

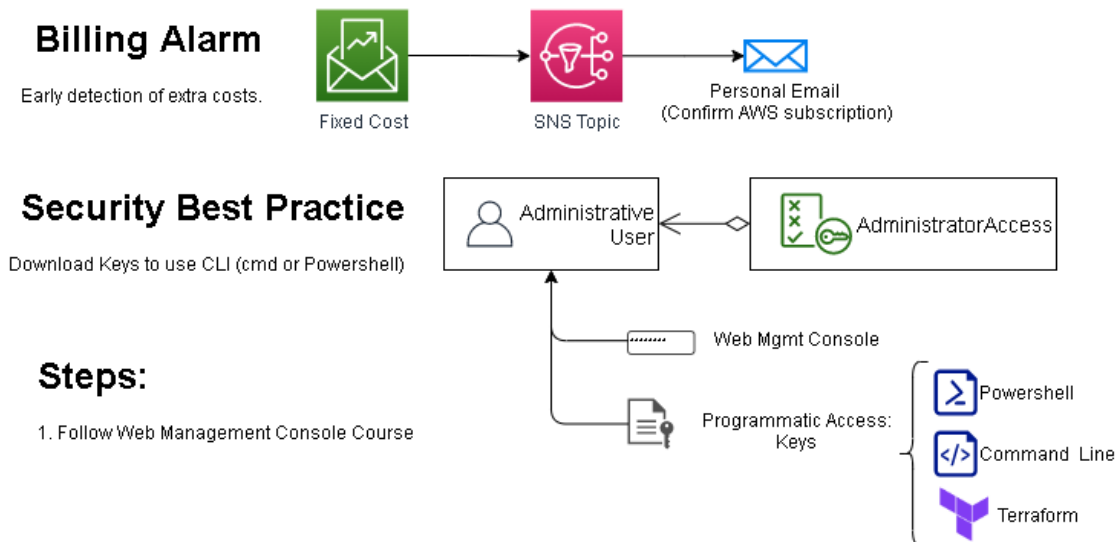
Contents

Purpose	2
General Diagram	2
Prerequisites	2
Lab 1 using Web Management Console.....	2
Stablish Billing Alarm.....	2
Create administrative user	4
Get Account Number	10
Lab 1 using Command Line (Windows)	12
Prerequisites	12
Identify current user	12
Get Account ID	13
Describe regions.....	14
Get Budget Information	14
Lab 1 using Powershell (Windows)	15
Prerequisites	15
Identify current user	15
Get Account ID	16
Describe regions.....	16
Get Budget Information	17
Lab 1 using Terraform (Windows).....	17
Evidence to send.	20

Purpose

General Idea of this lab is to have the management of your account using CLI or Web Console; in addition to control expenses using Billing Alarm, and fulfil the best practice to have an IAM user instead of having a root account to make programmatic changes.

General Diagram



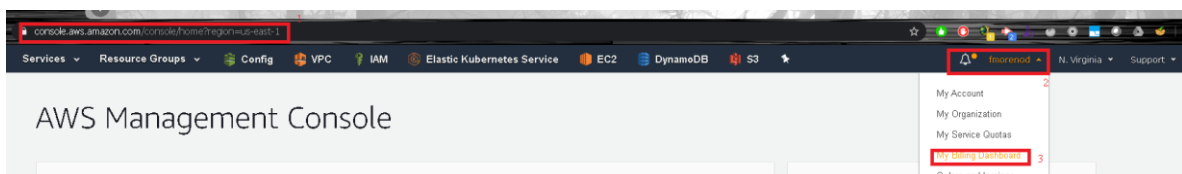
Prerequisites

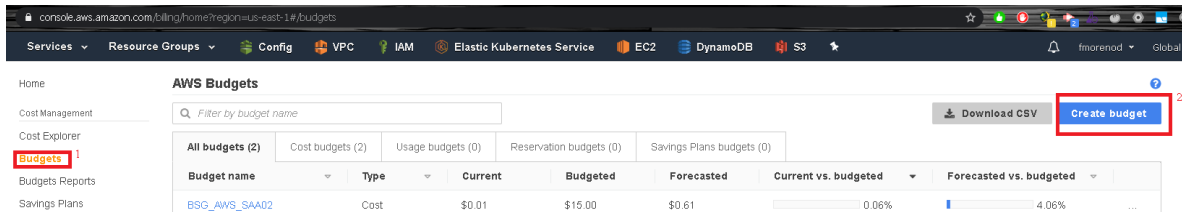
Create a AWS Free Tier Account following <https://aws.amazon.com/free/free-tier/> or to get a free one if your student or educator using <https://aws.amazon.com/education/awseducate/>

