Cloud Security Privilege Escalation Simulation Report | Oluwaseun Quadri

1. Objective

The objective of this lab was to simulate an IAM misconfiguration in AWS, perform privilege escalation using the AWS CLI, and remediate the vulnerability using least privilege principles.

The lab environment was built using services from Amazon Web Services.

2. Environment Setup

- Cloud Provider: AWS
- Service Used: IAM (Identity and Access Management)
- Tool Used: AWS CLI
- Test User: lowpriv-user
- Custom Policy: EscalationPolicy

3. Vulnerability Identification

A custom IAM policy (EscalationPolicy) was created with the following dangerous permissions:

- iam:AttachUserPolicy
- iam:CreateAccessKey
- iam:ListUsers
- iam:ListPolicies

The permission iam:AttachUserPolicy allowed the user to attach managed policies to themselves.

This violated the **principle of least privilege.**

4. Exploitation Steps

1. Configured AWS CLI with credentials for: lowpriv-user.
2. Verified identity using: **aws sts get-caller-identity**
3. Attached AWS managed policy **AdministratorAccess**

   Using: **aws iam attach-user-policy --user-name lowpriv-user --policy-arn arn:aws:iam::aws:policy/AdministratorAccess**
4. Verified escalation by successfully executing: **aws ec2 describe-instances**

   This confirmed full administrative access.

5. Impact Analysis

If exploited in a real environment, this misconfiguration could allow:

- Full AWS account takeover
- Creation or deletion of resources
- Data exfiltration
- Privilege persistence
- Financial loss due to unauthorized resource creation

This demonstrates how IAM misconfiguration is a critical cloud security risk.

6. Remediation

The vulnerability was remediated by:

1. Detaching the AdministratorAccess policy.
2. Editing the custom policy to remove:
   - **iam:AttachUserPolicy**
   - **iam:CreateAccessKey**
3. Applying least privilege principles.
4. Retesting the escalation attempt, which resulted in:
   - AccessDenied

Additionally, Multi-Factor Authentication (MFA) was enabled for the IAM user to strengthen authentication security.

7. Verification After Remediation

Attempted privilege escalation again:

```
aws iam attach-user-policy ...
```

Result:

