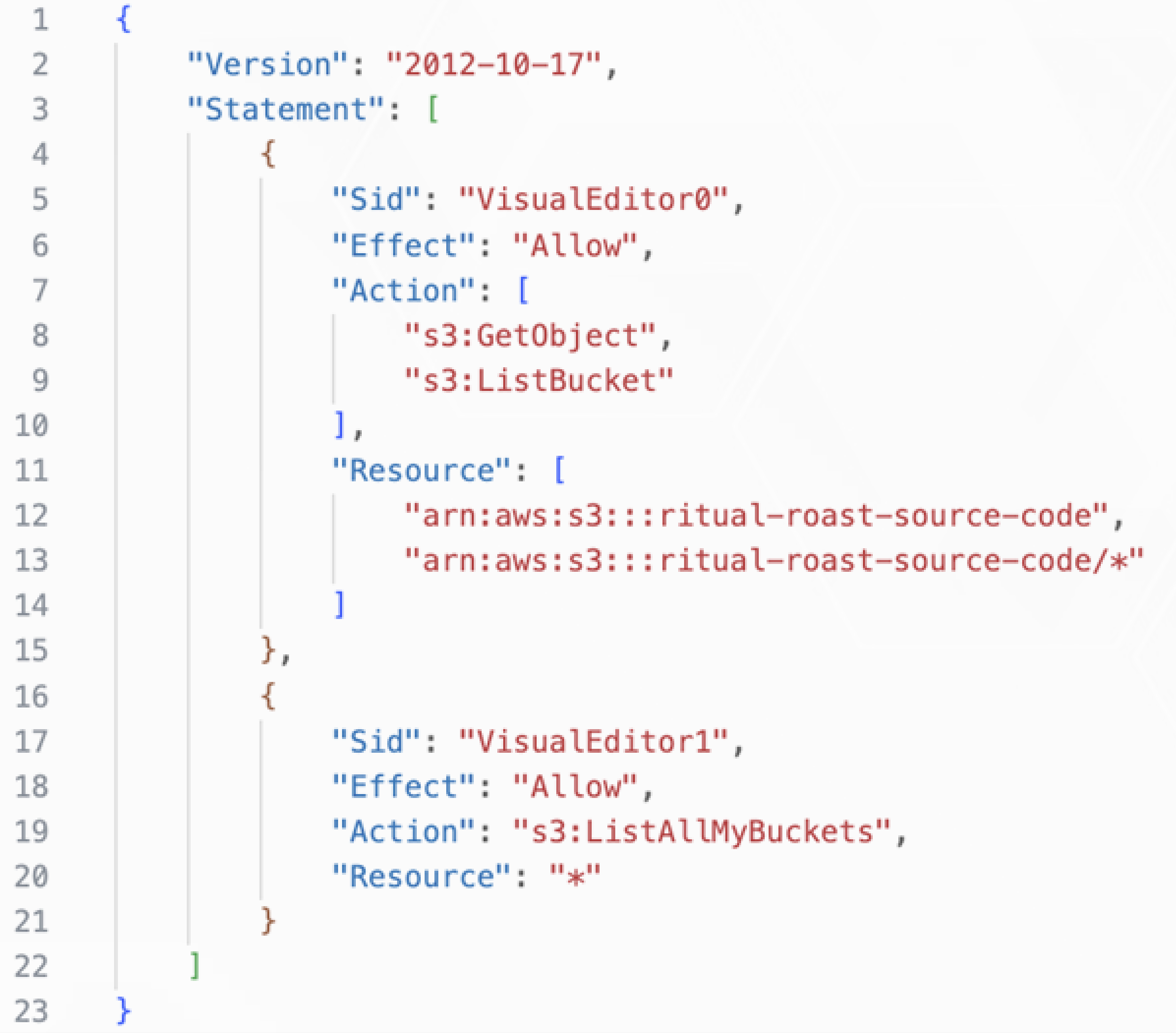
Example IAM Policy

Policy Version and Format

- Policy language version – 2012-10-17
- Statement IDs (optional)
- Statement: one or more individual statements

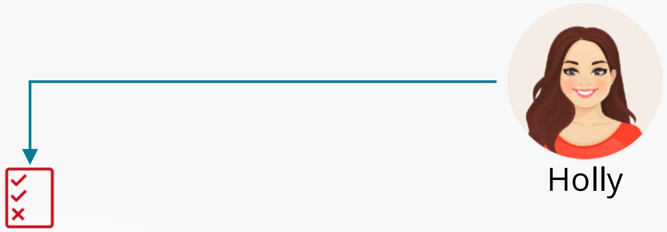
Statements consist of:

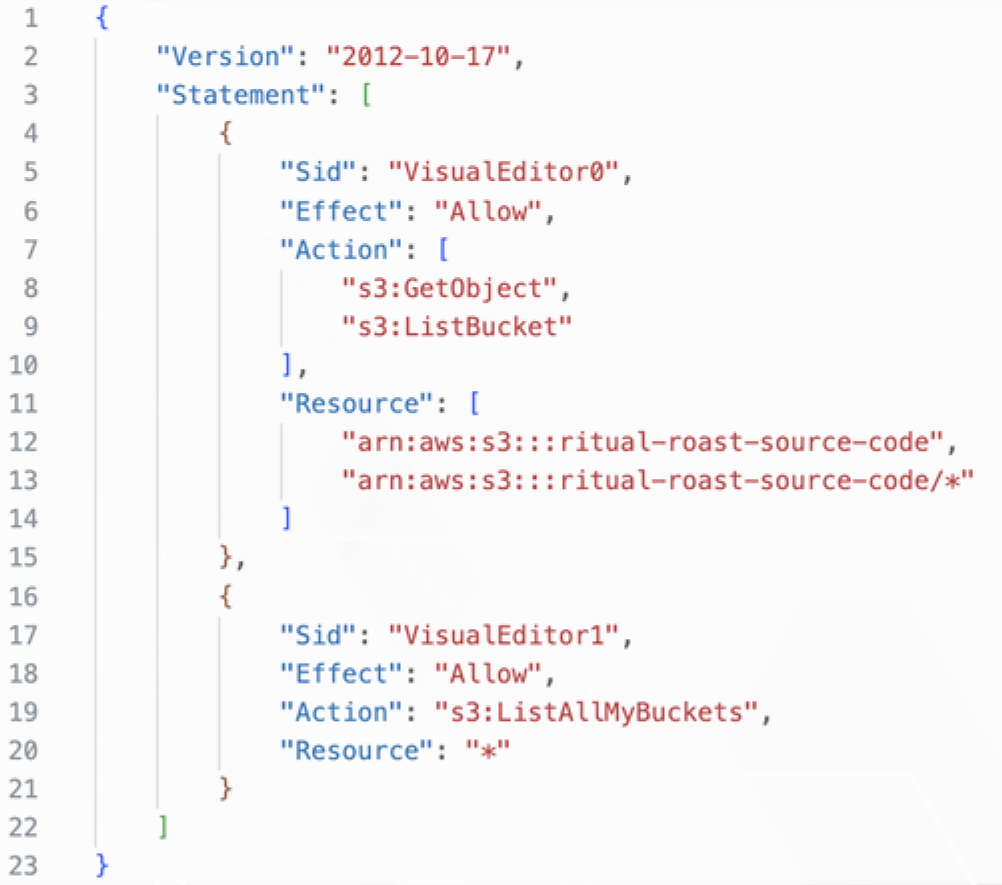
- Sid: an optional identifier
- Effect: determine if the policy will ‘Allow’ or ‘Deny’ an action
- Action: list of actions this policy allows or denies
- Resource: list of resources to which this policy applies to



```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::ritual-roast-source-code",
                "arn:aws:s3:::ritual-roast-source-code/*"
            ]
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": "s3:ListAllMyBuckets",
            "Resource": "*"
        }
    ]
}
```

IAM Policies Effects





```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::ritual-roast-source-code",
                "arn:aws:s3:::ritual-roast-source-code/*"
            ]
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": "s3:ListAllMyBuckets",
            "Resource": "*"
        }
    ]
}
```

ARN Formats

- arn:partition:service:region:account-id:resource-id
- arn:partition:service:region:account-id:resource-type/resource-id
- arn:partition:service:region:account-id:resource-type:resource-id

arn:aws:ec2:us-east-1:905418291234:instance/i-0d52f19f2e93eb3c8

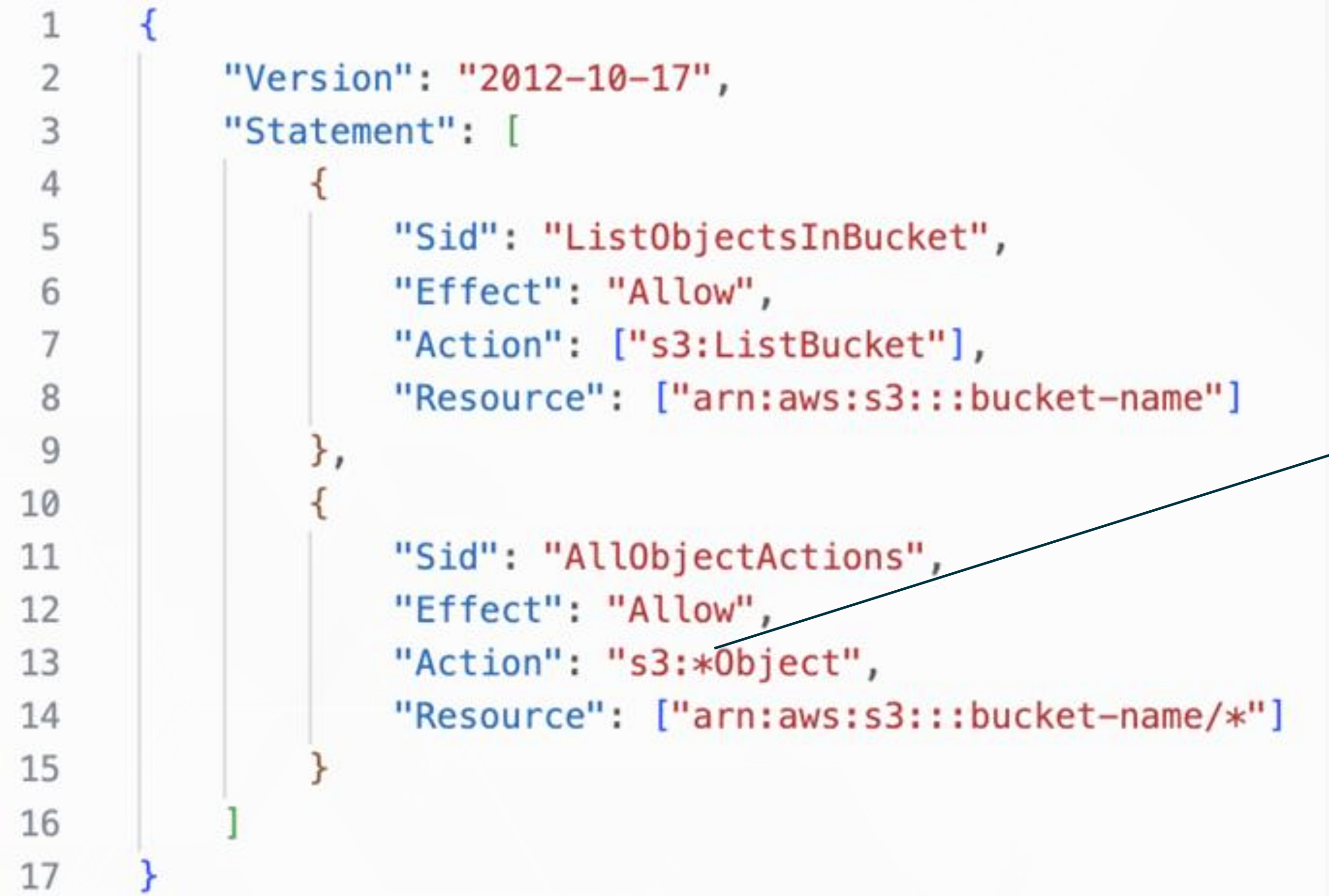
Ritual Roast Source Code
ARN: *arn:aws:s3:::ritual-roast-source-code*