Lab 1 using Web Management Console

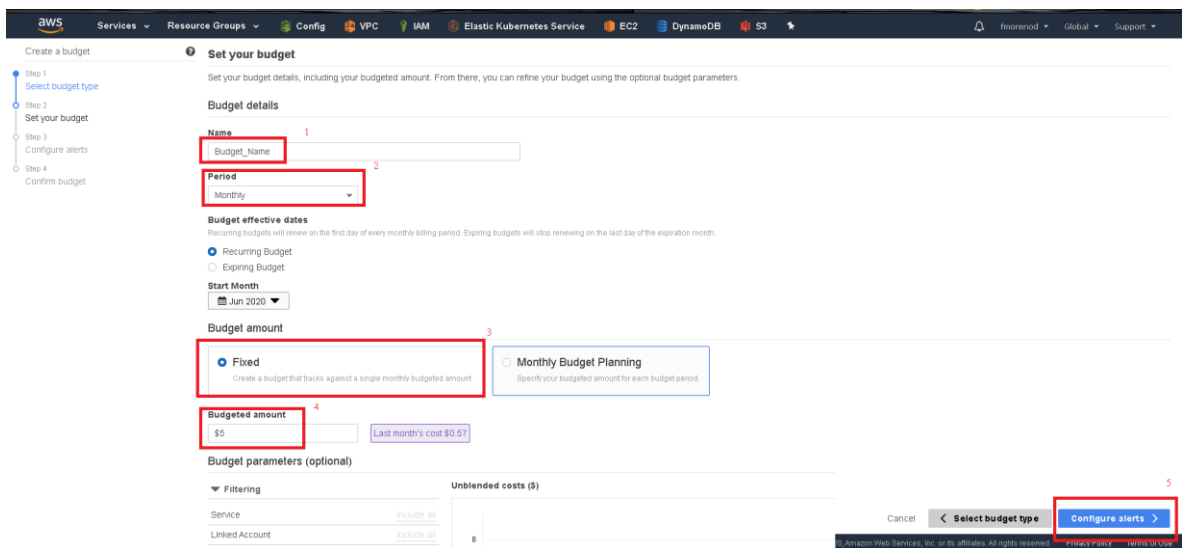
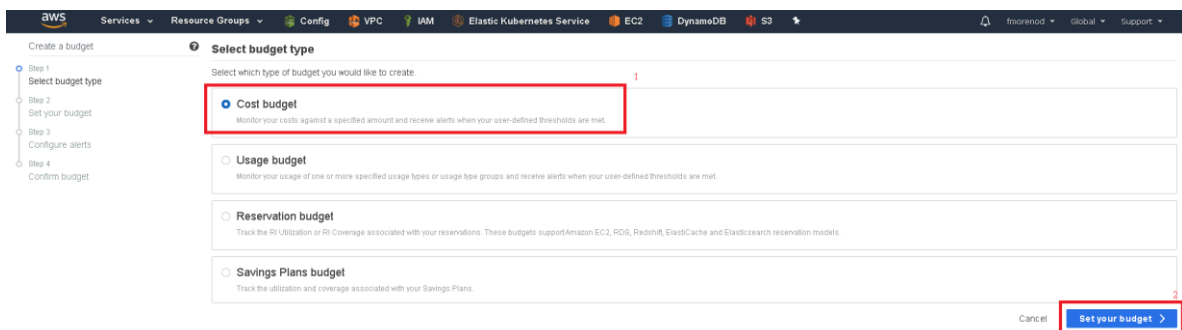
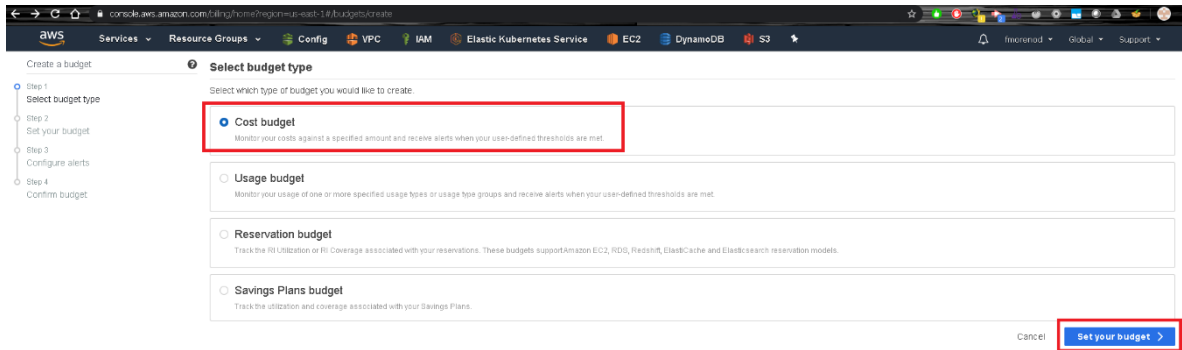
Stablish Billing Alarm