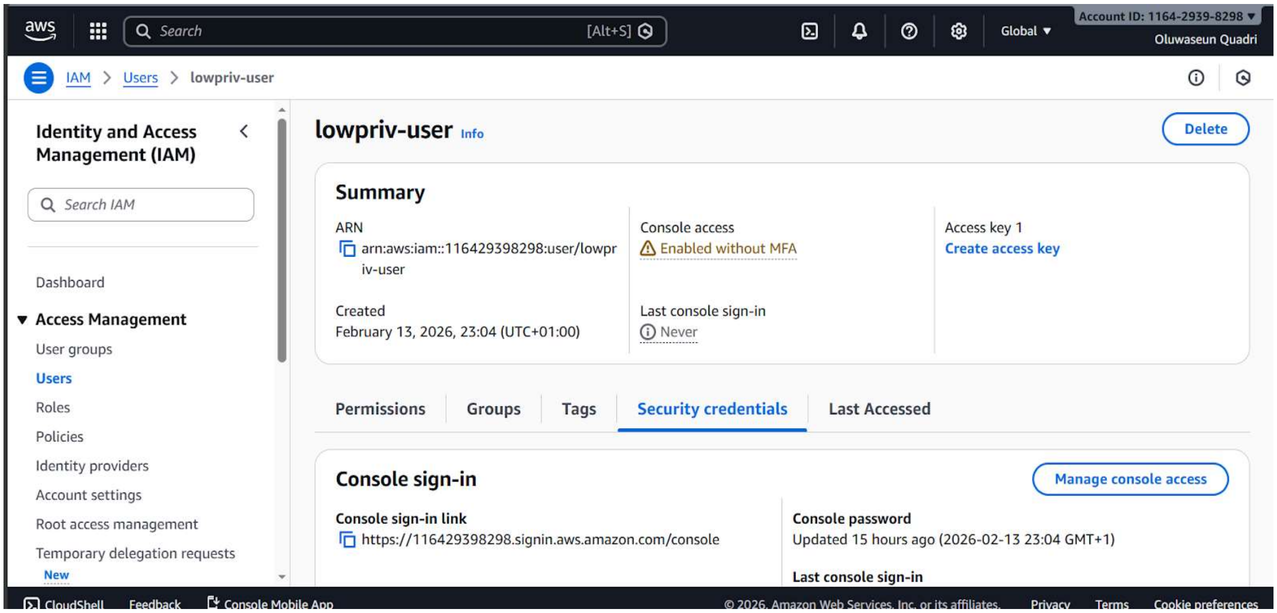
```
AccessDenied
```

Administrative EC2 actions were no longer permitted.

8. Lessons Learned

- IAM misconfigurations can lead to privilege escalation.
- The principle of least privilege must always be enforced.
- Avoid granting wildcard or policy-management permissions to non-admin users.
- MFA significantly reduces risk of credential compromise.
- Regular IAM audits are essential in cloud environments.

DELIVERABLE SCREENSHOTS



lowpriv-user

Summary

ARN
arn:aws:iam::116429398298:user/lowpriv-user

Created
February 13, 2026, 23:04 (UTC+01:00)

Console access
⚠ Enabled without MFA

Last console sign-in
ⓘ Never

Access key 1
Create access key

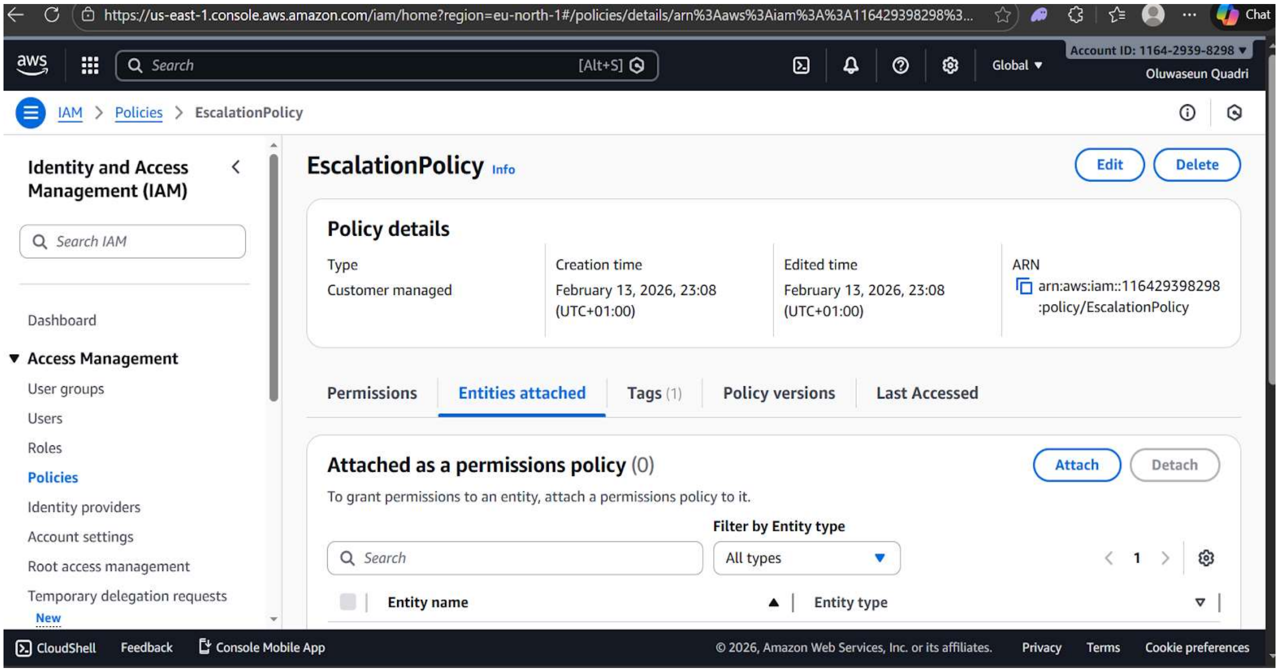
Console sign-in

Console sign-in link
https://116429398298.signin.aws.amazon.com/console

Console password
Updated 15 hours ago (2026-02-13 23:04 GMT+1)