List Buckets in Account

List Objects in Bucket

Get Objects in Bucket

IAM Policy Example

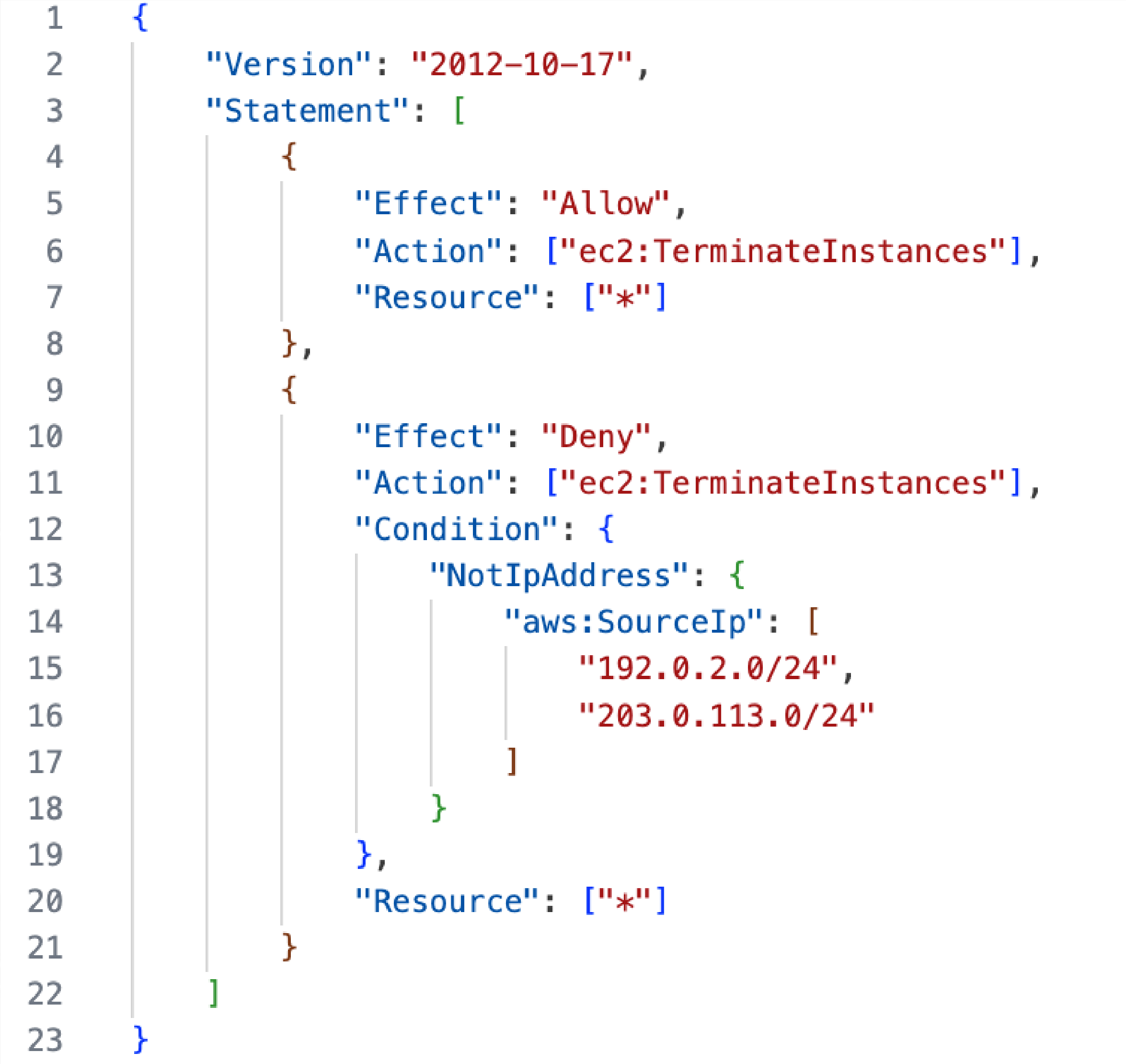


```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ListObjectsInBucket",
            "Effect": "Allow",
            "Action": ["s3:ListBucket"],
            "Resource": ["arn:aws:s3:::bucket-name"]
        },
        {
            "Sid": "AllObjectActions",
            "Effect": "Allow",
            "Action": "s3:*Object",
            "Resource": ["arn:aws:s3:::bucket-name/*"]
        }
    ]
}
```

Bucket
ARN: *arn:aws:s3:::bucket-name*

Using wildcards
(asterisks *)

IAM Policy Example with Conditional Statements



```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": ["ec2:TerminateInstances"],
            "Resource": ["*"]
        },
        {
            "Effect": "Deny",
            "Action": ["ec2:TerminateInstances"],
            "Condition": {
                "NotIpAddress": {
                    "aws:SourceIp": [
                        "192.0.2.0/24",
                        "203.0.113.0/24"
                    ]
                }
            },
            "Resource": ["*"]
        }
    ]
}
```

Termination of EC2 instances

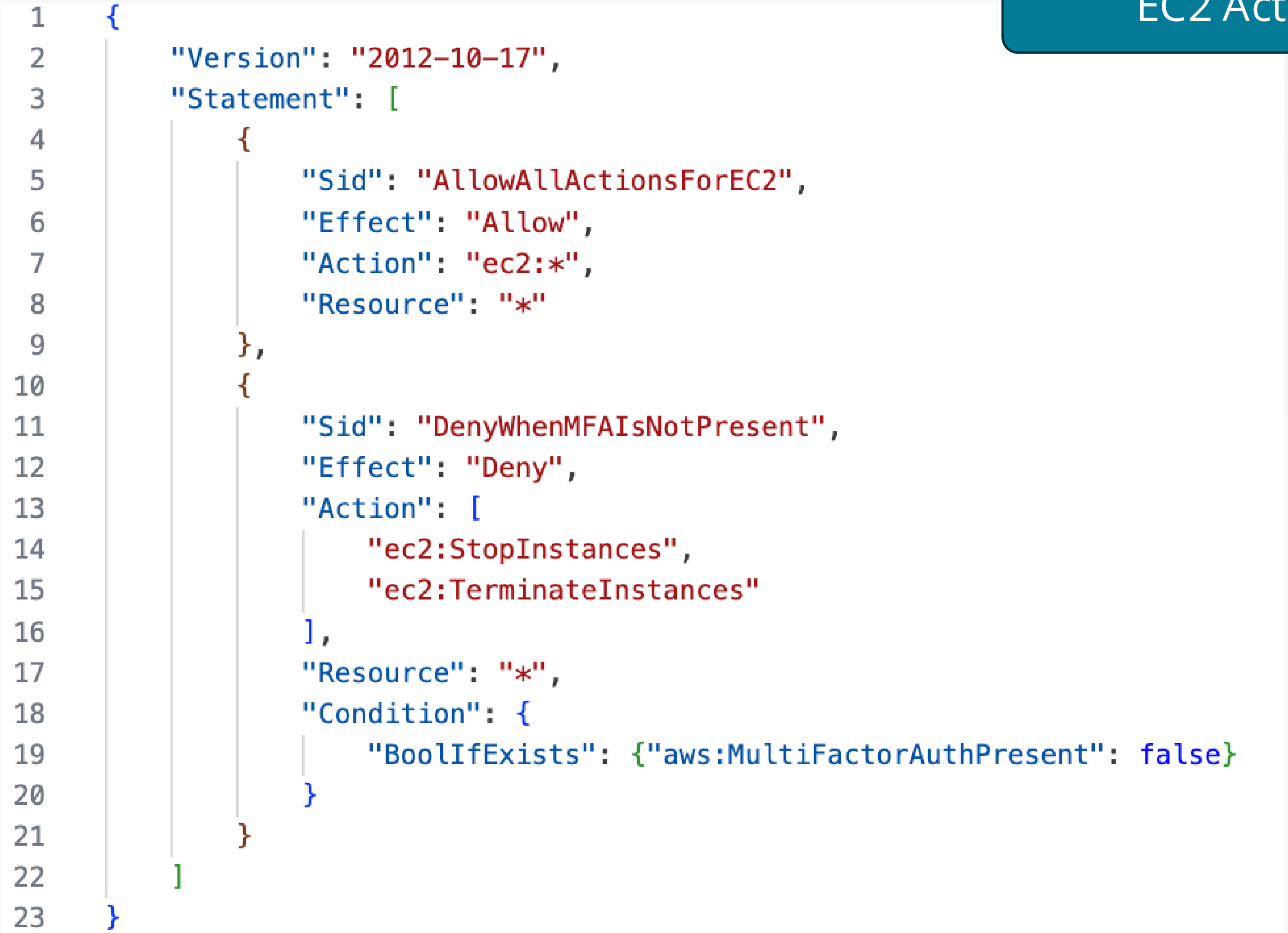
M5n instance

T2 instance

Allows the action to override IAM Policy *deny by default* feature

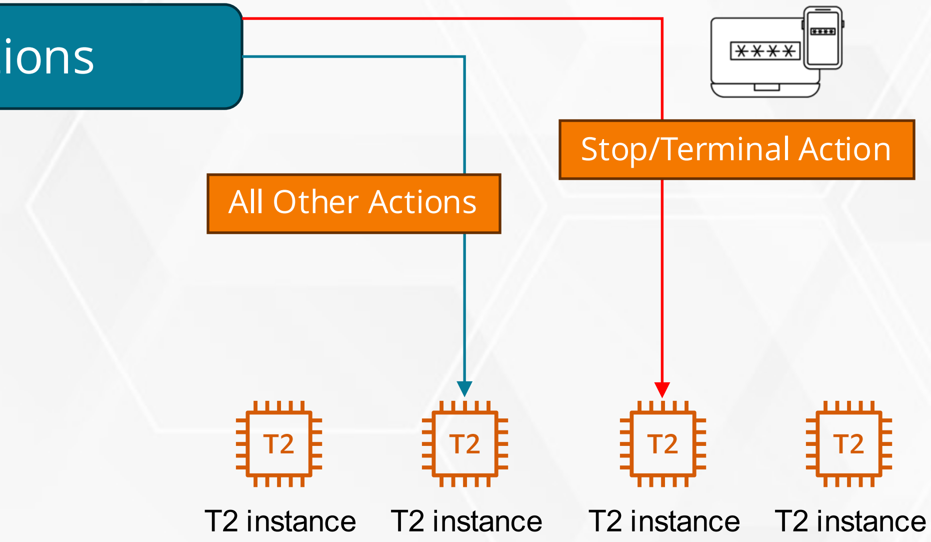
condition-based **deny** when the request comes from IP addresses outside the allowed ranges

Example Conditional Statements



```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowAllActionsForEC2",
            "Effect": "Allow",
            "Action": "ec2:*",
            "Resource": "*"
        },
        {
            "Sid": "DenyWhenMFAIsNotPresent",
            "Effect": "Deny",
            "Action": [
                "ec2:StopInstances",
                "ec2:TerminateInstances"
            ],
            "Resource": "*",
            "Condition": {
                "BoolIfExists": {"aws:MultiFactorAuthPresent": false}
            }
        }
    ]
}
```

