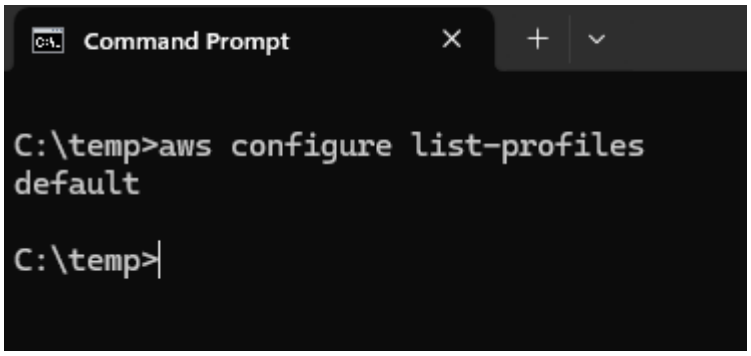
😊 Configuring AWS CLI for Multiple Accounts

The steps guide you through configuring the AWS CLI to manage multiple AWS accounts on your local machine. By setting up different profiles for each account, you can switch between them using simple commands. The process involves creating IAM users with access keys in each account, configuring those accounts in the CLI, and using specific commands to manage resources like S3 buckets and EC2 instances across the accounts. The end goal is to streamline the management of multiple AWS accounts from a single CLI setup.

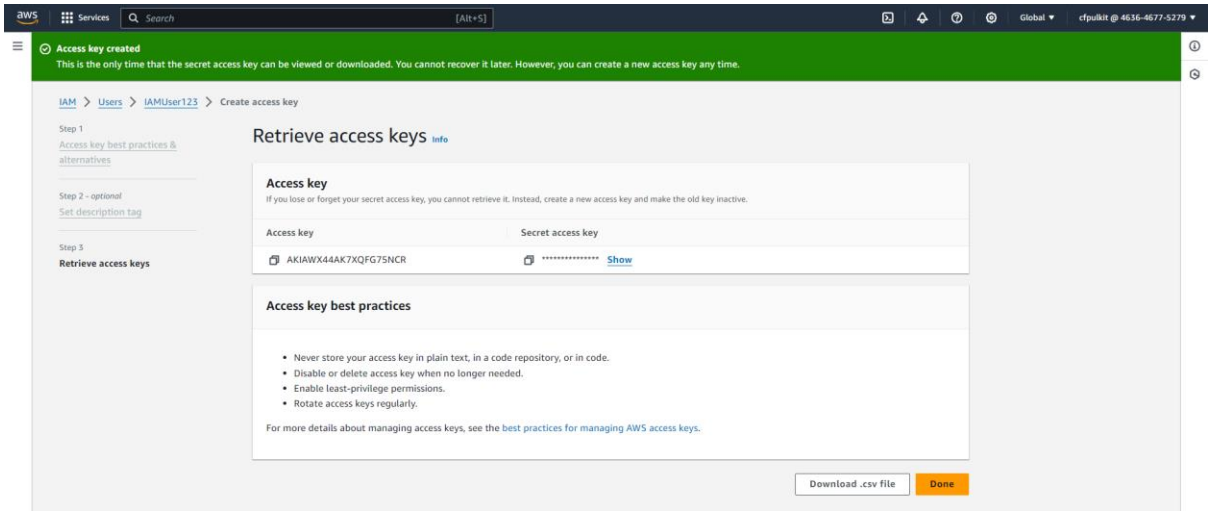
😄 To begin with the Lab:

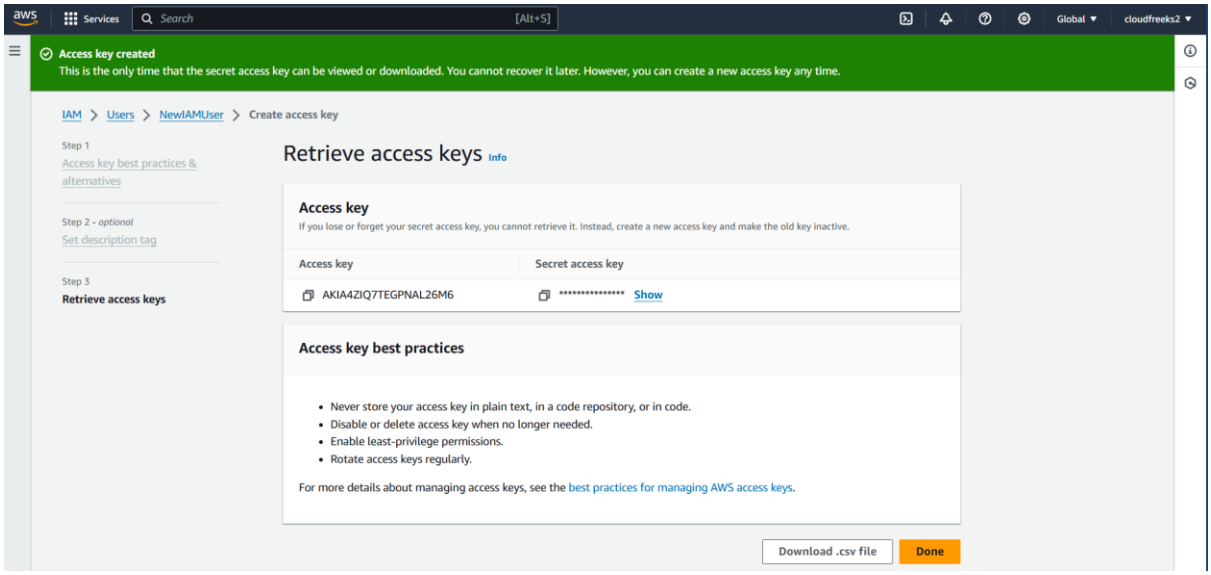
1. In this lab, we are going to configure AWS CLI for Multiple Accounts.
2. So, I believe you must have prior knowledge of AWS CLI, and you must have configured it on your local machine. Now we are going to run a command which you can see below.
3. The aws configure list-profiles command is used to list all the AWS CLI profiles that are configured on your system