Last console sign-in

FIG. 1 IMAGE INDICATING USER DASHBOARD



EscalationPolicy Info

Policy details

Type	Creation time	Edited time	ARN
Customer managed	February 13, 2026, 23:08 (UTC+01:00)	February 13, 2026, 23:08 (UTC+01:00)	arn:aws:iam::116429398298:policy/EscalationPolicy

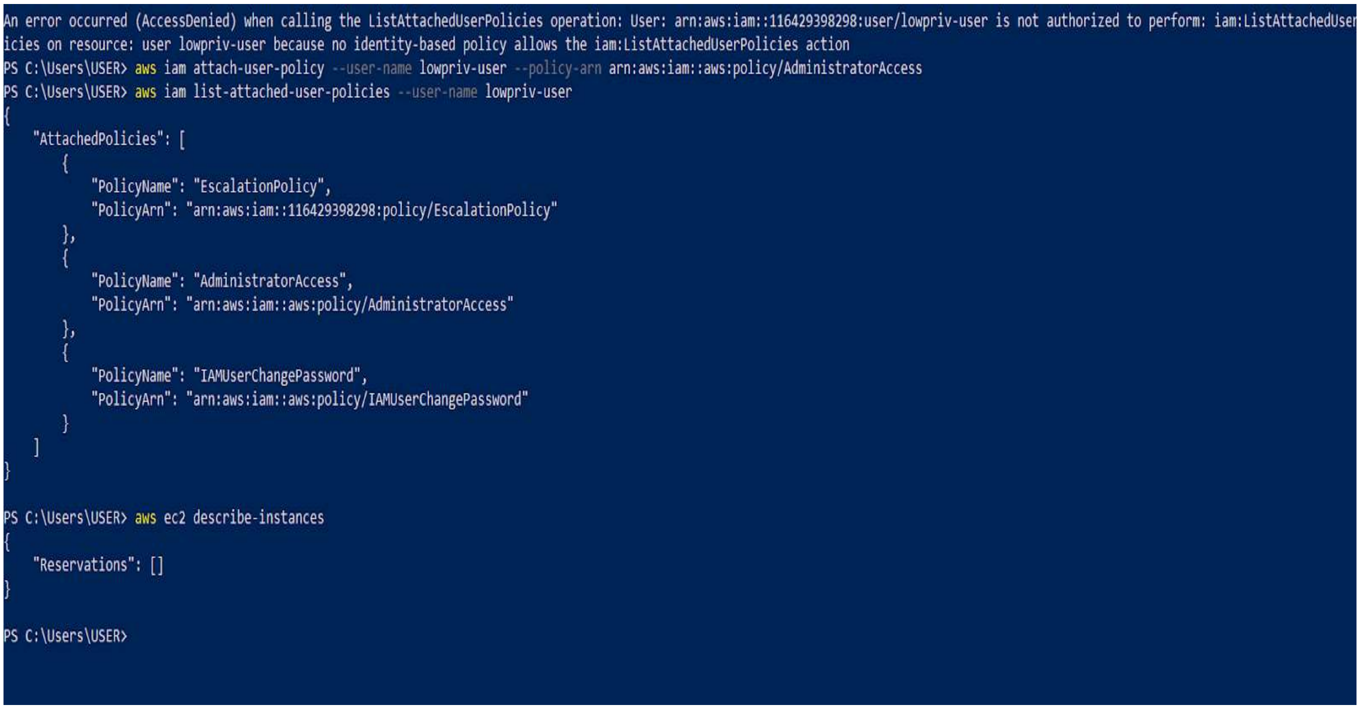
Attached as a permissions policy (0)

To grant permissions to an entity, attach a permissions policy to it.