Make an alarm to have the control de of your expenses of your AWS account. It's your responsibility stablish a value of money that you can spend on this course.

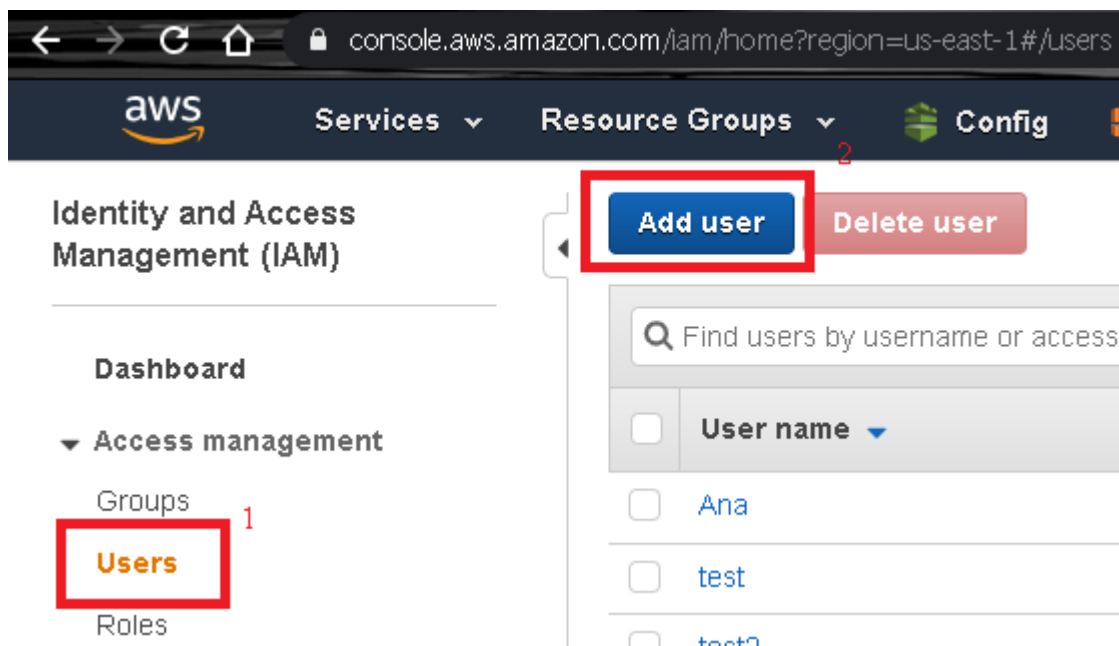
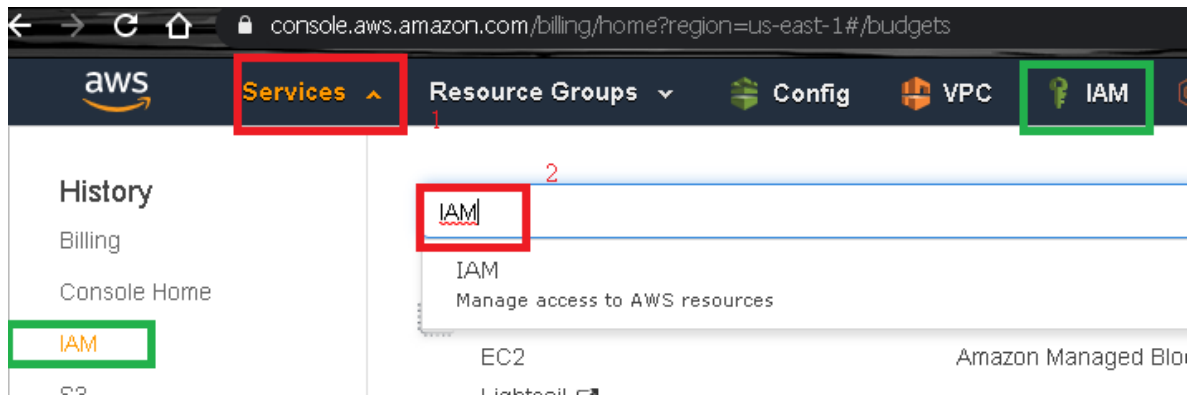