EC2 Actions



All Other Actions

Stop/Terminal Action

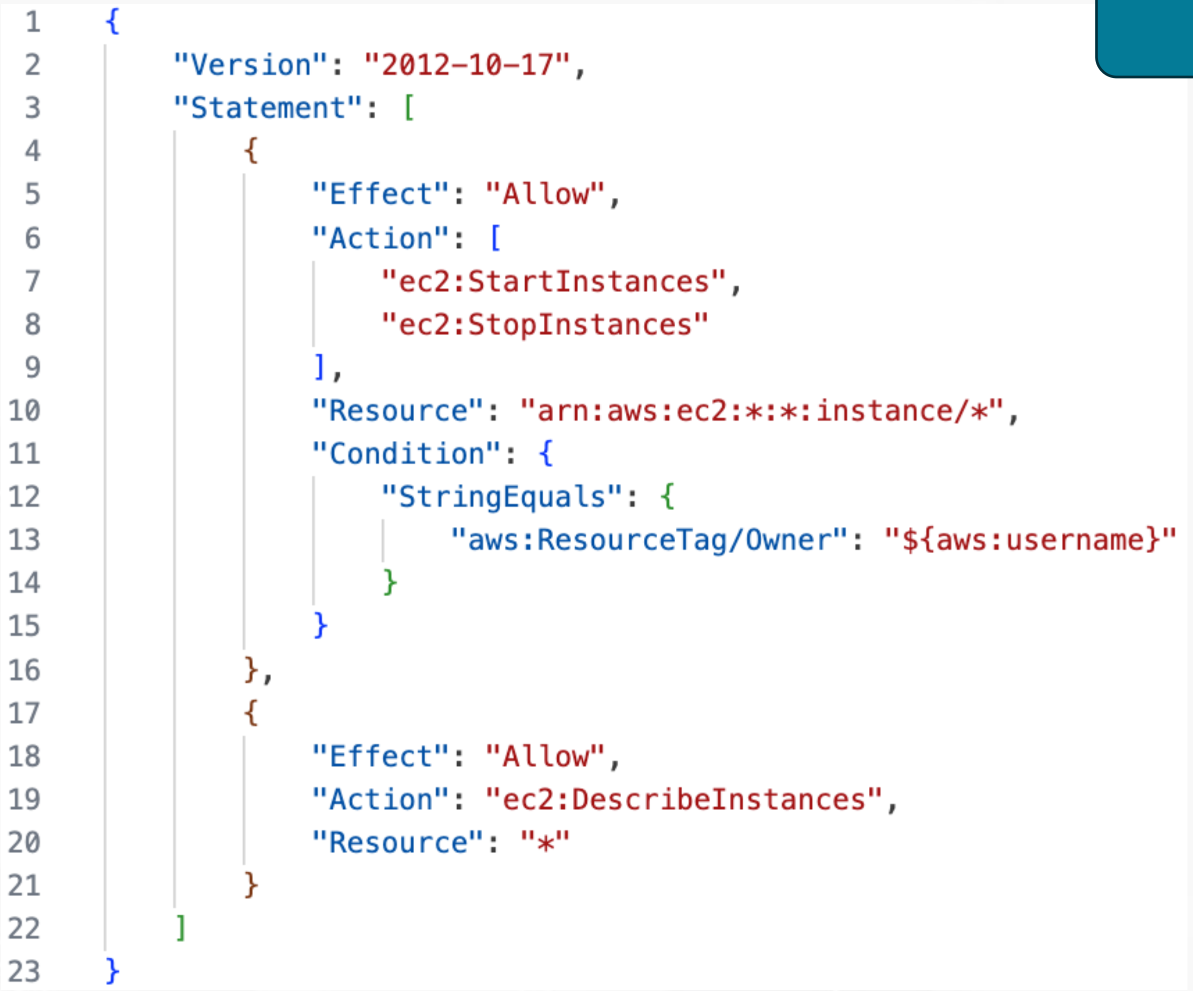
T2 instance

T2 instance

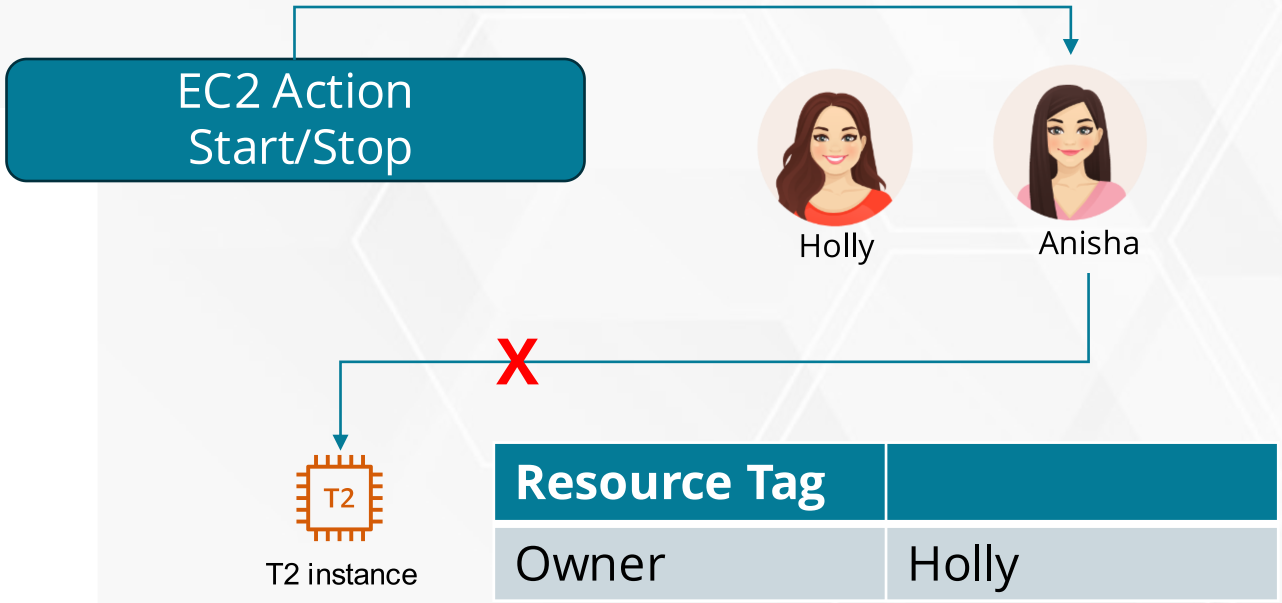
T2 instance

T2 instance

Example Conditional Statements

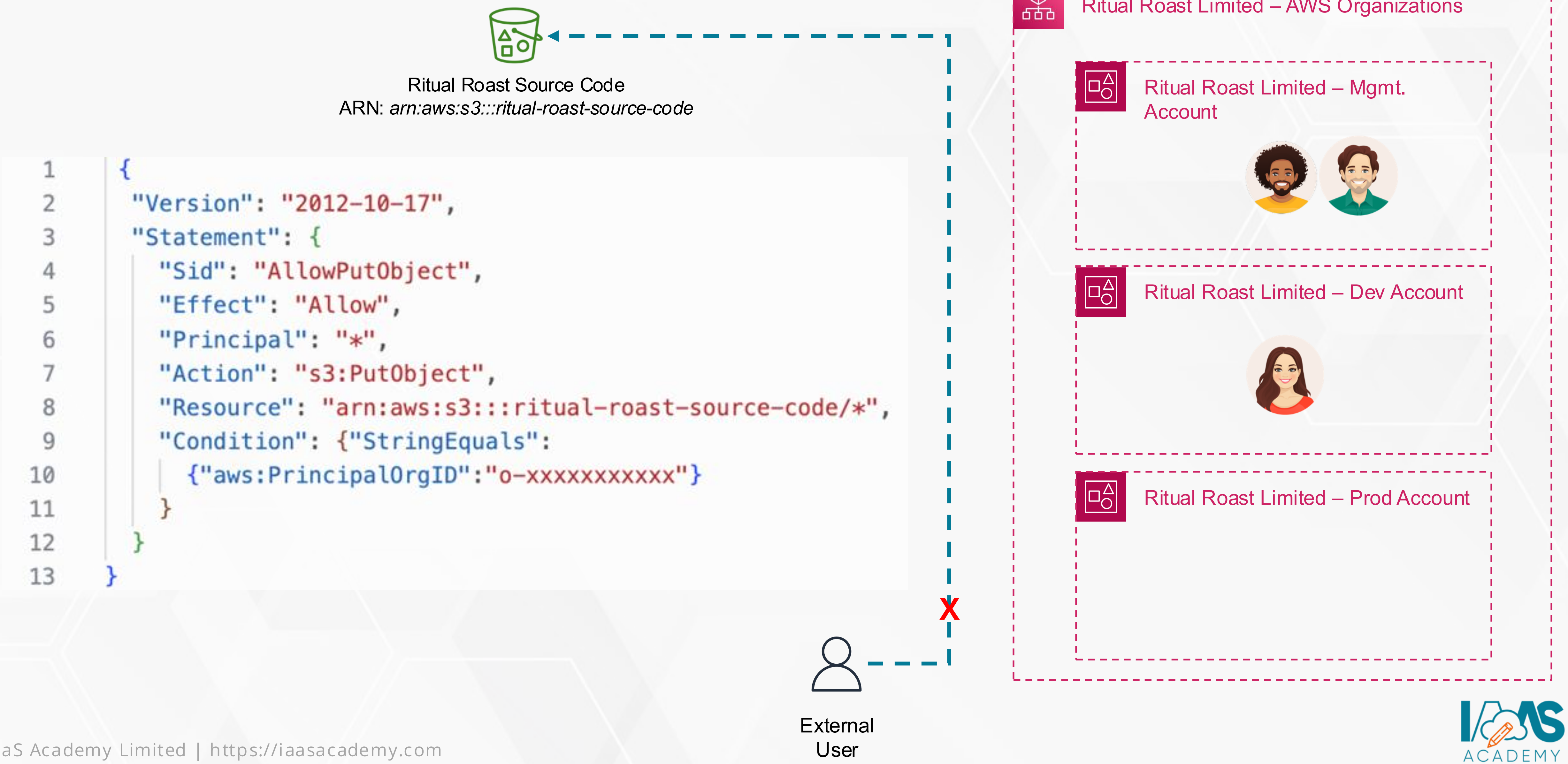


```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:StartInstances",
                "ec2:StopInstances"
            ],
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/Owner": "${aws:username}"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeInstances",
            "Resource": "*"
        }
    ]
}
```



Resource Tag	
Owner	Holly

Example Conditional Statements



Ritual Roast Source Code
ARN: *arn:aws:s3:::ritual-roast-source-code*

```
{
 "Version": "2012-10-17",
 "Statement": {
   "Sid": "AllowPutObject",
   "Effect": "Allow",
   "Principal": "*",
   "Action": "s3:PutObject",
   "Resource": "arn:aws:s3:::ritual-roast-source-code/*",
   "Condition": {"StringEquals":
     {"aws:PrincipalOrgID":"o-xxxxxxxxxx"}
   }
 }
}
```

X
External User
Ritual Roast Limited – AWS Organizations
Ritual Roast Limited – Mgmt. Account
Ritual Roast Limited – Dev Account
Ritual Roast Limited – Prod Account