Filter by Entity type: All types

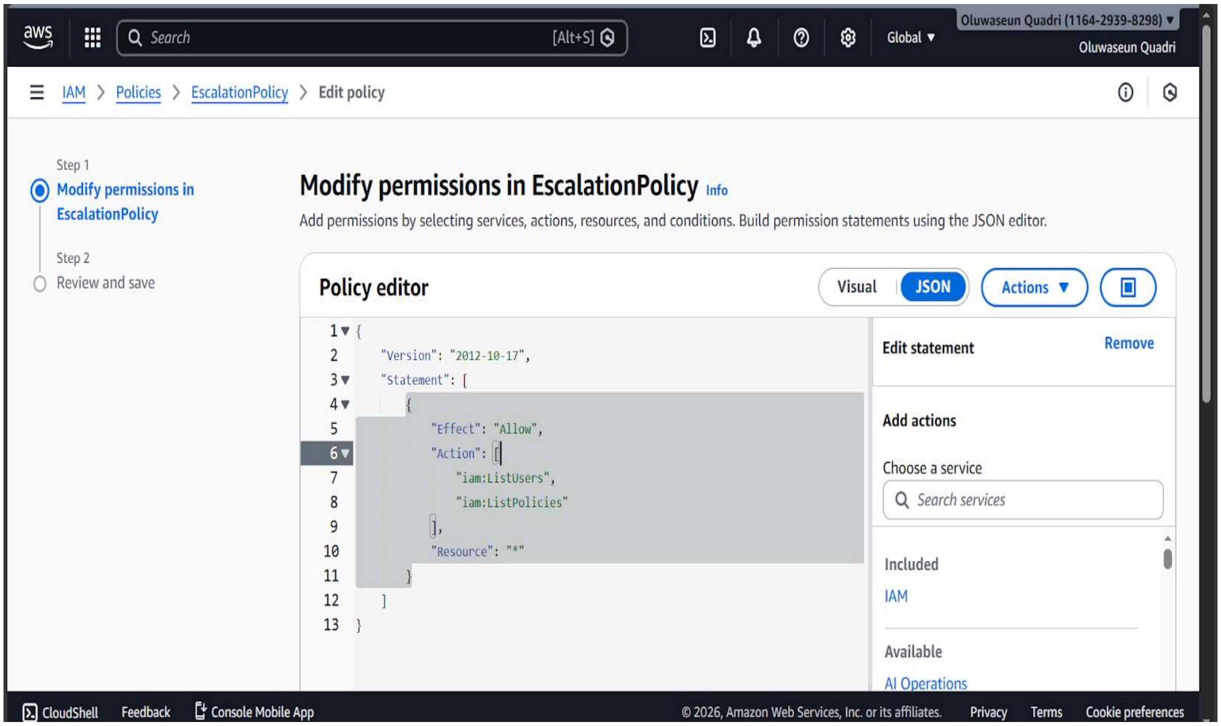
Entity name	Entity type

FIG 2. IMAGE INDICATING OF ESCALATION POLICY



```
An error occurred (AccessDenied) when calling the ListAttachedUserPolicies operation: User: arn:aws:iam::116429398298:user/lowpriv-user is not authorized to perform: iam:ListAttachedUser
icies on resource: user lowpriv-user because no identity-based policy allows the iam:ListAttachedUserPolicies action
PS C:\Users\USER> aws iam attach-user-policy --user-name lowpriv-user --policy-arn arn:aws:iam::aws:policy/AdministratorAccess
PS C:\Users\USER> aws iam list-attached-user-policies --user-name lowpriv-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "EscalationPolicy",
            "PolicyArn": "arn:aws:iam::116429398298:policy/EscalationPolicy"
        },
        {
            "PolicyName": "AdministratorAccess",
            "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"
        },
        {
            "PolicyName": "IAMUserChangePassword",
            "PolicyArn": "arn:aws:iam::aws:policy/IAMUserChangePassword"
        }
    ]
}

PS C:\Users\USER> aws ec2 describe-instances
{
    "Reservations": []
}

PS C:\Users\USER>
```

FIG 3. IMAGE SHOWING COMMAND LINE FOR ATTACHED POLICIES



aws
Search
[Alt+S]
Global ▼
Oluwaseun Quadri (1164-2939-8298) ▼
Oluwaseun Quadri
IAM > Policies > EscalationPolicy > Edit policy

Step 1

Modify permissions in EscalationPolicy

Step 2

Review and save

Modify permissions in EscalationPolicy Info

Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.

