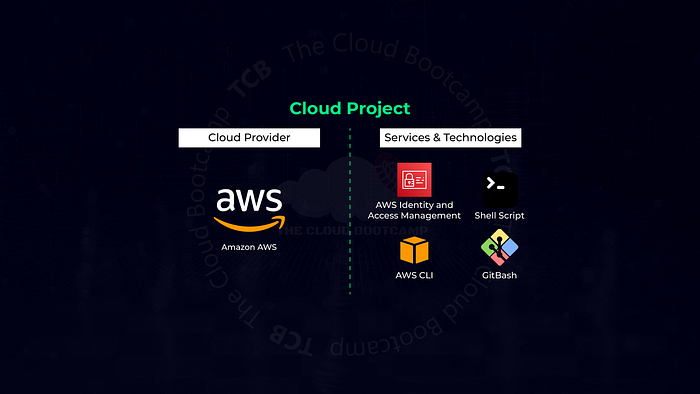
# Automating user creation with AWS Identity and Access Management (IAM) resources

## Introduction

In this article, I will demonstrate a simple automated user creation process to help you avoid the repetitive manual tasks in provisioning AWS accounts.

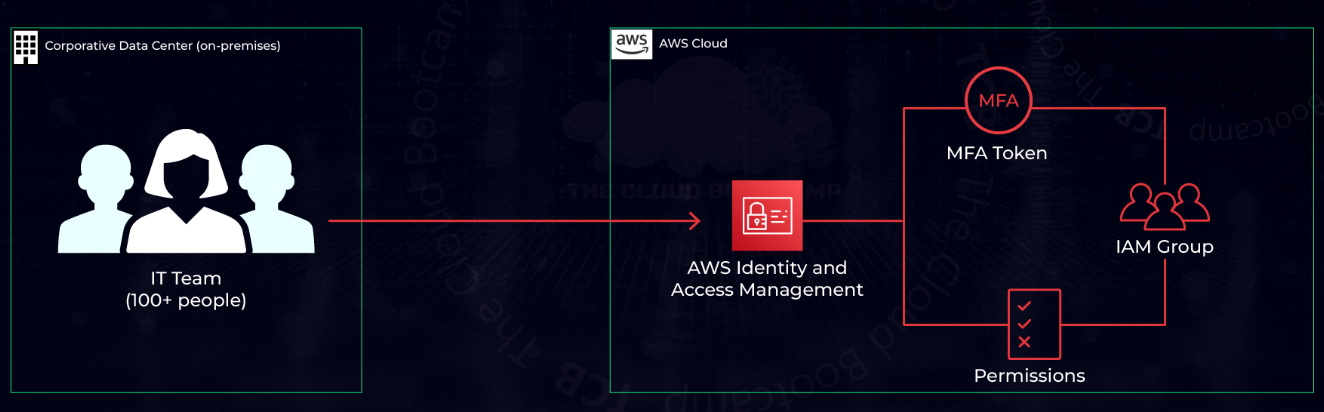


Note: AWS-specific account information will be redacted in all images and sample outputs. If you follow the steps in this article and change any object or file names, ensure you update the commands given herein with the names you used.

## Use Case

Let’s assume you need to migrate 100+ users; each user must have Multi-factor Authentication (MFA) enabled, and their profile must be set to require the password to be reset on the first login. Each user must be assigned a specific user group to maintain the best practice of minimal privileged user access required for their role. Since we are not federating user accounts, we must also provide the ability for the users to change their passwords.

## Solution Architecture



### MFA Policy

Our MFA policy must allow the logged-in user to:

* Allow the user to see their user profile.
* Allow the user to view MFA devices.
* Manage their own MFA device.
* Deny access to all other services if MFA has not been set up.

Download the policy JSON file here.