In this case, I configure a monthly stipend, however you can establish another fixed expense.



You will receive an email with the confirmation of the subscription of this topic/budget.



Add user



Set user details

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name

[Add another user](#)

Select AWS access type

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

- Access type*
- ☒ **Programmatic access**
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.
 - ☒ **AWS Management Console access**
Enables a **password** that allows users to sign-in to the AWS Management Console.

Console password*

☐ Autogenerated password

☒ Custom password

☒ Show password

Require password reset ☐ User must create a new password at next sign-in

* Required

[Cancel](#)


[Next: Permissions](#)


It's better to have a full administrator Access.


Add user

1 2 3 4 5


Set permissions

 Add user to group

 Copy permissions from existing user

 Attach existing policies directly

Create policy



Filter policies Showing 550 results

	Policy name	Type	Used as
<input checked="" type="checkbox"/>	AdministratorAccess	Job function	Permissions policy (3)
<input type="checkbox"/>	AlexaForBusinessDeviceSetup	AWS managed	None
<input type="checkbox"/>	AlexaForBusinessFullAccess	AWS managed	None
<input type="checkbox"/>	AlexaForBusinessGatewayExecution	AWS managed	None
<input type="checkbox"/>	AlexaForBusinessLifesizeDelegatedAccessPolicy	AWS managed	None
<input type="checkbox"/>	AlexaForBusinessPolyDelegatedAccessPolicy	AWS managed	None
<input type="checkbox"/>	AlexaForBusinessReadOnlyAccess	AWS managed	None
<input type="checkbox"/>	AmazonAPIGatewayAdministrator	AWS managed	None

Set permissions boundary

Cancel

Previous


Next: Tags

Add user

1 2 3 4 5

Add tags (optional)

IAM tags are key-value pairs you can add to your user. Tags can include user information, such as an email address, or can be descriptive, such as a job title. You can use the tags to organize, track, or control access for this user. [Learn more](#)

Key	Value (optional)	Remove
<input type="text" value="Description"/>	<input type="text" value="Administrator User"/>	
<input type="text" value="Add new key"/>	<input type="text"/>	

You can add 49 more tags.

Cancel Previous **Next: Review**

Add user



Review

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

User details

User name	testname
AWS access type	Programmatic access and AWS Management Console access
Console password type	Custom
Require password reset	No
Permissions boundary	Permissions boundary is not set

Permissions summary

The following policies will be attached to the user shown above.

Type	Name
Managed policy	AdministratorAccess

Tags

The new user will receive the following tag

Key	Value
Description	Administrator_User

[Cancel](#)[Previous](#)[Create user](#)

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Please store those credentials, because you have to configure using “aws configure” or on environment variables.