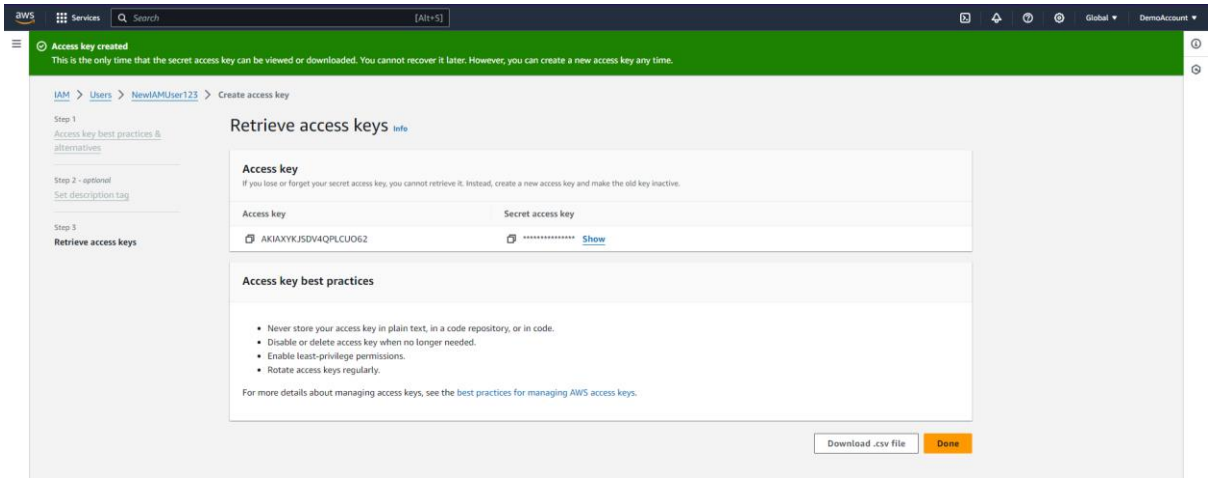
   aws configure list-profiles



4. Here you can see that a default profile has been set on your local machine. Now we are going to use different AWS Accounts, and we will create an IAM user in other accounts and create the Access and Secret access keys to configure those accounts on our local system.
5. First, we will create an IAM user in all three of our accounts with Administrator access.
6. Below you can see that I have created 3 IAM users with Access and Secret access keys in three different accounts.







7. Now we are going to configure all these accounts in our local machine using AWS CLI.
8. For that we are going to issue this command shown below.

   aws configure --profile account
9. Below you can see I have configured all three accounts in three different regions. For that, I simply run the command, and I have set the profile as account 1, account 2, and

