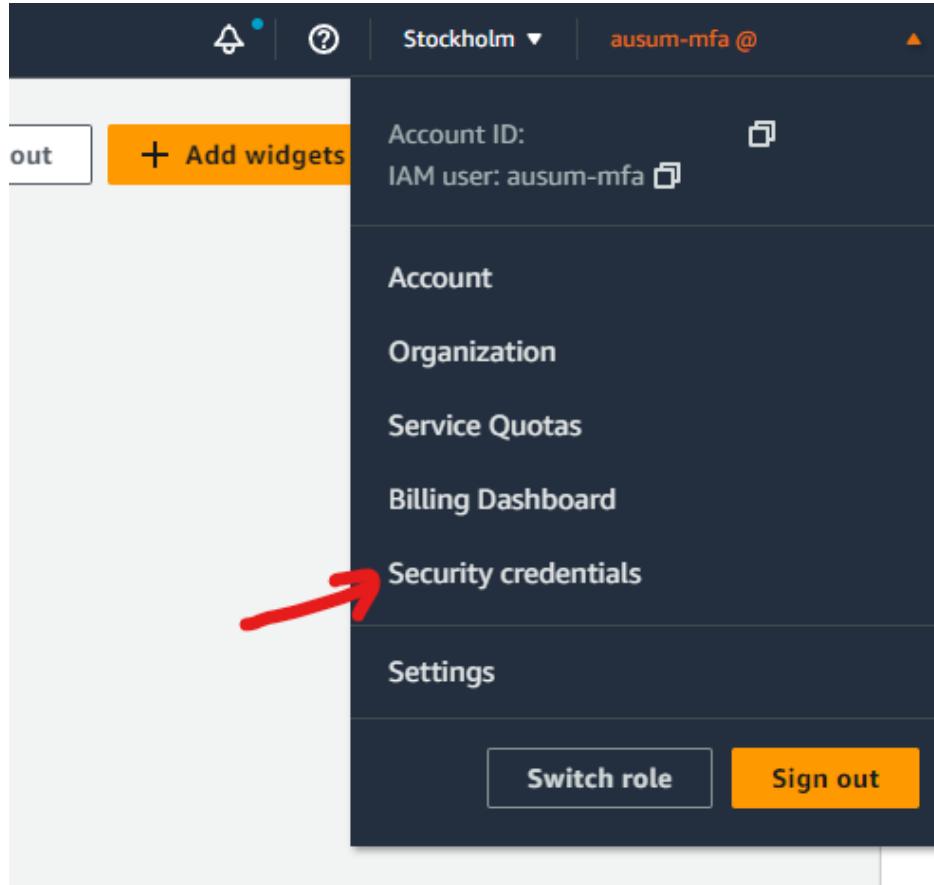




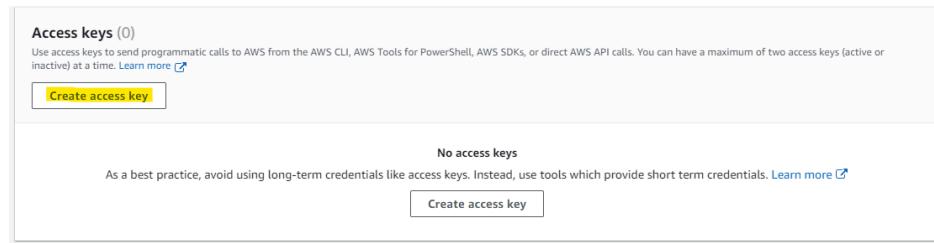
## AWS CLI AND ACCESS KEYS USAGE

## Create an Access Key

Go to your profile under security credentials:



Press Create Access key button under Access keys section:



Select Command Line Interface (CLI), accept recommendations and continue.

**Command Line Interface (CLI)**  
You plan to use this access key to enable the AWS CLI to access your AWS account.

**Local code**  
You plan to use this access key to enable application code in a local development environment to access your AWS account.

**Application running on an AWS compute service**  
You plan to use this access key to enable application code running on an AWS compute service like Amazon EC2, Amazon ECS, or AWS Lambda to access your AWS account.

**Third-party service**  
You plan to use this access key to enable access for a third-party application or service that monitors or manages your AWS resources.

**Application running outside AWS**  
You plan to use this access key to enable an application running on an on-premises host, or to use a local AWS client or third-party AWS plugin.