IAM Policies

Hands-On Labs

Accessing AWS Account via CLI

Configure Command Line Interface
Hands-On Labs

IAM Policy Simulator

Hands-On Labs

Create an IAM Role

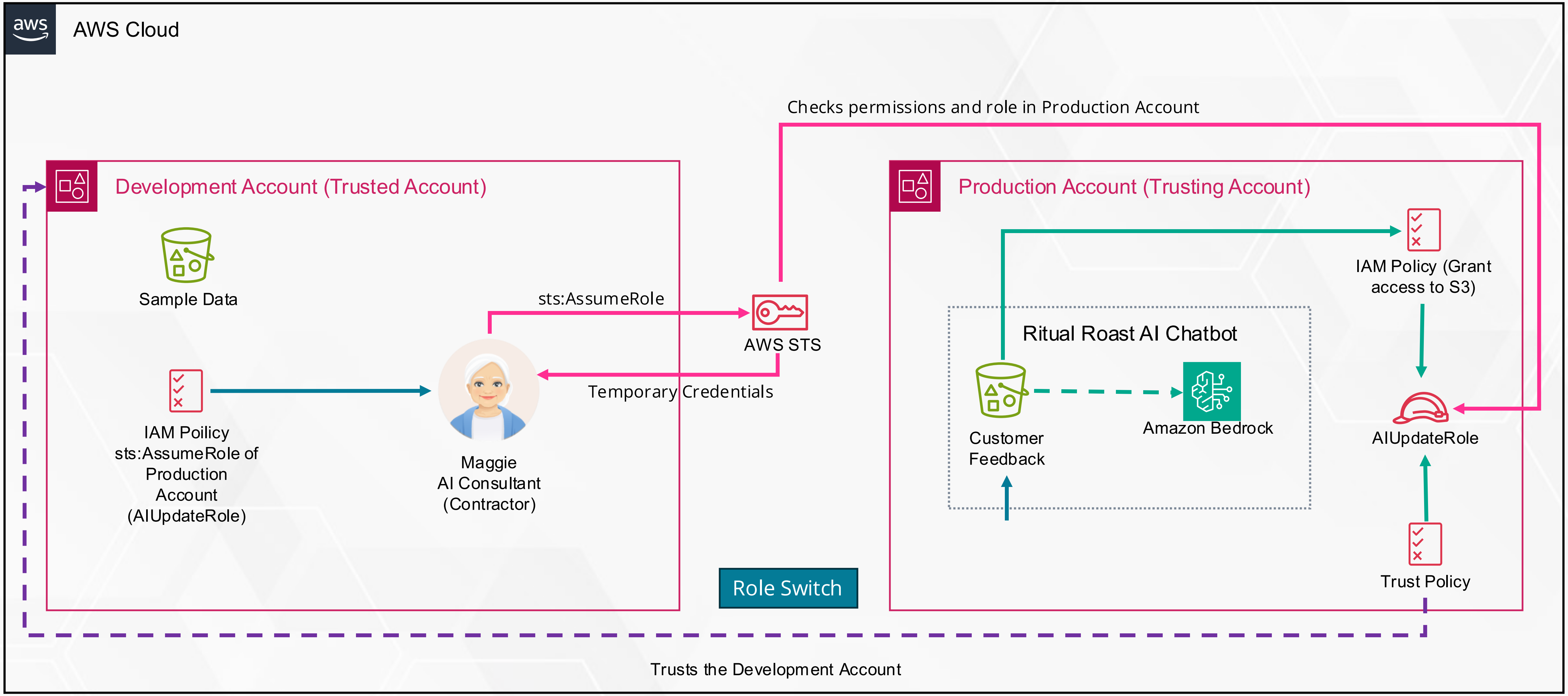
Hands-On Labs

IAAS ACADEMY

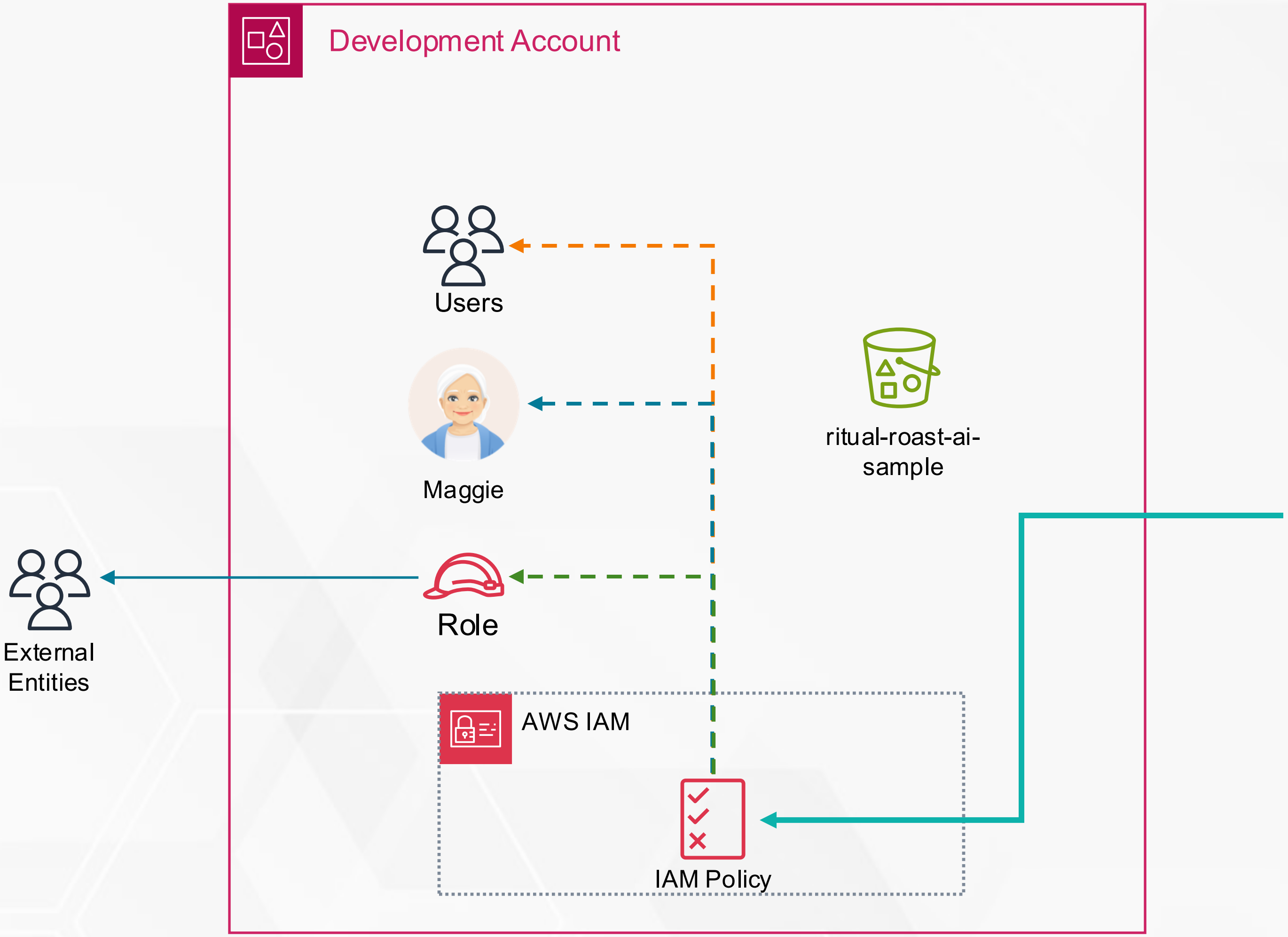
Create Resource-based Policies

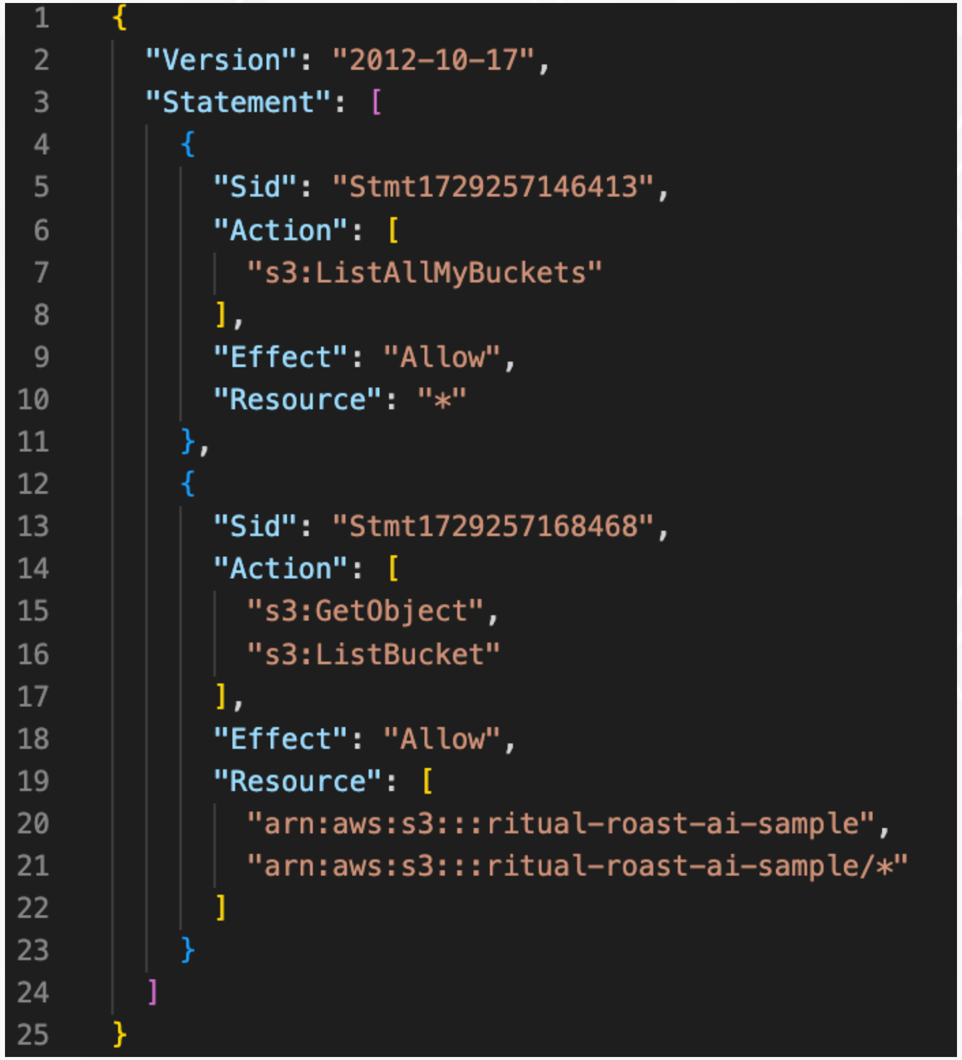
Hands-On Labs

Another Use Case for Cross Account Access



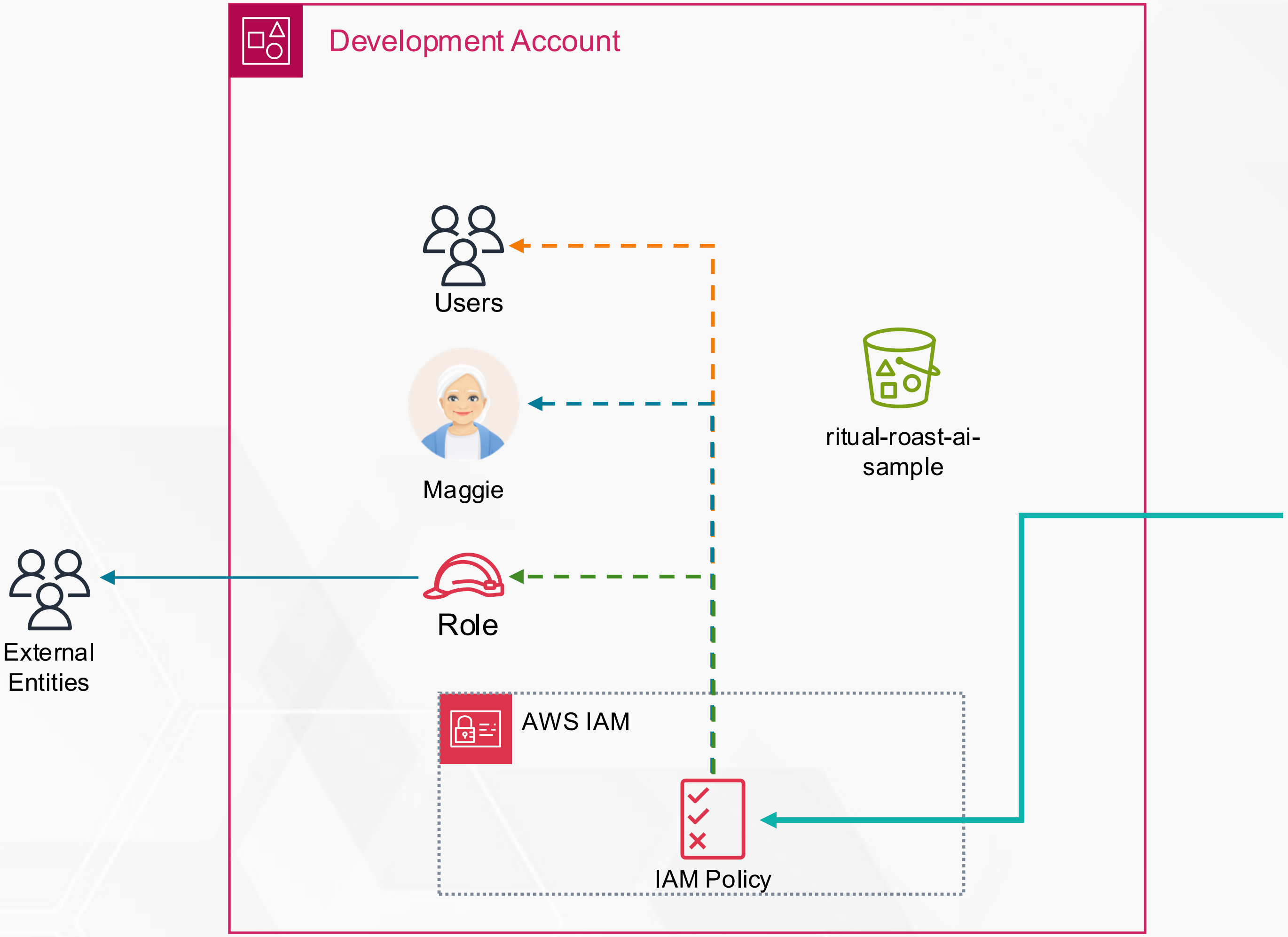
IAM-based policies

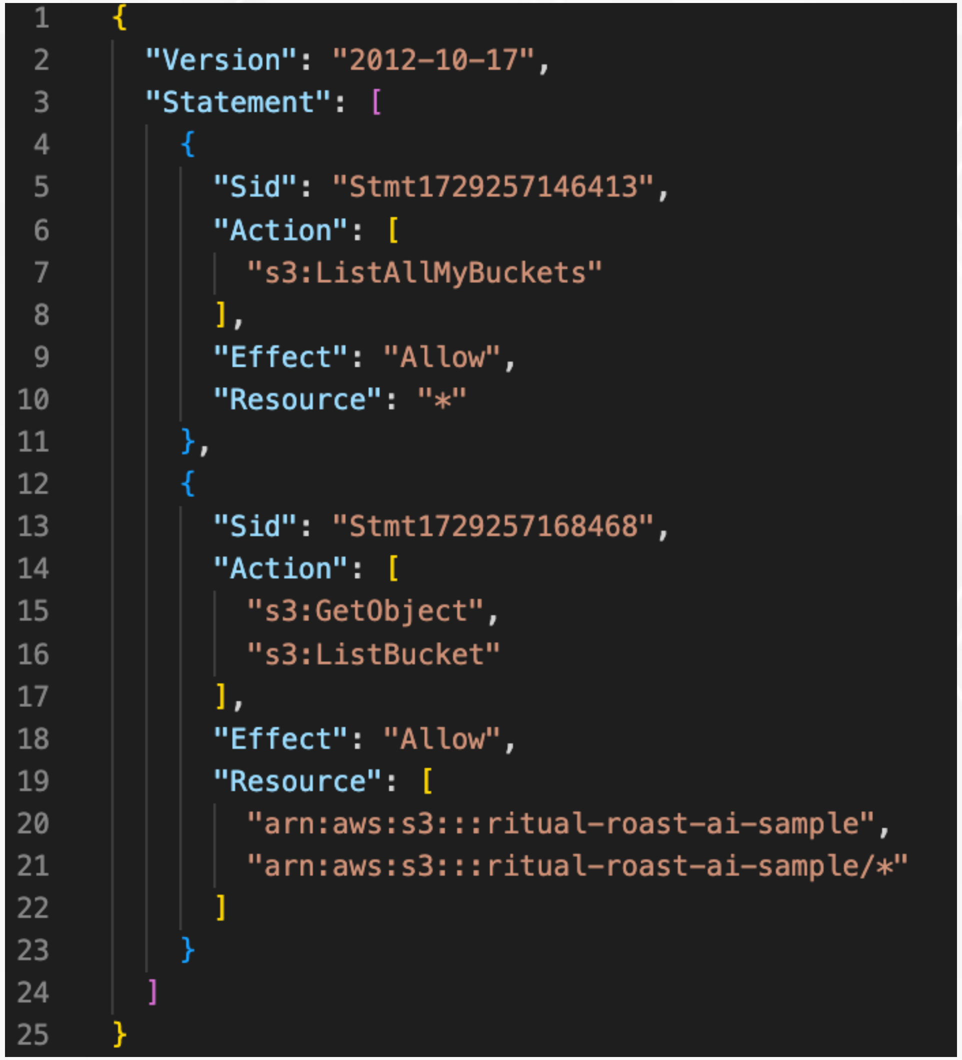




```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1729257146413",
      "Action": [
        "s3:ListAllMyBuckets"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Sid": "Stmt1729257168468",
      "Action": [
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3:::ritual-roast-ai-sample",
        "arn:aws:s3:::ritual-roast-ai-sample/*"
      ]
    }
  ]
}
```