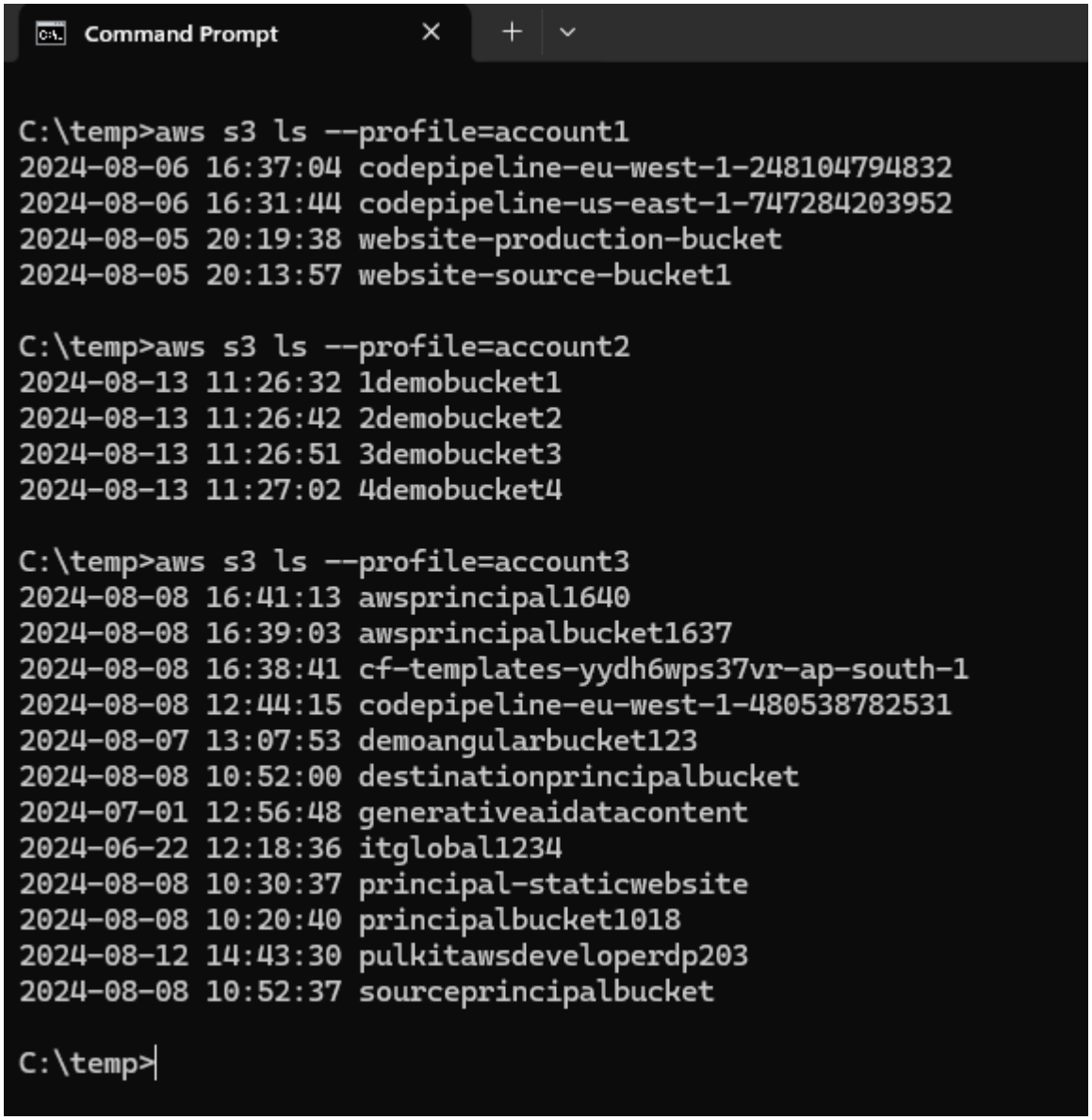
account 3 respectively. After that, I provided the Access and Secret access keys which you can see below and they are all different.

```
C:\temp>aws configure --profile account1
AWS Access Key ID [None]: AKIAXYKJSDV4QPLCUO62
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-1
Default output format [None]: json

C:\temp>aws configure --profile account2
AWS Access Key ID [None]: AKIA4ZIQ7TEGPNAL26M6
AWS Secret Access Key [None]: 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
Default region name [None]: ap-south-1
Default output format [None]: json

C:\temp>aws configure --profile account3
AWS Access Key ID [None]: AKIAWX44AK7XQFG75NCR
AWS Secret Access Key [None]: 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
Default region name [None]: ap-southeast-1
Default output format [None]: json
```

10. Now you can see that we have 4 different profiles listed below.

```
C:\temp>aws configure list-profiles
default
account1
account2
account3
```

11. Now we are going to run some commands here.
12. Below you can see that I have run the command to list S3 buckets using the different profiles as you can see below, and I can see all the buckets present in each of the accounts.



13. Now I ran the command to list the instances running in each of the accounts in the region that we have mentioned while configuring and you can see that no instance is currently running. So, what you can do is go and launch an instance in any of the account then run the same command.

    aws ec2 describe-instances --profile=account

```
C:\temp>aws ec2 describe-instances --profile=account1
{
    "Reservations": []
}


C:\temp>aws ec2 describe-instances --profile=account2
{
    "Reservations": []
}


C:\temp>aws ec2 describe-instances --profile=account3
{
    "Reservations": []
}


C:\temp>
```

14. Now the thing is every time you need to mention the profile of an account you want to see the resources for, but if you want to make an account the default account then you can use this command.
15. By using this command, you can change the default profile of your AWS CLI, and you won't see that while listing your profiles. But if you run the command to list S3 buckets then you will be able to see that your account1 is now the default account.

    set AWS_DEFAULT_PROFILE=account1

```
C:\temp>set AWS_DEFAULT_PROFILE=account1

C:\temp>aws configure list-profiles
default
account1
account2
account3

C:\temp>aws s3 ls
2024-08-06 16:37:04 codepipeline-eu-west-1-248104794832
2024-08-06 16:31:44 codepipeline-us-east-1-747284203952
2024-08-05 20:19:38 website-production-bucket
2024-08-05 20:13:57 website-source-bucket1

C:\temp>
```