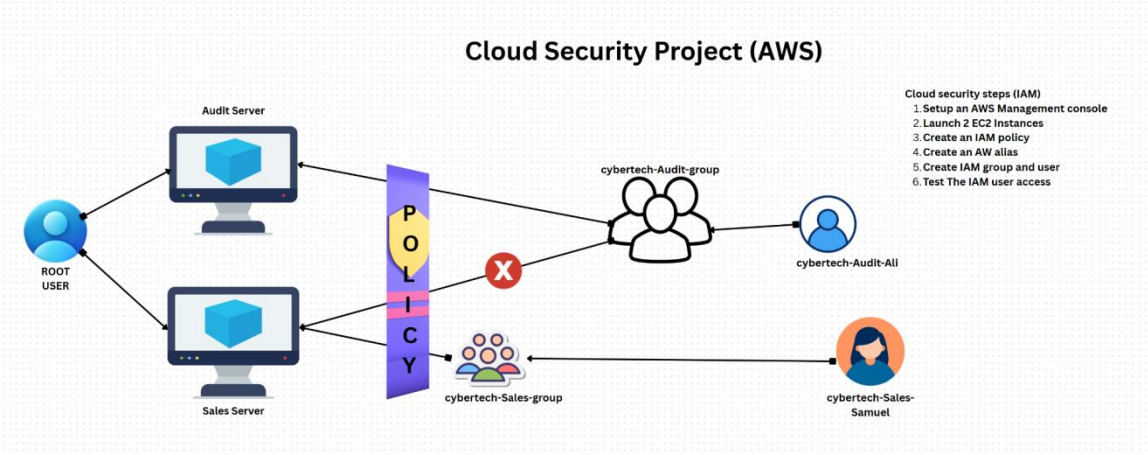
AWS IAM Cloud Security Project

1. Project Overview

I completed this project on cloud security controls in Amazon Web Services (AWS), focusing on Identity and Access Management (IAM). The goal was to create a least-privilege policy, attach it to a user group, and verify that the policy correctly restricts actions on two Amazon EC2 instances (audit and sales).

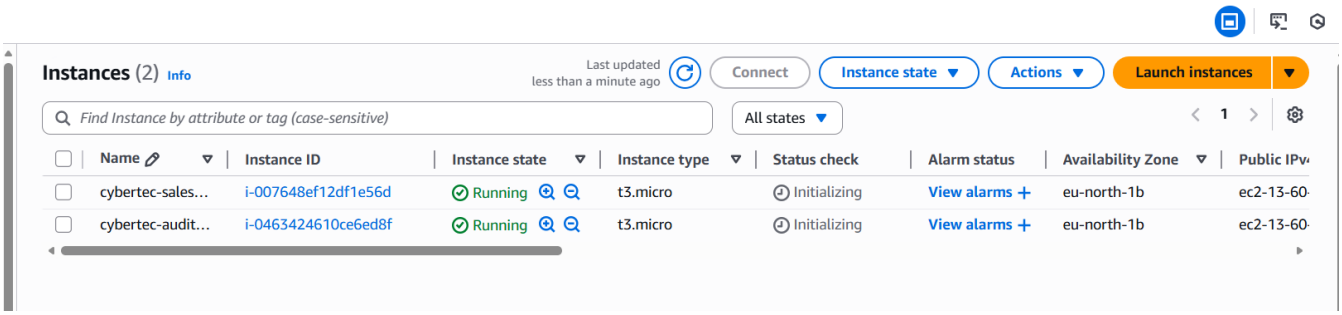


2. Tools & Concepts

- AWS IAM – users, groups, policies, account alias
- Amazon EC2 – instance tagging and lifecycle actions
- JSON policy syntax – Effect, Action, Resource
- Principle of least privilege and policy testing

3. Tagging Strategy

I applied a descriptive tag to each EC2 instance: Instance | Tag Key | Tag Value audit | Environment | Audit sales | Environment | Sales



4. Creating the IAM Policy

I created an IAM policy in JSON format that explicitly prevents stop and start operations on the audit server, while permitting those same actions on the sales server. This policy enforces operational restrictions by using instance-level conditions, ensuring the audit server remains continuously available while allowing controlled management of the sales server.