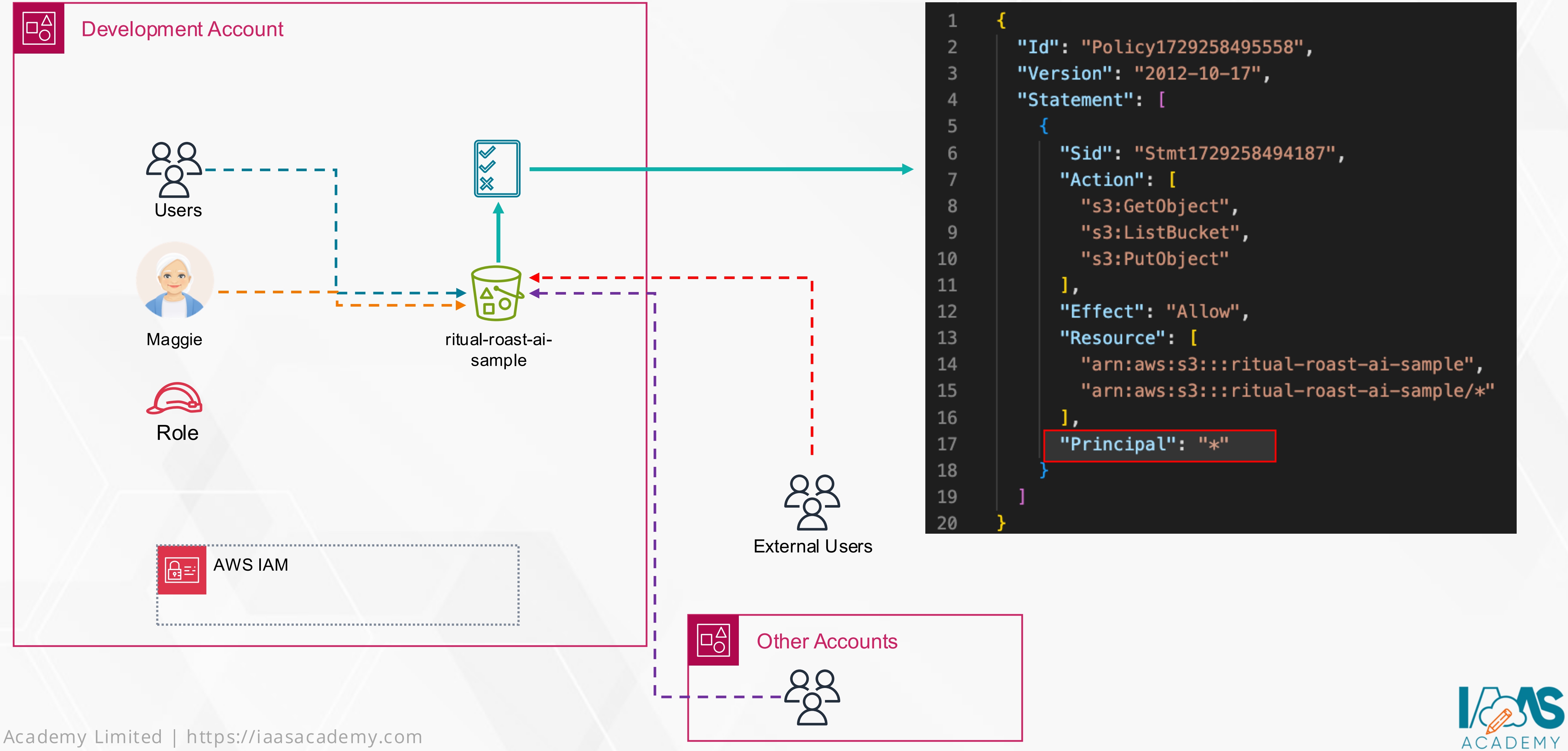
IAM-based policies





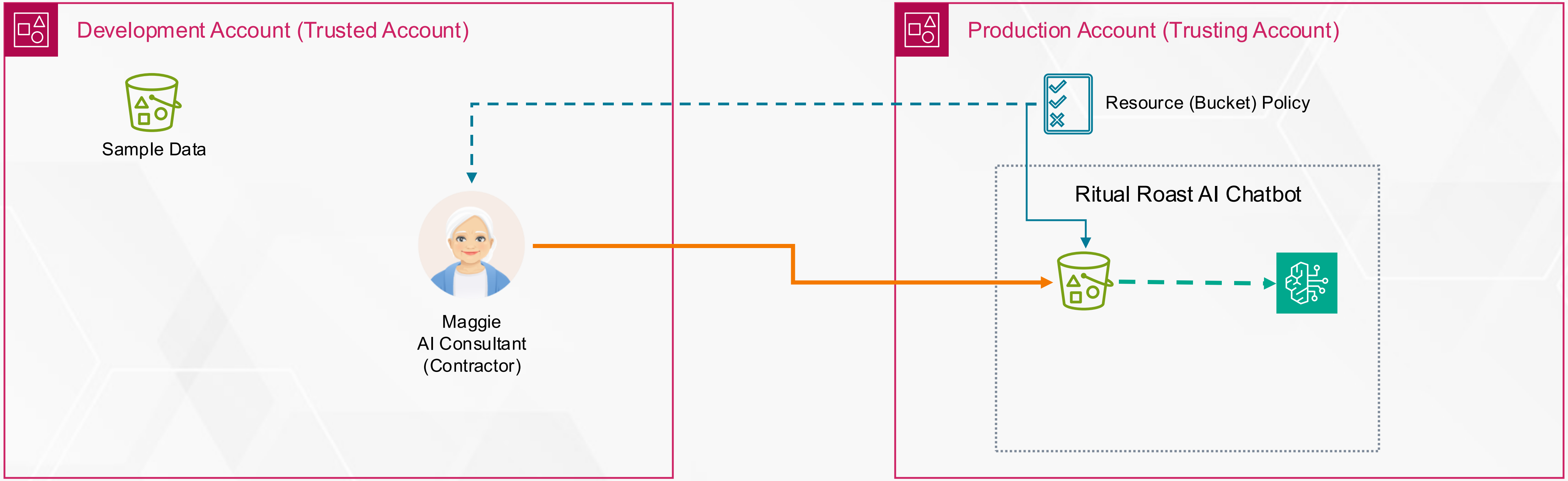
```json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1729257146413",
      "Action": [
        "s3:ListAllMyBuckets"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Sid": "Stmt1729257168468",
      "Action": [
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3:::ritual-roast-ai-sample",
        "arn:aws:s3:::ritual-roast-ai-sample/*"
      ]
    }
  ]
}
```

Resource-based policies



```
{
  "Id": "Policy1729258495558",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1729258494187",
      "Action": [
        "s3:GetObject",
        "s3:ListBucket",
        "s3:PutObject"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3:::ritual-roast-ai-sample",
        "arn:aws:s3:::ritual-roast-ai-sample/*"
      ],
      "Principal": "*"
    }
  ]
}
```

Cross Account with Resource-based policy



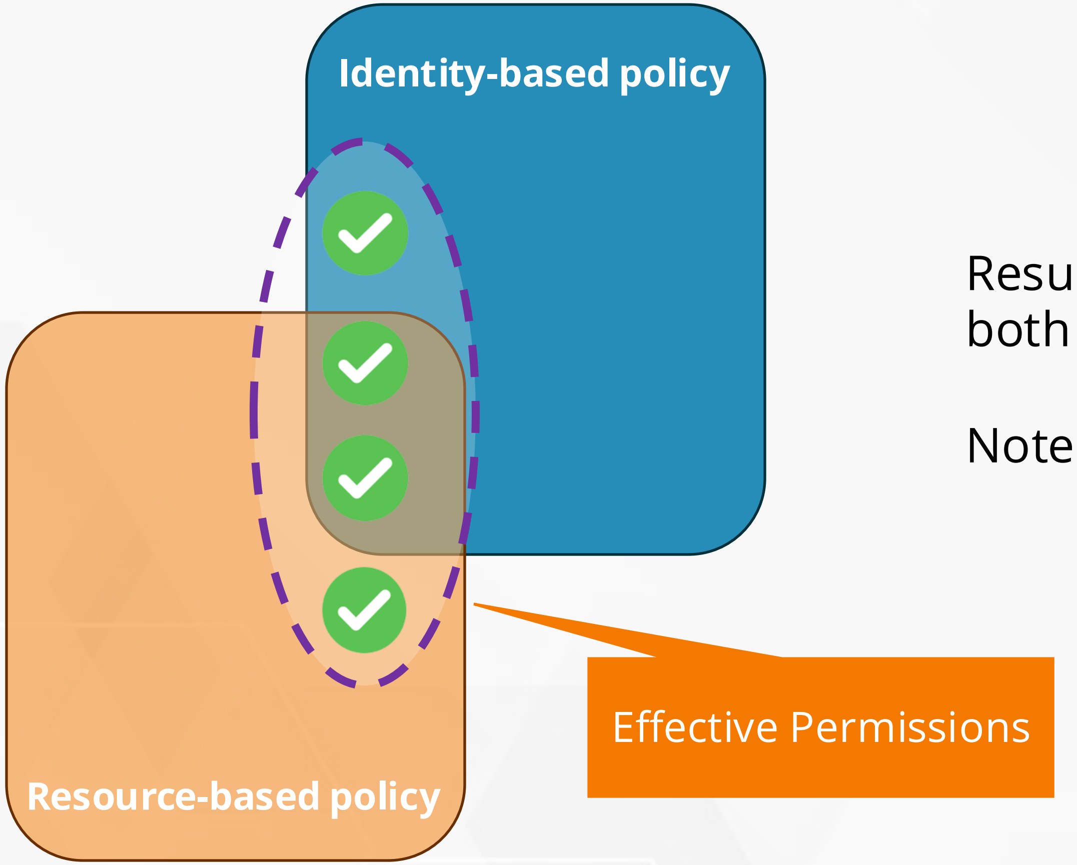
IAM Policy Evaluation

Policy Logic

Evaluating Policies – Types of Policies

- **Identity-based policies (IAM Policies) -** are attached to an IAM identity. Define the permission that identity has in the AWS account.
- **Resource-based policies** – attached to specific resources and apply to a principal (account, user, role, federated users defining what action can be taken against the resource.
- **Permission boundaries** – set the maximum permission an IAM policy can grant an IAM entity.
- **Service Control Policies (SCPs)** – specify maximum permissions for Organization or OU and apply to principals in member accounts, including root users.
- **Session policies** – allow you to define policies for temporary sessions for a role or federated user, using the AssumeRole* API operations:
    - **AssumeRole** to assume a role
    - **AssumeRoleWithSAML** for identities authenticated with a SAML 2.0 compatible identity provider
    - **AssumeRoleWithWebIdentity** for authentication with web identity providers, e.g. OAuth 2.0

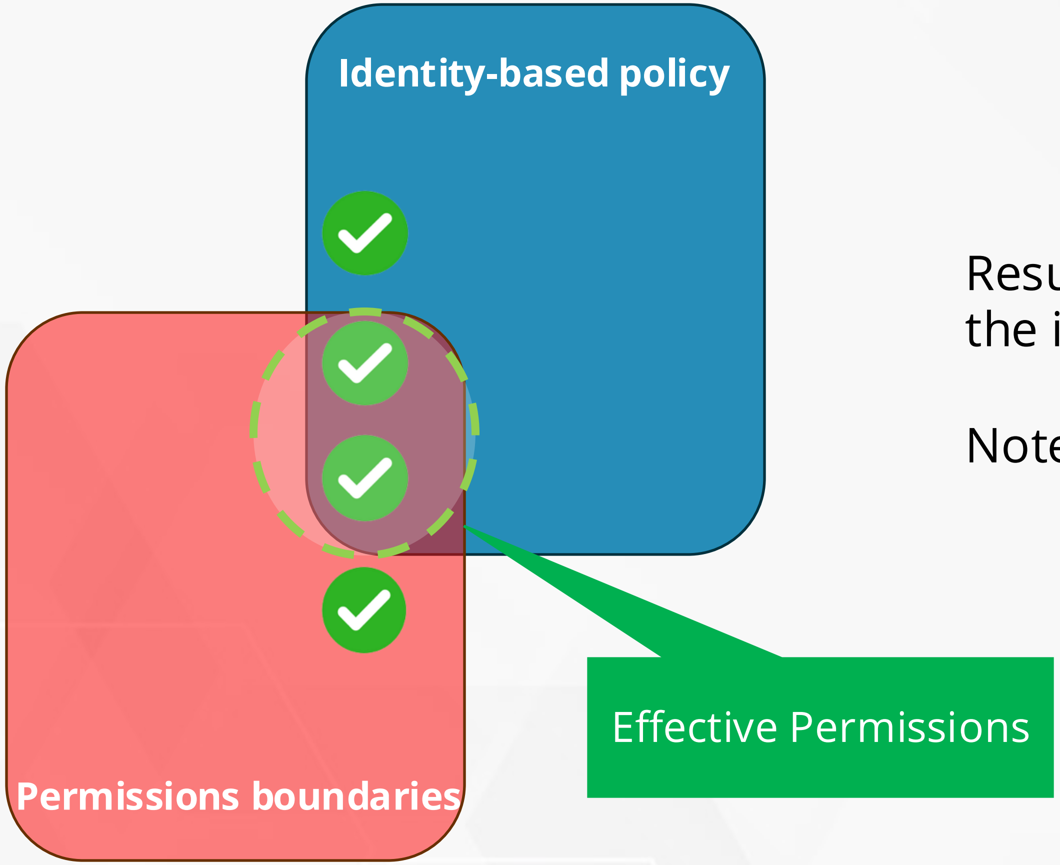
Identity-based with resource-based policies



Resulting permissions are the total permissions of both identity and resource-based policies

