Lab 04: IAM Users, Groups & Policies

Lab Overview

AWS IAM is the security backbone of every AWS account. In this lab you will create IAM users and groups, write a custom least-privilege JSON policy scoped to a single resource, configure a service role, and verify permissions by signing in as the restricted user.

Service	Purpose	Free Tier
AWS IAM	Identity and Access Management — users, groups, roles, and policies	Always Free
Amazon S3	Target resource used to test and verify IAM policy permissions	Free Tier
AWS STS	Security Token Service — issues temporary credentials for roles	Always Free

■ **NOTE:** IAM is completely free. There are no charges for creating users, groups, roles, or policies.

Key Concepts

Concept	Definition	Exam Weight
IAM User	A permanent identity with a username, password, and optional access keys.	High
IAM Group	A collection of users. Policies on a group are inherited by all members.	High
IAM Policy	A JSON document defining Allow or Deny for actions on resources.	Very High
IAM Role	An identity without a password. Assumed temporarily by services or users.	Very High
Least Privilege	Grant only the minimum permissions required for a task.	Very High
MFA	Multi-Factor Authentication — second verification step for sign-in.	High

1 AWS IAM Console
Enable MFA on the Root Account

Before anything else, secure the root account with MFA. The root account has unrestricted access to everything.

1. Sign in with root credentials → search for IAM → click IAM
2. On the IAM Dashboard find Security recommendations → click Add MFA for root user
3. Select Authenticator app → scan the QR code with Google Authenticator or Authy
4. Enter two consecutive 6-digit codes to verify → click Add MFA

✓ **TIP:** After enabling MFA the Security recommendations warning disappears — this confirms the root account is secured.

2 AWS IAM — Users
Create Two IAM Users

Create lab-developer

1. Left sidebar → Users → Create user
2. Username: lab-developer
3. Check Provide user access to the AWS Management Console
4. Select I want to create an IAM user → set a custom password
5. Uncheck Users must create a new password at next sign-in
6. Click Next → Next (skip permissions for now) → Create user
7. Save the Console sign-in URL shown on the confirmation page

Create lab-readonly

1. Create user again with username: lab-readonly — same settings

■ **NOTE:** Both users have ZERO permissions. Adding them to groups in Step 3 grants access.

3 AWS IAM — Groups
Create IAM Groups and Assign Users

Create Developers group

1. Left sidebar → User groups → Create group
2. Group name: Developers → add lab-developer as member
3. Attach AmazonS3ReadOnlyAccess → Create user group

Create ReadOnly group

1. Create group: ReadOnly → add lab-readonly
2. Attach ReadOnlyAccess (AWS managed policy) → Create user group

✓ **TIP:** Attaching policies to groups means every member inherits those permissions. This is the recommended approach over attaching policies to individual users.

4 AWS IAM — Policies
Write a Custom Least-Privilege Policy

Replace YOUR-BUCKET-NAME with your Lab 01 S3 bucket name.

```
{
  "Version": "2012-10-17",
  "Statement": [
    { "Sid": "ListSpecificBucket",
      "Effect": "Allow",
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::YOUR-BUCKET-NAME" },
    { "Sid": "ReadObjectsInBucket",
      "Effect": "Allow",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::YOUR-BUCKET-NAME/*" }
  ]
}
```

1. IAM → Policies → Create policy → JSON tab → paste policy above
2. Policy name: S3-ReadOnly-SpecificBucket → Create policy
3. User groups → Developers → Permissions → Attach policies → add S3-ReadOnly-SpecificBucket
4. Remove the broad AmazonS3ReadOnlyAccess policy you attached in Step 3

■ **NOTE:** The custom policy grants access to ONE specific bucket only. The user receives Access Denied on any other bucket.

5 AWS IAM — Roles
Create a Service Role for EC2

1. Left sidebar → Roles → Create role
2. Trusted entity: AWS service → Use case: EC2 → Next
3. Attach AmazonS3ReadOnlyAccess → Next
4. Role name: EC2-S3-ReadOnly-Role → Create role

✓ **TIP:** When this role is attached to an EC2 instance, code running on that instance can call S3 read APIs without any access keys embedded in the code.

6 Web Browser
Test Permissions as the Developer User

1. Open an incognito browser window
2. Navigate to the IAM Console sign-in URL you saved in Step 2
3. Sign in as lab-developer with the password you set
4. Navigate to S3 — you can browse the specific bucket
5. Click a different bucket — Access Denied confirms least privilege is working

■ **NOTE:** Always test permissions from the restricted user's perspective. Never assume a policy is correct without verification.

Verification Checklist

- ■ MFA enabled on the root account
- ■ lab-developer and lab-readonly users created with console access
- ■ Developers and ReadOnly groups created with users assigned
- ■ Custom policy S3-ReadOnly-SpecificBucket created with correct bucket ARN
- ■ Custom policy attached to Developers; broad managed policy removed
- ■ EC2-S3-ReadOnly-Role created with EC2 as trusted entity
- ■ lab-developer confirmed: access to specific bucket only, Access Denied elsewhere

What You Learned

- IAM Users, Groups, and Roles — when to use each identity type
- Least Privilege Principle — grant only the minimum permissions required
- JSON Policy Structure — Effect, Action, Resource — anatomy of every IAM policy
- Managed vs Custom Policies — pre-built convenience vs custom precision
- Service Roles — how AWS services call other services without hardcoded credentials
- MFA — multi-factor authentication as the first layer of account security

Lab Cleanup

✗ **IMPORTANT:** Unused IAM users with credentials are a security risk. Clean up when finished.

#	Resource	How to Delete
1	Custom Policy	IAM → Policies → filter Customer managed → S3-ReadOnly-SpecificBucket → Delete
2	Developers Group	IAM → User groups → Developers → Delete group
3	ReadOnly Group	IAM → User groups → ReadOnly → Delete group
4	lab-developer	IAM → Users → select lab-developer → Delete
5	lab-readonly	IAM → Users → select lab-readonly → Delete
6	EC2 Role	IAM → Roles → EC2-S3-ReadOnly-Role → Delete

■ **NOTE:** MFA on the root account should remain active permanently — do not remove it.