**Other**  
Your use case is not listed here.

**⚠ Alternatives recommended**

- Use [AWS CloudShell](#), a browser-based CLI, to run commands. [Learn more ↗](#)
- Use the [AWS CLI V2](#) and enable authentication through a user in IAM Identity Center. [Learn more ↗](#)

I understand the above recommendation and want to proceed to create an access key.

[Cancel](#)
Next

## Create access key

### Set description tag - *optional*

The description for this access key will be attached to this user as a tag and shown alongside the access key.

Description tag value  
Describe the purpose of this access key and where it will be used. A good description will help you rotate this access key confidently later.

Maximum 256 characters. Allowed characters are letters, numbers, spaces representable in UTF-8, and: \_ . : / = + - @

[Cancel](#) [Previous](#) [Create access key](#)

At next window we will get our key id and password. Copy and save this as we are going to use it before.

## Retrieve access keys

**Access key**

If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.

Access key	Secret access key
AKIARISYTLTPOA6HZE4	***** <a href="#">Show</a>

**Access key best practices**

- Never store your access key in plain text, in a code repository, or in code.
- Disable or delete access key when no longer needed.
- Enable least-privilege permissions.
- Rotate access keys regularly.

For more details about managing access keys, see the [Best practices for managing AWS access keys](#).

[Download .csv file](#) [Done](#)

## Instal AWS Cli

Just follow steps for your system:

<https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html>

## Configure AWS Cli with your credentials

Add your access key to credentials file (located at C:\users\username\.aws\credentials or ~/.aws/credentials).

*[default]*

*aws\_access\_key\_id=AKIAIOSFODNN7EXAMPLE*

*aws\_secret\_access\_key=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY*

Once added content to file execute following command to get temp MFA credentials and token:

```
aws sts get-session-token --serial-number arn-of-the-mfa-device --token-code code-from-token
--duration-seconds 129600
```

arn-of-the-mfa-device can be retrieved at your profile Security Credentials:

**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 ↗](#)

Device type	Identifier	Created on
Virtual	arn:aws:iam::087158905062:mfa/ausum-mfa	22 minutes ago

code-from-token it's your MFA app active token/number.

Once executed you will receive response like:

```
PS C:\Users\hecto> aws sts get-session-token --serial-number arn:aws:iam::087158905062:mfa/ausum-mfa --token-code 257885
--duration-seconds 129600
{
    "Credentials": {
        "AccessKeyId": "ASIAIRISYTLDTPSONWN5U",
        "SecretAccessKey": "YczXX0tz708CQPvVUG/axTDGnNdhrNFXYn4kZrBK",
        "SessionToken": "IQoJb3JpZ2luX2VjEFkaCmV1LXNvdXRoLTEiRzBFAiAoUKNWVdnyV6A+TGRWswRA01yE6av5u3ZLA7LaL2EsXwIhAKCp08e
EI4Yj7Q+mK5MBET8rz6eYCo3WJtefDrYTDQ2Ku88CDiQABoMMMdg3MTU4OTA1MDYyIgxzz8zDoi68+G4eDmwqzAF215w6uZjqZ/g8wV8QfOWSMa+Tq6bUOE
lFEZE0Bdc04CEteNsWU+OZmY54seSBYUMJv6u3miMZ1BOP9dKwp3Na9cTLmLRZeUvtL7wfbnM/s4NPM1jXLA1lzt1XLJG0hIo+UMnE9YHT0fBN4/+9rh+S
MyZYEVdw7A4X27KFI187PU557iCc7eo+DkmogVfQxAdsybxPfe8uTfkmOWWP2JRM63ijtr5Q8WiRZI/D6TdqQWHJwbqemrr075sxbccS7oX3MU
w75KHoQY6mAFCoGuQ1s/NGfa/jWg9n71zAhSbuBH5WAPCeZQXBGBvZP+7rK7V/Wyzd4t6h116JH1+mVnou7tdpCLv/qq471QZE/gkcFyb4BVgWYUwPmYlnvn
N5A+qTvZa1nq9bpJ4UK120xbY9foXVeQMHsXZvagCwwdVYDOLgoIND8Uyiu3chlOFV8iJlQr3G/Gry38HIQWWlCAqMdSQg==",
        "Expiration": "2023-03-29T04:50:55+00:00"
    }
}

PS C:\Users\hecto>
```

Create new profile at your credentials file with obtained data  
(C:\users\username\.aws\credentials or ~/.aws/credentials)