Note: an explicit deny always overrides any allow

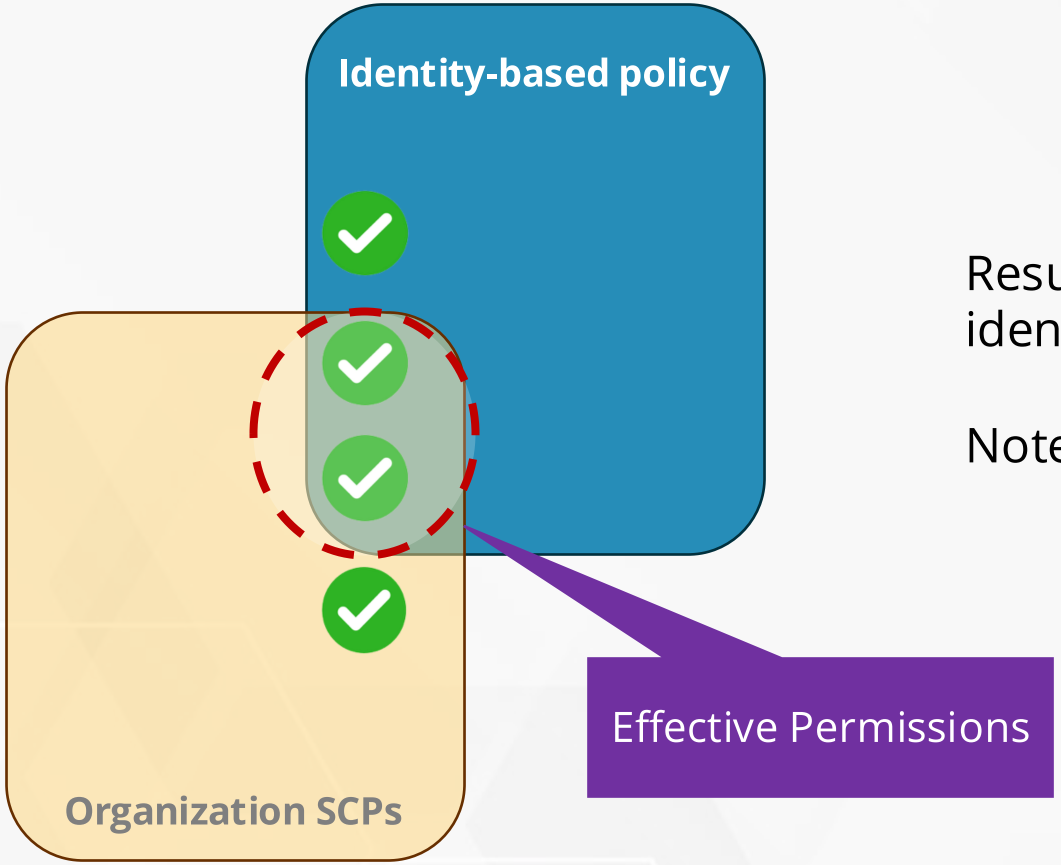
Identity-based policies with permission boundaries



Resulting permissions are the intersection of the identity policy and the permission boundary

Note: an explicit deny always overrides any allow

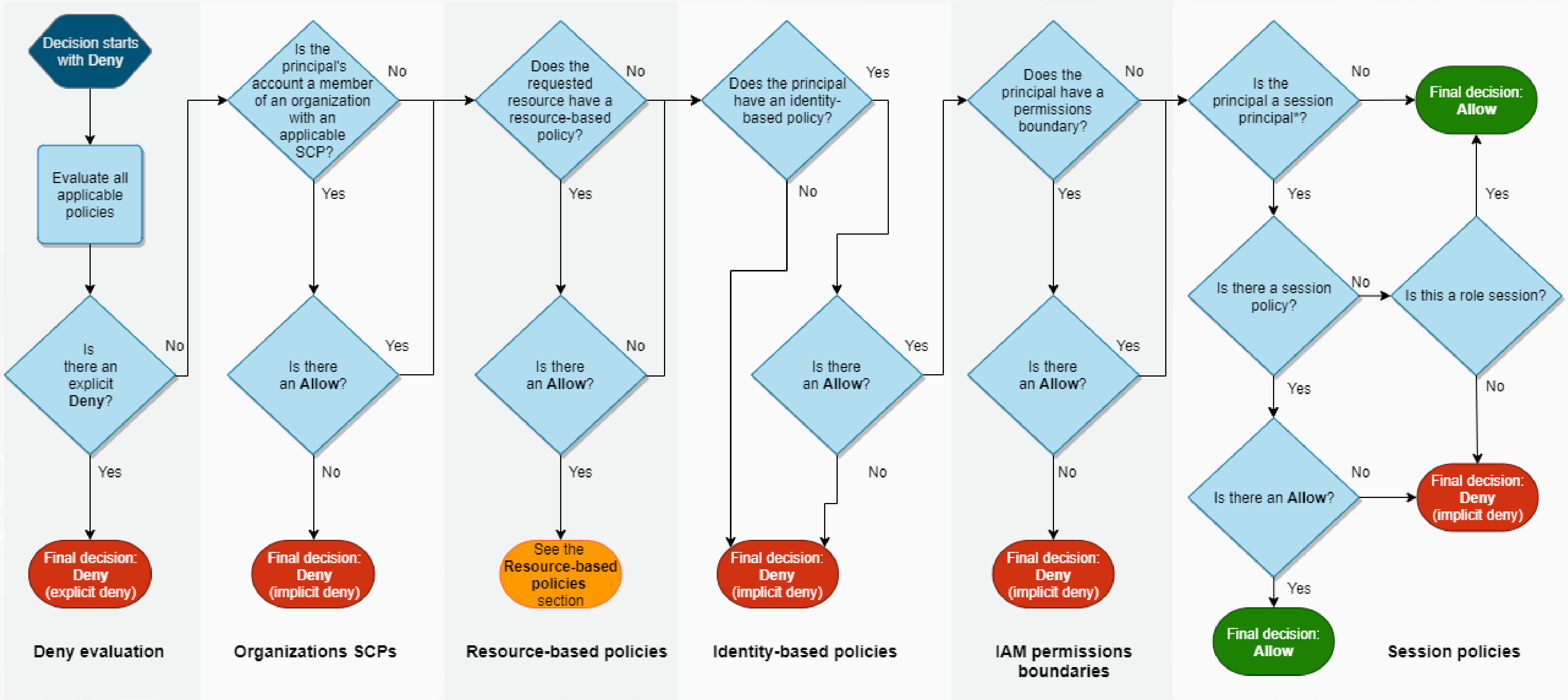
Identity-based policies with Organization SCPs



Resulting permissions are the intersection of the identity policy and the SCP

Note: an explicit deny always overrides any allow

Policy Evaluation Workflow



Decision starts with Deny

Evaluate all applicable policies

Is there an explicit Deny?

Final decision: Deny (explicit deny)

Deny evaluation

Is the principal's account a member of an organization with an applicable SCP?

Is there an Allow?

Final decision: Deny (implicit deny)

Organizations SCPs

Does the requested resource have a resource-based policy?

Is there an Allow?

See the Resource-based policies section

Resource-based policies

Does the principal have an identity-based policy?

Is there an Allow?

Final decision: Deny (implicit deny)

Identity-based policies

Does the principal have a permissions boundary?

Is there an Allow?

Final decision: Deny (implicit deny)

IAM permissions boundaries

Is the principal a session principal*?

Is there a session policy?

Is this a role session?

Is there an Allow?

Final decision: Allow

Final decision: Deny (implicit deny)

Final decision: Allow

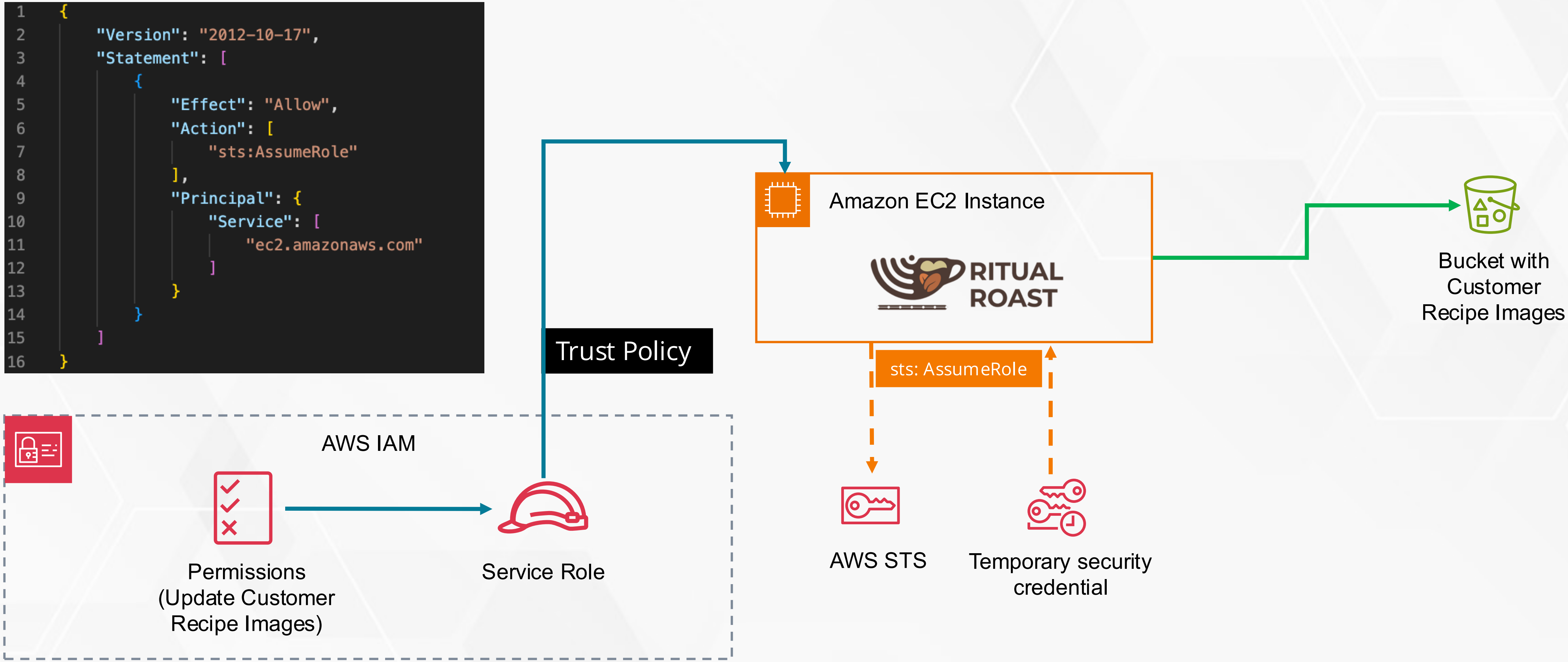
Session policies

*A session principal is either a role session or an IAM federated user session.

Types of IAM Roles

Service, Service-Linked and IAM-PassRole

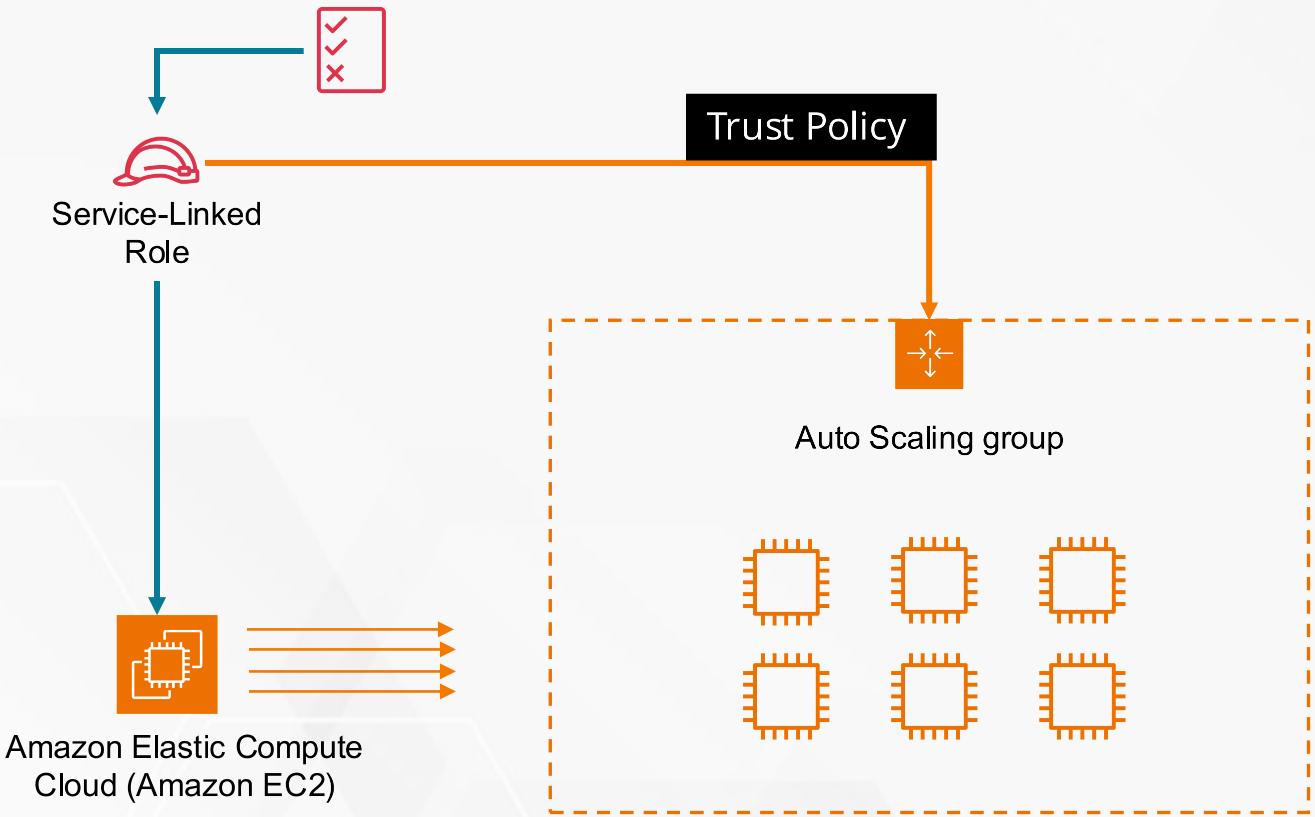
Service Roles



```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sts:AssumeRole"
            ],
            "Principal": {
                "Service": [
                    "ec2.amazonaws.com"
                ]
            }
        }
    ]
}
```