Policy editor

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iam:ListUsers",
                "iam:ListPolicies"
            ],
            "Resource": "*"
        }
    ]
}
```

Edit statement

Add actions

Choose a service

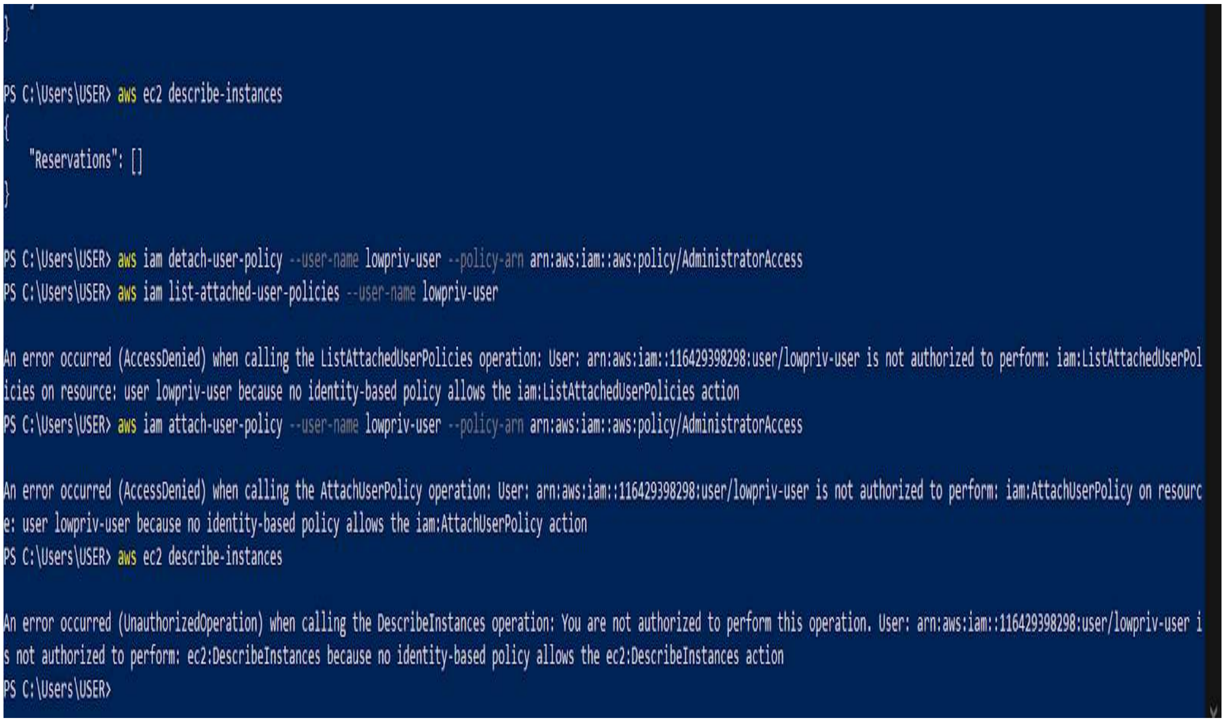
Included

IAM

Available

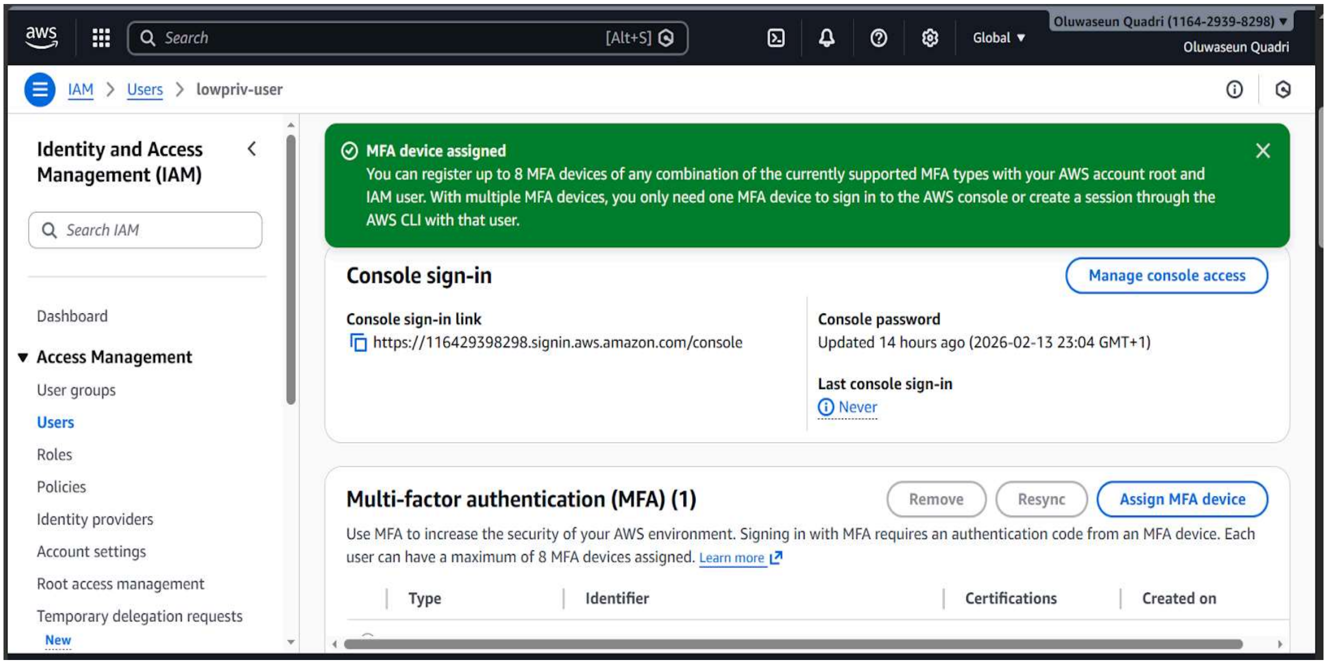
AI Operations

FIG.4 IMAGE INDICATING MODIFICATION OF PERMISSION ON AWS