Add user

1 2 3 4 5



Success

You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: <https://fmoreno-test.signin.aws.amazon.com/console>

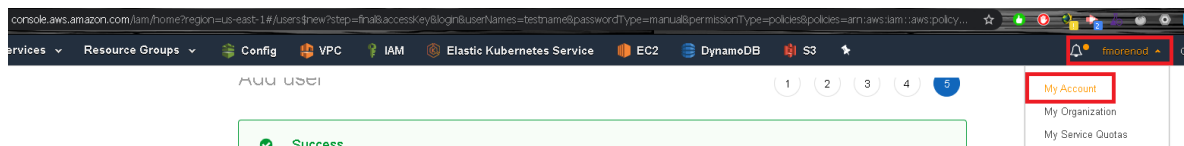
Download .csv

	User	Access key ID	Secret access key	Email login instructions
▶	testname	AKIA3FYICIHBS6XCWLWA	***** Show	Send email
			alrnogYSLh+cpCAfbxYyJC O67Sd/1YboHsxKws5P Hide	

Close

Copy the text for Access key ID and Secret Access key, the same as the CSV file.

Get Account Number



Top navigation bar: **fmorenod** (1)

Left sidebar: **My Account** (2), My Organization

Account Settings

Account Id: 68312754627
Seller: AWS Inc.
Account Name: fmorenod
Password: *****

Contact Information

Please note that updating your contact information on this page will not update the information displayed on your PDF Invoices. If you wish to update the billing address information associated with your invoice, please edit it through the Payment Methods page, located [here](#).

Lab 1 using Command Line (Windows)

Follow the same steps of Web Management Console for:

Stablish Billing Alarm

Create Administrative User

Get AWS Account ID

Prerequisites

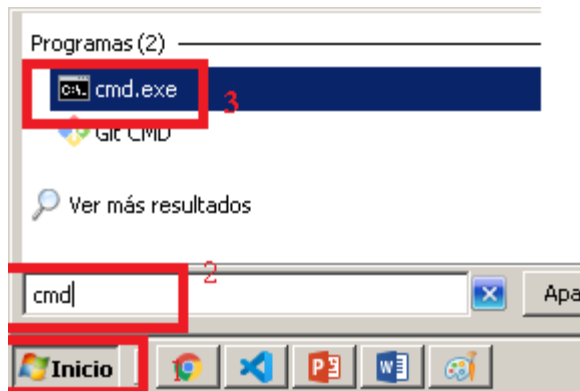
Download an AWS CLI for according OS, following

<https://docs.aws.amazon.com/cli/latest/userguide/install-cliv2.html>

Jq is a tool to parse JSON text, so you can download to make readable output information. You can install using instructions on <https://stedolan.github.io/jq/download/>

Identify current user

Open a console



Configure the variables of Access key ID and Secret Access key that you create on Create Administrative User.

In addition you have you configure the default region.

```
set AWS_ACCESS_KEY_ID=AKIA3FYCYIHBS6XCWLWA
```

```
set AWS_SECRET_ACCESS_KEY=almogYSLh+cpCAfbX/yYJC067Sd/1YboHsxKws5P
```

```
set AWS_DEFAULT_REGION=us-east-1
```

```
C:\Users\Administrador>set AWS_ACCESS_KEY_ID=AKIA3FYCYIHBS6XCWLWA
C:\Users\Administrador>set AWS_SECRET_ACCESS_KEY=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
C:\Users\Administrador>set AWS_DEFAULT_REGION=us-east-1
C:\Users\Administrador>
```

You have to make the same activity everytime that you like to send commands or you can configure the CLI to identify AWS User automatically using "aws configure" and insert the same data.

```
ca. Administrador: C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Users\Administrador>aws configure
AWS Access Key ID [None]: AKIA3FYQCIHBS6XCWLWA
AWS Secret Access Key [None]: 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
Default region name [us-east-1]:
Default output format [json]:

C:\Users\Administrador>_
```