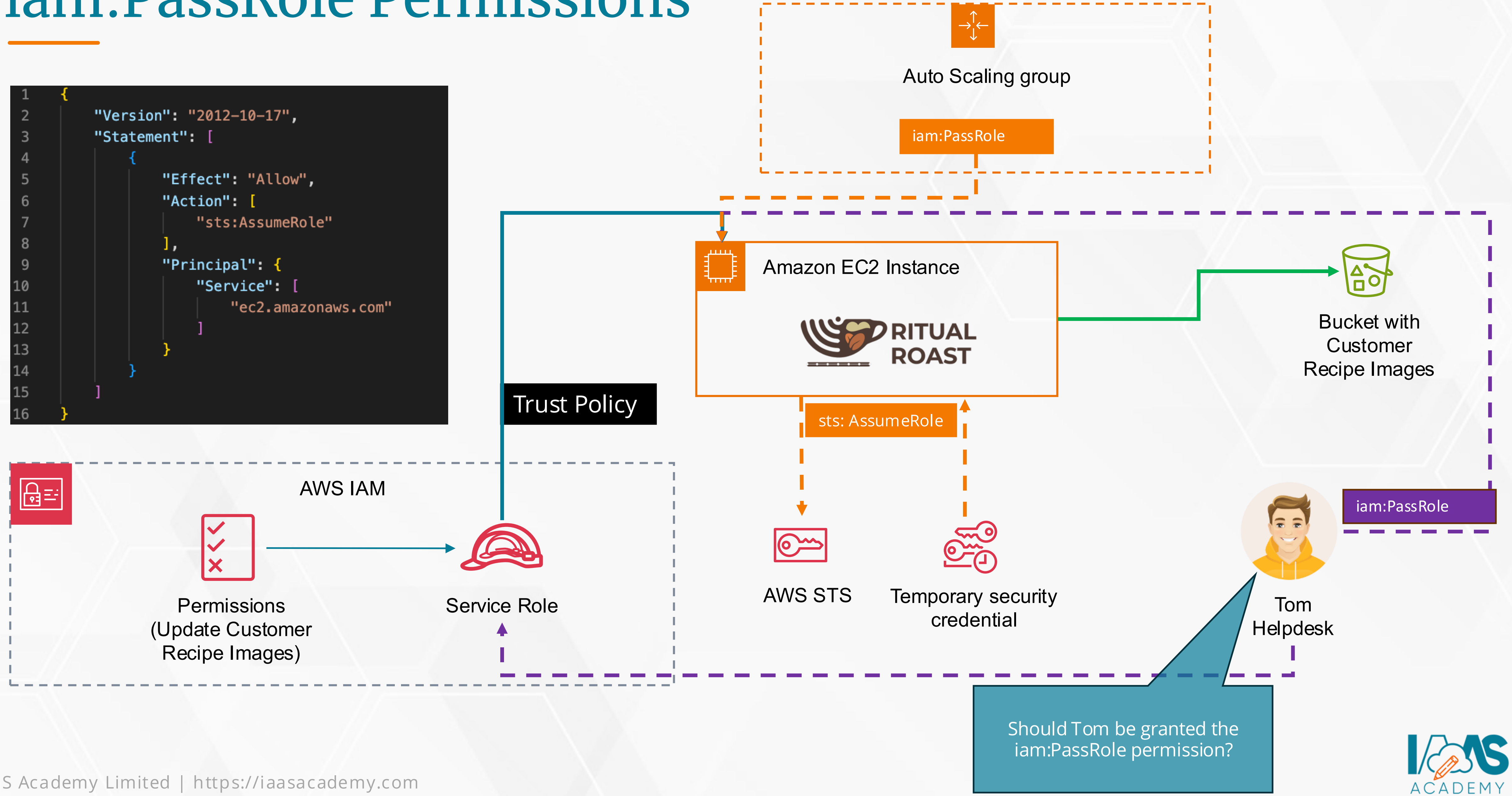
Trust Policy
Amazon EC2 Instance
RITUAL ROAST
Bucket with Customer Recipe Images
sts: AssumeRole
AWS IAM
Permissions (Update Customer Recipe Images)
Service Role
AWS STS
Temporary security credential

Service-linked Roles



- IAM role linked directly to a service and owned by the service
- Predefined with permissions required to call other AWS services on your behalf
- An IAM administrator can view, but not edit the permissions for service-linked roles.
- The service defines how to create, modify and delete the role
- Service may automatically create the role or might allow you to create, modify or delete the role
- Unlike service IAM roles, you do not have to manually configure all the permissions