Permissions defined in this policy Info

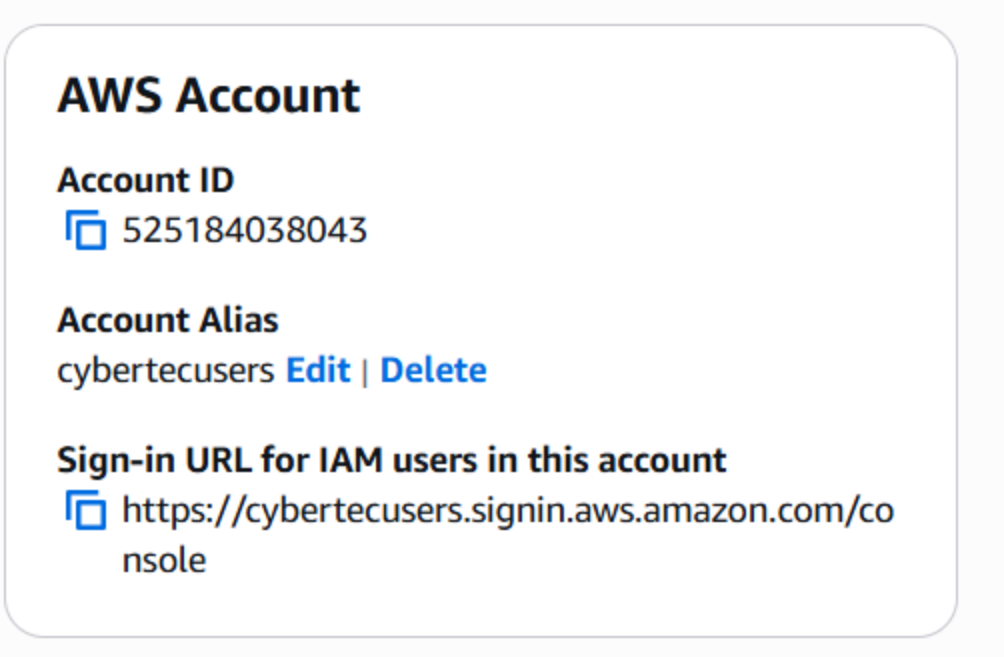
Copy | Edit | Summary | JSON

Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity (user, user group, or role), attach a policy to it

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:*",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "ec2:ResourceTag/Env": "Audit"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:Describe*",
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "ec2:DeleteTags",
                "ec2:CreateTags"
            ],
            "Resource": "*"
        }
    ]
}
```

5. Account Alias

I set a memorable account alias to replace the default numeric URL, making sign-in easier for team members.

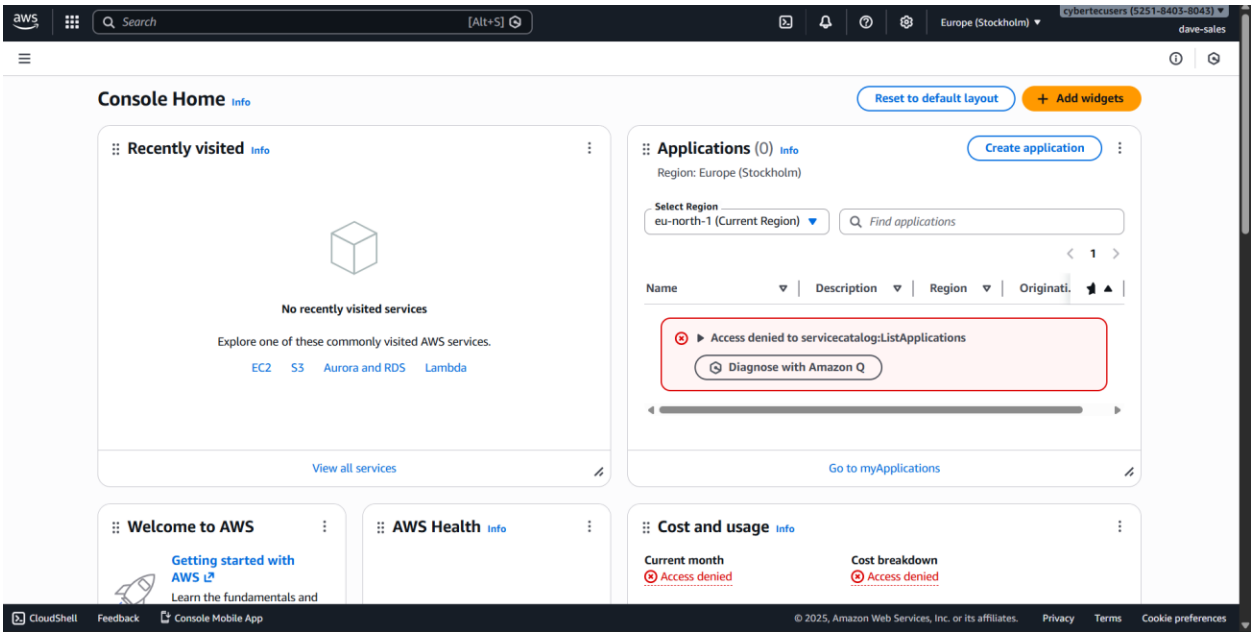


6. IAM Users & Groups

1. Created an IAM user group called Developers.
2. Attached the **CybertechAuditEnvPolicy** policy to the group.
3. Added individual IAM users who require controlled EC2 access.

7. Logging in as an IAM User

IAM users can sign in through:
- AWS Management Console (using the new alias URL)
- AWS CLI via programmatic keys



8. Testing the Policy

Test Action | Expected Result | Actual Result

Stop audit instance | Denied | Access denied error displayed

Stop sales instance | Allowed | Instance stopped successfully

Start audit instance | Denied | Access denied error displayed

Start sales instance | Allowed | Instance started successfully