Get Account ID

Following the instructions

```
rem Obtenemos datos de la cuenta por medio de una obtencion de identidad STS
aws sts get-caller-identity
aws sts get-caller-identity --query Account
aws sts get-caller-identity --query Account --output text
rem Unicamente obtenemos un valor de esa cuenta, podriamos realizarlo usando
jq si tenemos json
aws sts get-caller-identity|jq ".Account"
rem Obtener variable ejecutada en entorno
aws sts get-caller-identity --query Account --output text >tmpFile
set /p accountId= < tmpFile
del tmpFile
echo La variable de entorno es %accountId%
```

```
C:\Users\Administrador>aws sts get-caller-identity
{
  "UserId": "AIDA3FYQCIHBS37BTU4M7U",
  "Account": "768312754627",
  "Arn": "arn:aws:iam::768312754627:user/testname"
}

C:\Users\Administrador>aws sts get-caller-identity --query Account
"768312754627"

C:\Users\Administrador>aws sts get-caller-identity --query Account --output text
768312754627

C:\Users\Administrador>aws sts get-caller-identity|jq ".Account"
"768312754627"

C:\Users\Administrador>aws sts get-caller-identity --query Account --output text >tmpFile
C:\Users\Administrador>set /p accountId= < tmpFile
C:\Users\Administrador>del tmpFile

C:\Users\Administrador>echo La variable de entorno es %accountId%
La variable de entorno es 768312754627

C:\Users\Administrador>
```

Describe regions

You can check available regions for your account using

```
aws ec2 describe-regions
```

```
C:\Users\Administrador>aws ec2 describe-regions
{
  "Regions": [
    {
      "Endpoint": "ec2.eu-north-1.amazonaws.com",
      "RegionName": "eu-north-1",
      "OptInStatus": "opt-in-not-required"
    },
    {
      "Endpoint": "ec2.ap-south-1.amazonaws.com",
      "RegionName": "ap-south-1",
      "OptInStatus": "opt-in-not-required"
    },
    {
      "Endpoint": "ec2.eu-west-3.amazonaws.com",
      "RegionName": "eu-west-3",
      "OptInStatus": "opt-in-not-required"
    }
  ]
}
```

Get Budget Information

Running

```
aws budgets describe-budgets --account-id %accountId%
```

or

```
aws budgets describe-budgets --account-id %accountId% |jq ".Budgets[0].BudgetName"
```

, you get detailed information of budget that you already created.

```
C:\Users\Administrador>aws budgets describe-budgets --account-id %accountId%
{
  "Budgets": [
    {
      "BudgetName": "BSG_AWS_SAA02",
      "BudgetLimit": {
        "Amount": "15.0",
        "Unit": "USD"
      },
      "PlannedBudgetLimits": [
        {
          "Amount": "15.0",
          "Unit": "USD",
          "PlannedBudgetLimit": "1590969600"
        },
        {
          "Amount": "50.0",
          "Unit": "USD",
          "PlannedBudgetLimit": "1593561600"
        },
        {
          "Amount": "15.0",
          "Unit": "USD",
          "PlannedBudgetLimit": "1596240000"
        }
      ]
    }
  ]
}
```

Lab 1 using Powershell (Windows)

Follow the same steps of Web Management Console for:

Stablish Billing Alarm

Create Administrative User.

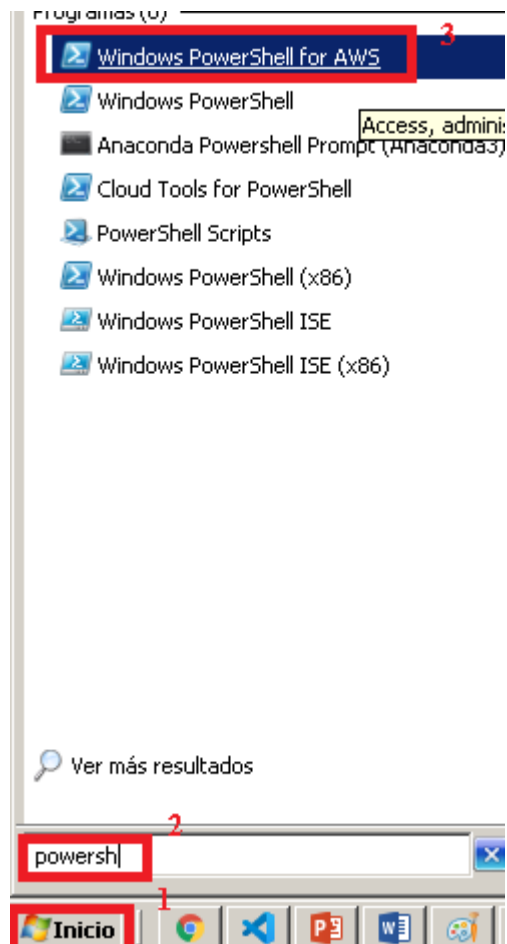
Prerequisites

Download an AWS Powershell Tools for according OS, following

<https://docs.aws.amazon.com/powershell/latest/userguide/pstools-getting-set-up.html>