Important note: By defining the resource in the policy, the user can only create an MFA device named as their username. Example: "Resource": "arn:aws:iam::\*:mfa/${aws:username}”

To implement the MFA policy in your AWS environment:

1. Log in to your AWS console
2. Open the CloudShell
3. Upload the MFA policy JSON file
4. Execute the following command

aws iam create-policy --policy-document file://enforce\_mfapolicy.json --policy-name EnforceMFAPolicy

The output from the command will look something like this:

{

"Policy": {

"PolicyName": "EnforceMFAPolicy",

"PolicyId": "ANPA2M5UXEMCJ65EWEUEZ",

"Arn": "arn:aws:iam::714954187524:policy/EnforceMFAPolicy",

"Path": "/",

"DefaultVersionId": "v1",

"AttachmentCount": 0,

"PermissionsBoundaryUsageCount": 0,

"IsAttachable": true,

"CreateDate": "2023-04-26T03:29:09+00:00",

"UpdateDate": "2023-04-26T03:29:09+00:00"

}

}

## Prerequisites

The following package is required to run the scripts to create user groups and accounts.

1. Log in to your AWS console
2. Open the CloudShell
3. Execute the following command

sudo yum install dos2unix -y

### Automating User Group Creation

The script to create the predefined user groups automatically will take the group name and default policy from an input CSV file. The script will also apply the standard AWS IAM User Change Password and the MFA enforcement policies we created to the user group.

The input file is a simple CSV file with group and policy attributes. Example:

group,policy

CloudAdmin,AdministratorAccess

DBA,AmazonRDSFullAccess

LinuxAdmin,AmazonEC2FullAccess

NetworkAdmin,AmazonVPCFullAccess

Trainees,ReadOnlyAccess

Download the script file here.

Edit the aws-iam-create-group.sh file and replace the <policy arn> for the MFA Policy, for example:

Change:

aws iam attach-group-policy --group-name $group --policy-arn <policy arn>

To:

aws iam attach-group-policy --group-name $group --policy-arn arn:aws:iam::714954187524:policy/EnforceMFAPolicy

Steps to create the groups:

1. Log in to your AWS console
2. Open the CloudShell
3. Upload the aws-iam-create-group.sh script file and your groups.csv file
4. Execute the following command to set up permission to execute the script

chmod +x aws-iam-create-group.sh

1. Execute the following command to create the groups

./aws-iam-create-group.sh groups.csv

Sample output for the CloudAdmin group given in the sample input file data:

{

"Group": {

"Path": "/",

"GroupName": "CloudAdmin",

"GroupId": "AGPA2M5UXEMCAMKRBTLTZ",

"Arn": "arn:aws:iam::714954187524:group/CloudAdmin",

"CreateDate": "2023-04-26T01:19:13+00:00"

}

}

Graphical user interface, text, application, email

Description automatically generated

AWS UI User Group Listing

Graphical user interface, text, application, email

Description automatically generated

Permissions listing of an AWS User Group

### Automatically Create Users and Assign them to Designated Groups

The script to create the predefined list of users automatically will take the username, group name, and password from an input CSV file. The script will also implement the requirement to reset the user’s password on the first login.

The input file is a simple CSV file with user, group, and password attributes. Example:

user,group,password

jane.doe,DBA,ChangeMe123456!

john.doe,NetworkAdmin,ChangeMe123456!

billy.joe,CloudAdmin,ChangeMe123456!

jim.bob,LinuxAdmin,ChangeMe123456!

mary.sunshine,Trainees,ChangeMe123456!

Download the script file here.

Steps to create the groups:

1. Log in to your AWS console
2. Open the CloudShell
3. Upload the aws-iam-create-user.sh script file and your users.csv file
4. Execute the following command to set up permission to execute the script

chmod +x aws-iam-create-user.sh

1. Execute the following command to create the groups

./aws-iam-create-group.sh groups.csv

Sample output for the jane.doe user given in the sample input file data:

{

"User": {

"Path": "/",

"UserName": "jane.doe",

"UserId": "AIDA2M5UXEMCA64D2IFUL",

"Arn": "arn:aws:iam::714954187524:user/jane.doe",

"CreateDate": "2023-04-26T02:54:20+00:00"

}

}

{

"LoginProfile": {

"UserName": "jane.doe",

"CreateDate": "2023-04-26T02:54:22+00:00",

"PasswordResetRequired": true

}

}

Graphical user interface, text, application, email

Description automatically generated

AWS User Group Listing

Shows that each group now has user assigned.

Graphical user interface, text, application, email

Description automatically generated

AWS CloudAdmin Users Listing

Shows that the expected user has been created and assigned to the CloudAdmin group.

Graphical user interface, application

Description automatically generated

AWS Users Listing

Shows that the user, mary.sunshine, has logged in and set up her MFA device.