[default]

aws\_access\_key\_id=XXXXXXXXXXXXXXXXXXXX

aws\_secret\_access\_key=XXXXXXXXXXXXXXXXXXXX

[mfa]

aws\_access\_key\_id = ASIAIRISYTLDTPSONWN5U

aws\_secret\_access\_key = YczXX0tz7Q8CQPvVUG/axTDGnNdhrNFXYn4kZrBK

aws\_session\_token =  
IQoJb3JpZ2luX2VjEFkaCmV1LXNvdXRoLTEiRzBFAiAoUKNWVdnyV6A+TGRWswRA01yE6av5u3ZLA7LaL2EsXwIhAKCp08e
A7LaL2EsXwIhAKCp08eEI4Yj7Q+mK5MBET8rz6eYCo3WJtefDrYTDQ2Ku88CDiQABoMMMdg3MTU4OTA1MDYyIgxzz8zDoi68+G4eDmwqzAF215w6uZjqZ/g8wV8QfOWSMa+Tq6bUOE
co4CEteNsWU+OZmY54seSBYUMJv6u3miMZ1BOP9dKwp3Na9cTLmLRZeUvtL7wfbnM/s4NPM1jXLA1lzt1XLJG0hIo+UMnE9YHT0fBN4/+9rh+S
MyZYEVdw7A4X27KFI187PU557iCc7eo+DkmogVfQxAdsybxPfe8uTfkmOWWP2JRM63ijtr5Q8WiRZI/D6TdqQWHJwbqemrr075sxbccS7oX3MU
w75KHoQY6mAFCoGuQ1s/NGfa/jWg9n71zAhSbuBH5WAPCeZQXBGBvZP+7rK7V/Wyzd4t6h116JH1+mVnou7tdpCLv/qq471QZE/gkcFyb4BVgWYUwPmYlnvn
N5A+qTvZa1nq9bpJ4UK120xbY9foXVeQMHsXZvagCwwdVYDOLgoIND8Uyiu3chlOFV8iJlQr3G/Gry38HIQWWlCAqMdSQg==

Once done we could execute commands with --profile mfa parameter and get Access to resources:

```
PS C:\Users\hecto> aws sts get-caller-identity --profile mfa
{
    "UserId": "AIDARISYTLDTKGMKAGOB2",
    "Account": "123456789012",
    "Arn": "arn:aws:iam::123456789012:user/ausum-mfa"
}
```

MFA credentials will be active for 36 hours (--duration-seconds 129600).

### **Assume Playlogiq DEV/STA role to access dev and staging account**

You will have to add config profile with role config to Access dev and staging account and resources. Edit config file (C:\users\username\.aws\credentials or ~/.aws/credentials) and add as follows:

*[default]*

*region = eu-south-1*

*[profile mfa]*

*region = eu-south-1*

*[profile playlogiqdev]*

*region = eu-south-1*

*role\_arn = arn:aws:iam::129323733169:role/playlogiq*

*source\_profile = mfa*

To execute commands using this profile and Access to dev/staging resources you Will need to add --profile playlogiqdev:

```
PS C:\Users\hecto> aws sts get-caller-identity --profile assumed-role
{
    "UserId": "AROAR4HCLSCYTAPSPOL50:botocore-session-1679936566",
    "Account": "123456789012",
    "Arn": "arn:aws:sts::123456789012:assumed-role/playlogiq/botocore-session-1679936566"
}
```

### Retrieve logs using aws-cli (example)

If you need to get all logs for last day you will have to run command following commands:

*Betmaker – PRO*

```
aws logs tail betmaker --since 1d > betmaker_pro.log
```

*Betmaker – Staging*

```
aws logs tail betmaker-sta --since 1d --profile playlogiqdev > betmaker_stा.łog
```

*Betmaker – Dev*

```
aws logs tail betmaker-dev --since 1d --profile playlogiqdev > betmaker_dev.log
```

*Backoffice – PRO*

```
aws logs tail backoffice --since 1d > backoffice_pro.log
```

*Backoffice – Staging*

```
aws logs tail backoffice-sta --since 1d --profile playlogiqdev > backoffice_stा.łog
```

*Backoffice – Dev*

```
aws logs tail backoffice-dev --since 1d --profile playlogiqdev > backoffice_dev.log
```

This are example command you can use filters or other commands to get logs just check AWS doc if you need more data:

<https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/SearchDataFilterPattern.html>