```
}

PS C:\Users\USER> aws ec2 describe-instances
{
    "Reservations": []
}

PS C:\Users\USER> aws iam detach-user-policy --user-name lowpriv-user --policy-arn arn:aws:iam::aws:policy/AdministratorAccess
PS C:\Users\USER> aws iam list-attached-user-policies --user-name lowpriv-user

An error occurred (AccessDenied) when calling the ListAttachedUserPolicies operation: User: arn:aws:iam::116429398298:user/lowpriv-user is not authorized to perform: iam:ListAttachedUserPol
icies on resource: user lowpriv-user because no identity-based policy allows the iam:ListAttachedUserPolicies action
PS C:\Users\USER> aws iam attach-user-policy --user-name lowpriv-user --policy-arn arn:aws:iam::aws:policy/AdministratorAccess

An error occurred (AccessDenied) when calling the AttachUserPolicy operation: User: arn:aws:iam::116429398298:user/lowpriv-user is not authorized to perform: iam:AttachUserPolicy on resourc
e: user lowpriv-user because no identity-based policy allows the iam:AttachUserPolicy action
PS C:\Users\USER> aws ec2 describe-instances

An error occurred (UnauthorizedOperation) when calling the DescribeInstances operation: You are not authorized to perform this operation. User: arn:aws:iam::116429398298:user/lowpriv-user i
s not authorized to perform: ec2:DescribeInstances because no identity-based policy allows the ec2:DescribeInstances action
PS C:\Users\USER>
```

FIG.5 IMAGE INDICATING DENIAL OF ACCESS AFTER MODIFICATION



MFA device assigned

You can register up to 8 MFA devices of any combination of the currently supported MFA types with your AWS account root and IAM user. With multiple MFA devices, you only need one MFA device to sign in to the AWS console or create a session through the AWS CLI with that user.

Console sign-in

Console sign-in link
https://116429398298.signin.aws.amazon.com/console

Console password
Updated 14 hours ago (2026-02-13 23:04 GMT+1)

Last console sign-in
Never

Multi-factor authentication (MFA) (1)

Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. Learn more

Type	Identifier	Certifications	Created on

FIG.6 IMAGE INDICATING ENABLING OF MFA ON AWS TO FURTHER ENHANCE SECURITY