**Cross-Account Role Assumption in AWS IAM: Step-by-Step Guide**

**Prerequisites:**

* **Account A**: The source AWS account (where the user/role initiating the request resides).
* **Account B**: The target AWS account (where the resources and the role to assume are located).
* Ensure you have sufficient IAM permissions to create roles and policies in both accounts.

**Step 1: Create a Role in the Target Account (Account B)**

1. **Sign in to Account B**:
   * Use the AWS Management Console with admin permissions.
2. **Navigate to IAM Roles**:
   * Go to the **IAM** service in the AWS Management Console.
   * Select **Roles** from the left-hand menu.
3. **Create a New Role**:
   * Click **Create role**.
   * In the **Trusted entity type**, select **Another AWS account**.
   * Enter the **Account ID** of Account A (the source account).
4. **Add Permissions**:
   * Attach policies that define what the assumed role can access in Account B (e.g., AmazonS3FullAccess for S3 resources).
   * You can also create a custom policy if needed.
5. **Review and Create Role**:
   * Provide a meaningful **role name** (e.g., CrossAccountRole).
   * Click **Create role**.

**Step 2: Update Trust Policy for the Role in Account B**

1. **Edit the Role's Trust Policy**:
   * Open the newly created role in Account B.
   * Go to the **Trust relationships** tab and click **Edit trust policy**.
2. **Specify Source Account or Role**: Use the following trust policy as a template:

json

Copy code

{

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Principal": {

"AWS": "arn:aws:iam::<AccountA-ID>:root"

},

"Action": "sts:AssumeRole",

"Condition": {}

}

]

}

* + Replace <AccountA-ID> with the AWS account ID of Account A.

1. **Save the Trust Policy**:
   * Click **Update policy**.

**Step 3: Grant Permissions in the Source Account (Account A)**

1. **Create or Update an IAM User/Role in Account A**:
   * If not already done, create an IAM user or role that will assume the role in Account B.
2. **Attach Inline Policy to IAM User/Role**:
   * Attach the following policy to the user/role in Account A:

json

Copy code

{

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Action": "sts:AssumeRole",

"Resource": "arn:aws:iam::<AccountB-ID>:role/<RoleName>"

}

]

}

* + - Replace <AccountB-ID> with the account ID of Account B.
    - Replace <RoleName> with the name of the role created in Account B.

**Step 4: Assume the Role**

1. **Using the AWS CLI**:
   * Use the following command to assume the role:

bash

Copy code

aws sts assume-role \

--role-arn "arn:aws:iam::<AccountB-ID>:role/<RoleName>" \

--role-session-name "SessionName"

Replace <AccountB-ID> and <RoleName> with appropriate values.

* + The command will return temporary security credentials (AccessKeyId, SecretAccessKey, and SessionToken).

1. **Set the Temporary Credentials**:
   * Configure the credentials in your environment:

bash

Copy code

export AWS\_ACCESS\_KEY\_ID=<AccessKeyId>

export AWS\_SECRET\_ACCESS\_KEY=<SecretAccessKey>

export AWS\_SESSION\_TOKEN=<SessionToken>

1. **Perform Operations in Account B**:
   * Use the assumed role's temporary credentials to interact with resources in Account B.