iam:PassRole Permissions



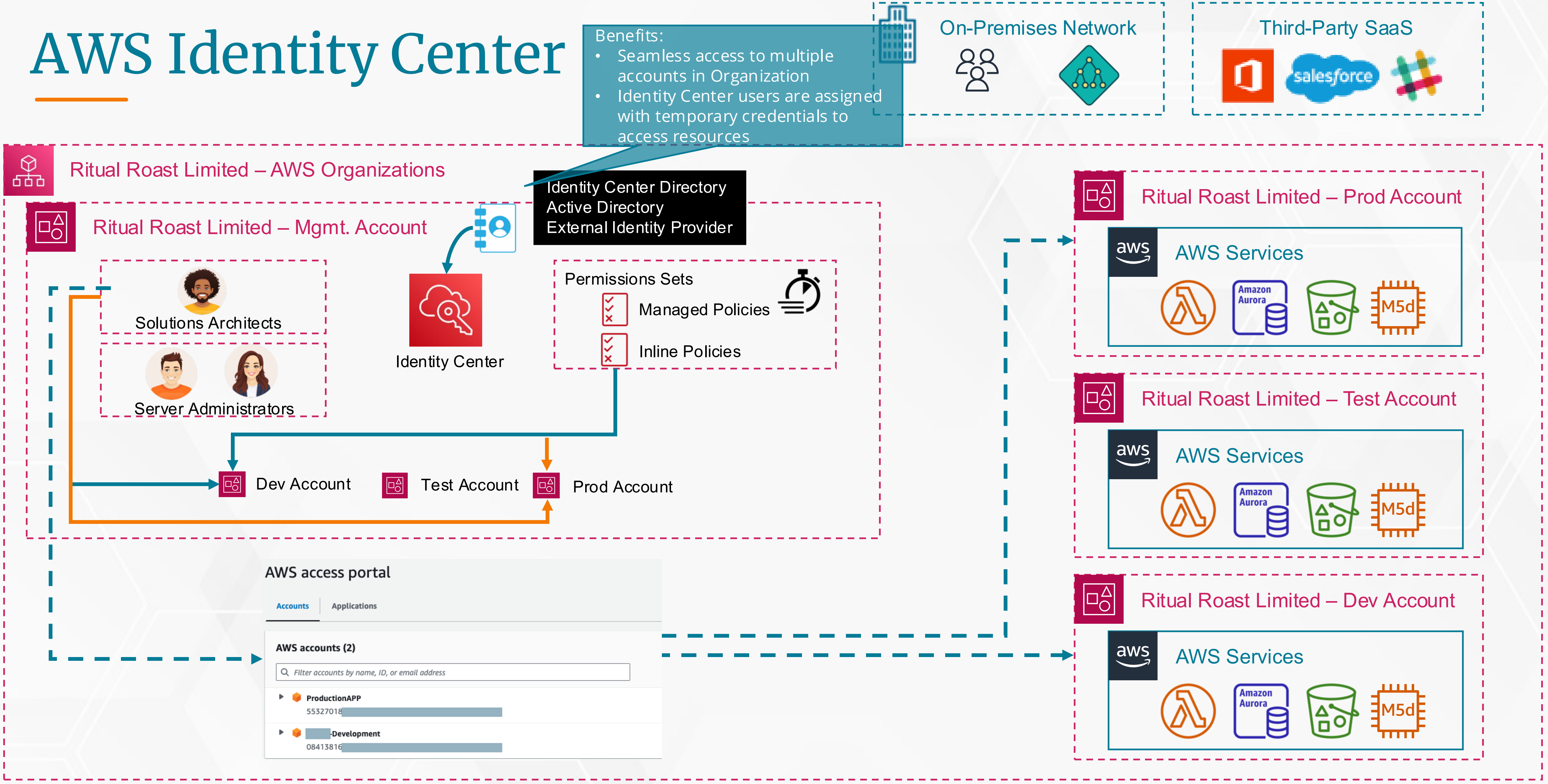
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sts:AssumeRole"
            ],
            "Principal": {
                "Service": [
                    "ec2.amazonaws.com"
                ]
            }
        }
    ]
}
```

Should Tom be granted the iam:PassRole permission?

Deploy AWS Identity Center

Implementing Workforce Identities on AWS

AWS Identity Center



Benefits:
- Seamless access to multiple accounts in Organization
- Identity Center users are assigned with temporary credentials to access resources

On-Premises Network
Third-Party SaaS
salesforce
Ritual Roast Limited – AWS Organizations
Ritual Roast Limited – Mgmt. Account
Identity Center Directory
Active Directory
External Identity Provider
Solutions Architects
Server Administrators
Identity Center
Permissions Sets
Managed Policies
Inline Policies
Dev Account
Test Account
Prod Account
Ritual Roast Limited – Prod Account
Ritual Roast Limited – Test Account
Ritual Roast Limited – Dev Account
aws
AWS Services
Amazon Aurora
M5d

AWS access portal

Accounts | Applications

AWS accounts (2)

Filter accounts by name, ID, or email address

▸ ProductionAPP
55327018[redacted]

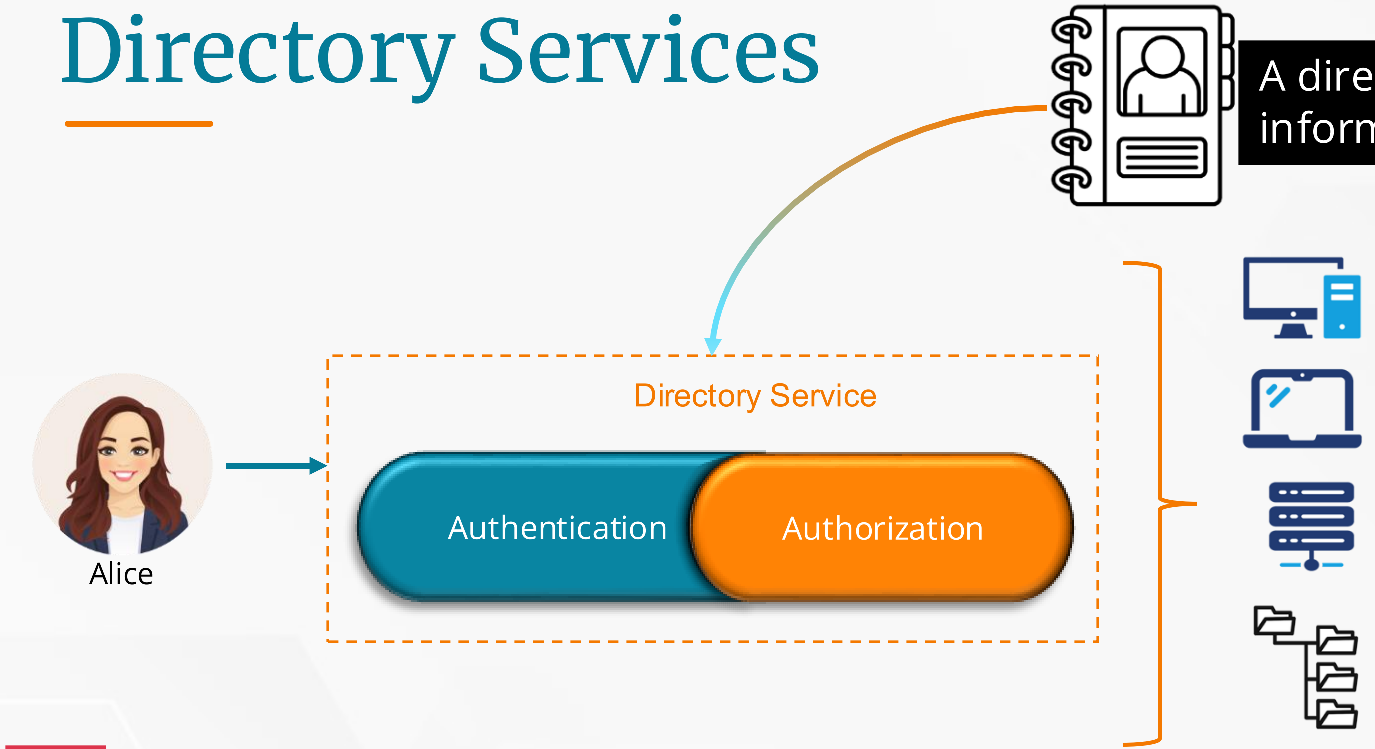
▸ [redacted]-Development
08413816[redacted]

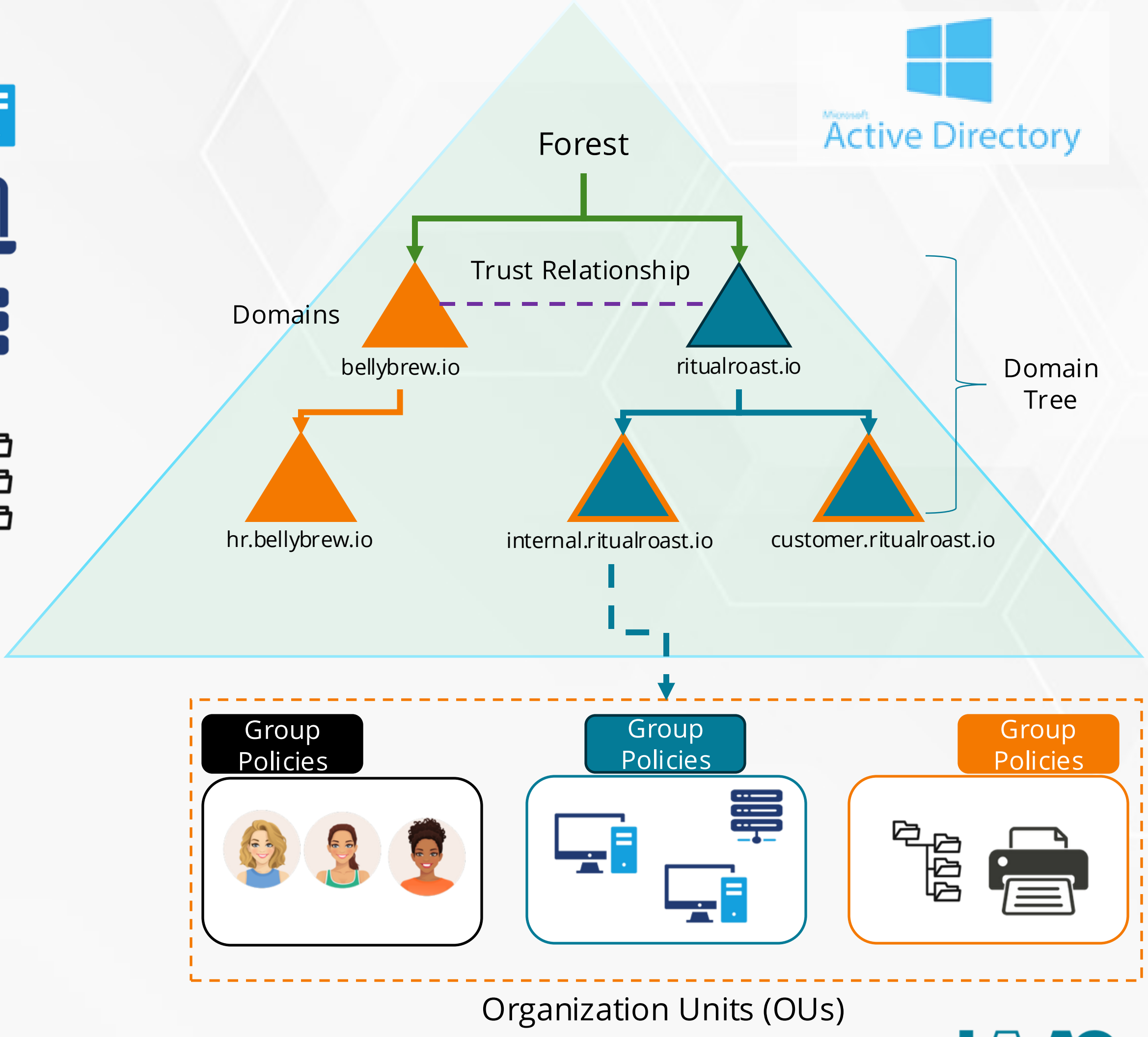
AWS Directory Services

Active Directory on AWS

Directory Services

A directory service is a database that stores and manages information about users and resources on a network.

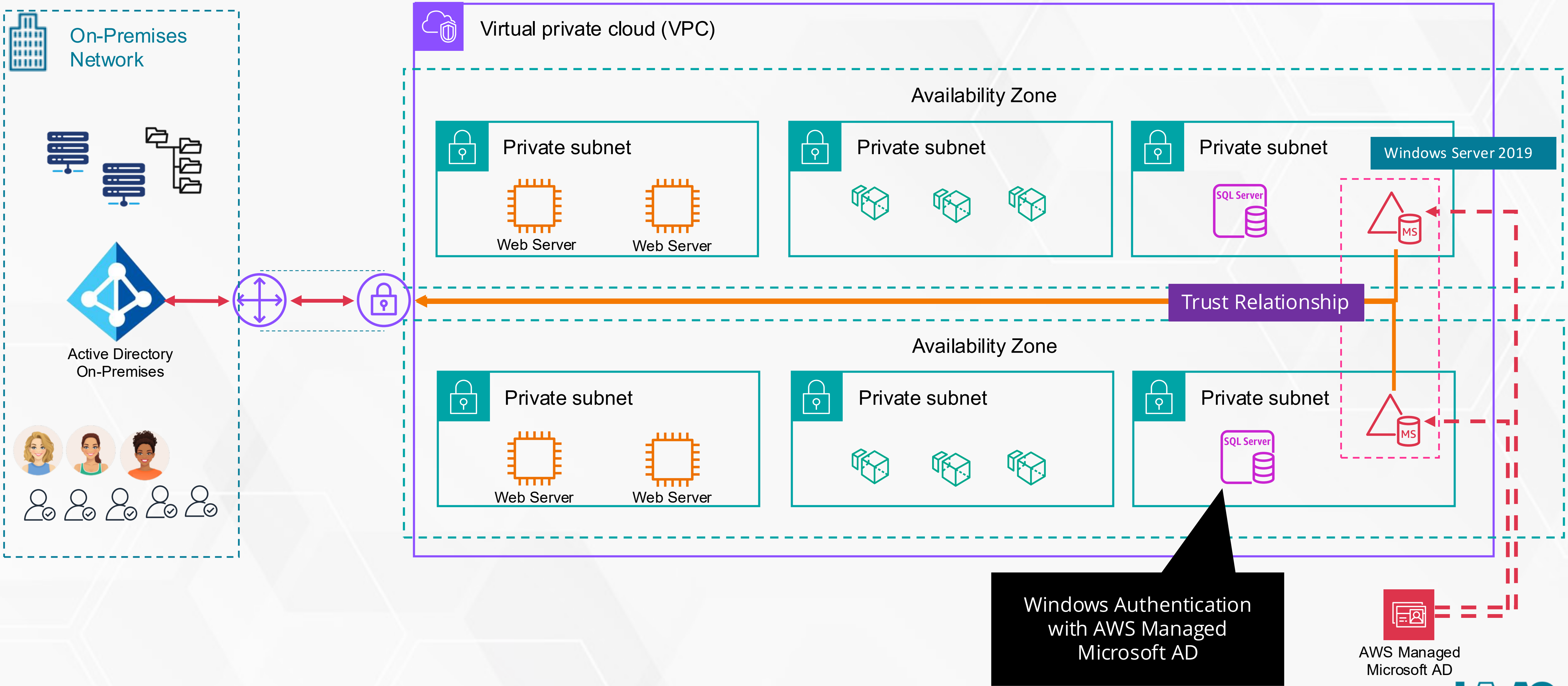




- AWS Identity and Access Management
- AWS IAM Identity Center
- AWS Directory Service

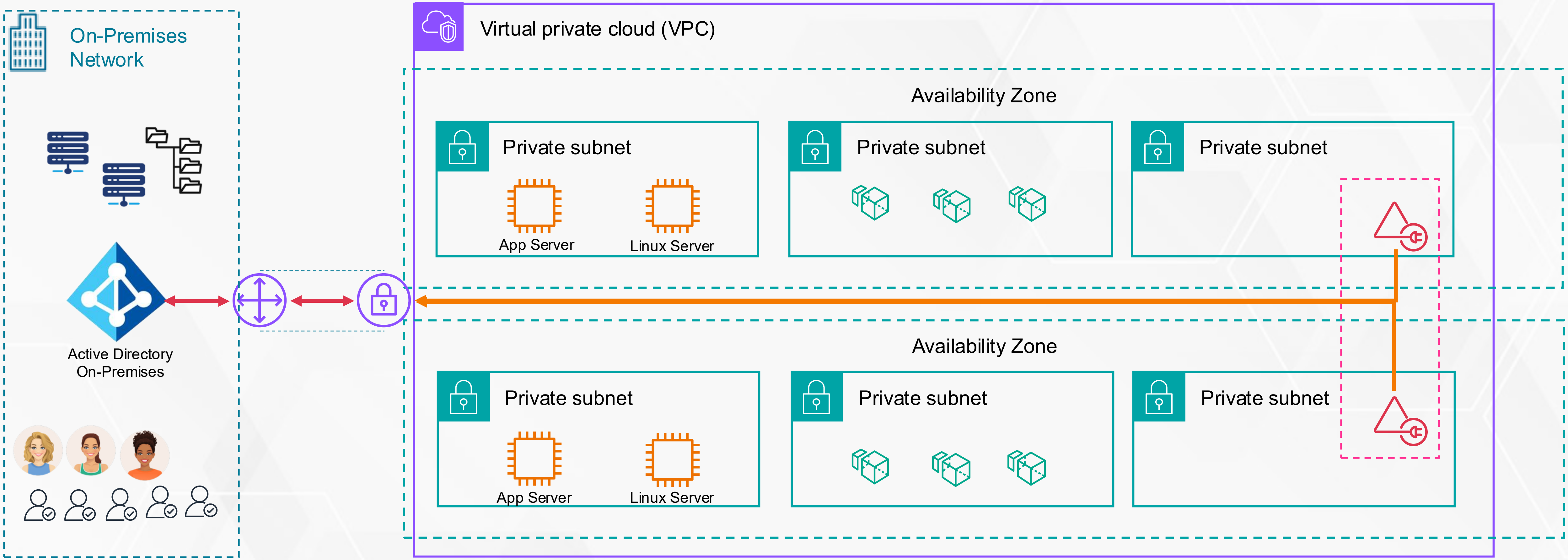
Microsoft Applications | Amazon WorkSpaces | Amazon WorkSpaces | Amazon FSx | Amazon WorkDocs

AWS Directory Services - AWS Managed Microsoft AD

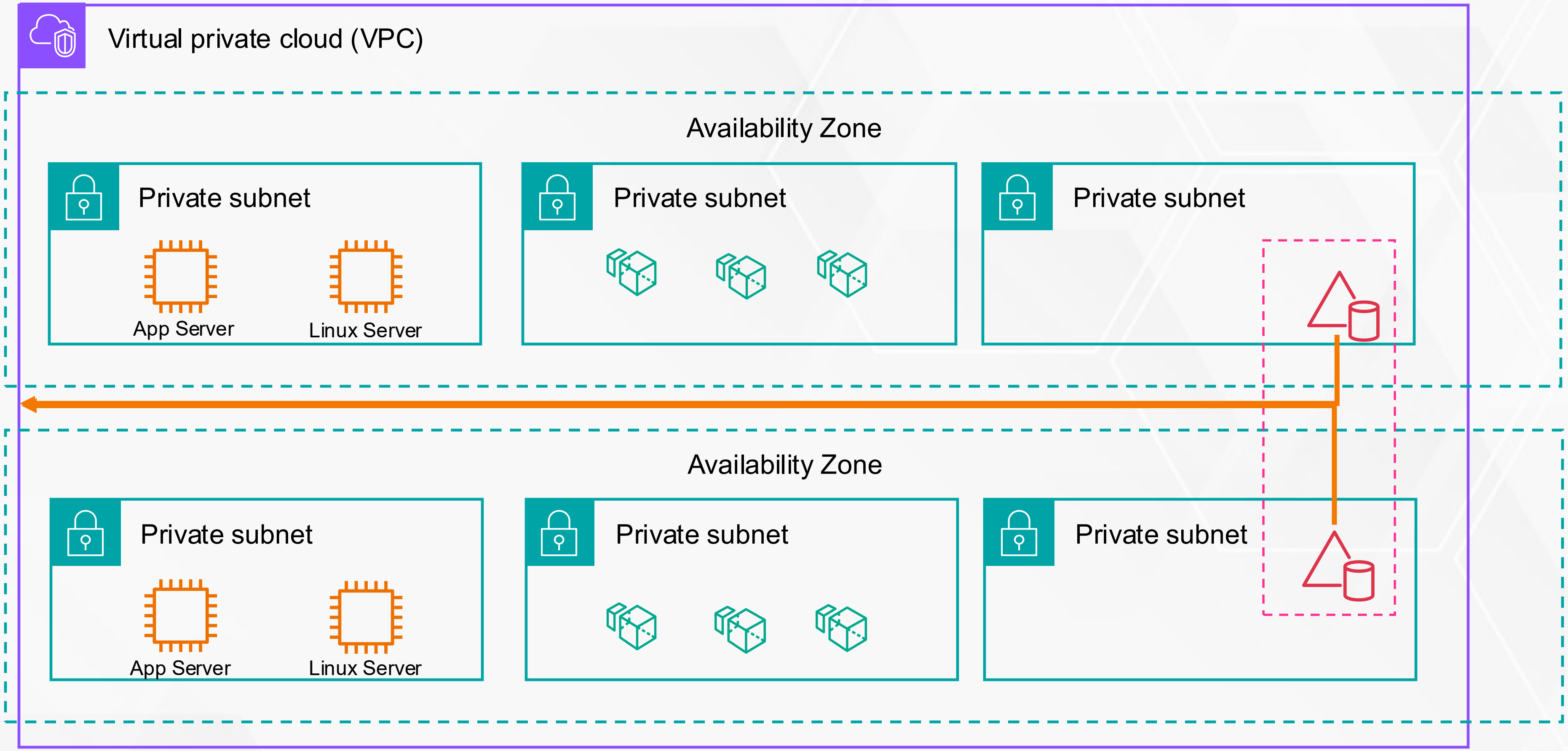


Windows Authentication with AWS Managed Microsoft AD

AWS Directory Services – AD Connector



AWS Directory Services – Simple AD



AWS Security Tools

Credentials Report & Access Advisor