Identify current user

Open Powershell Tools for AWS



For default, Powershell request the user using the same configuration of “aws configure”, however we will create the environment variables to identify the current user.

<# Primero ingresar los datos de configuracion de la cuenta, obtenidos del archivo csv del IAM #>

\$Env:AWS_ACCESS_KEY_ID="AKIA3FYCCIHBQTOK4ZXS"

\$Env:AWS_SECRET_ACCESS_KEY="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"

\$Env:AWS_DEFAULT_REGION="us-east-1"

```
Administrador: Windows PowerShell for AWS
PS>$Env:AWS_ACCESS_KEY_ID="AKIA3FYCCIHBQTOK4ZXS"
PS>$Env:AWS_SECRET_ACCESS_KEY="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"
PS>$Env:AWS_DEFAULT_REGION="us-east-1"
PS>
```

Get Account ID

<# Obtenemos datos de la cuenta por medio de una obtencion de identidad STS #>

Get-STSCallerIdentity

<# Unicamente obtenemos un valor de esa cuenta, podriamos realizarlo usando jq si tenemos json #>

(Get-STSCallerIdentity).Account

\$accountId = @(Get-STSCallerIdentity).Account

\$accountId

```
PS>Get-STSCallerIdentity
Account                               Arn                                     UserId
-----
768312754627                         arn:aws:iam::768312754627:user/test    AIDA3FYCCIHBXGUB6WM3G

PS>(Get-STSCallerIdentity).Account
768312754627
PS>$accountId = @(Get-STSCallerIdentity).Account
PS>$accountId
768312754627
PS>
```

Describe regions

<# Vamos a traer el listado de regiones que estan disponibles para esa cuenta #>

Get-EC2Region


```
PS>Get-EC2Region
```

Endpoint	RegionName
ec2.eu-north-1.amazonaws.com	eu-north-1
ec2.ap-south-1.amazonaws.com	ap-south-1
ec2.eu-west-3.amazonaws.com	eu-west-3
ec2.eu-west-2.amazonaws.com	eu-west-2
ec2.eu-west-1.amazonaws.com	eu-west-1
ec2.ap-northeast-2.amazonaws.com	ap-northeast-2
ec2.ap-northeast-1.amazonaws.com	ap-northeast-1
ec2.sa-east-1.amazonaws.com	sa-east-1
ec2.ca-central-1.amazonaws.com	ca-central-1
ec2.ap-southeast-1.amazonaws.com	ap-southeast-1
ec2.ap-southeast-2.amazonaws.com	ap-southeast-2
ec2.eu-central-1.amazonaws.com	eu-central-1
ec2.us-east-1.amazonaws.com	us-east-1
ec2.us-east-2.amazonaws.com	us-east-2
ec2.us-west-1.amazonaws.com	us-west-1
ec2.us-west-2.amazonaws.com	us-west-2

Get Budget Information

```
<# Vamos a traer alguna informacion del budget que realizamos recientemente
#>
```

```
Get-BGTBudgetList -AccountId $accountId
```

```
PS>Get-BGTBudgetList -AccountId $accountId
```

BudgetLimit	: Amazon.Budgets.Model.Spend
BudgetName	: BSG_AWS_SAA02
BudgetType	: COST
CalculatedSpend	: Amazon.Budgets.Model.CalculatedSpend
CostFilters	: <>
CostTypes	: Amazon.Budgets.Model.CostTypes
LastUpdatedTime	: 06/12/2020 6:50:40
PlannedBudgetLimits	: <[1590969600, Amazon.Budgets.Model.Spend], [1593561600, Amazon.Budgets.Model.Spend], [1596240000, Amazon.Budgets.Model.Spend], [1598918400, Amazon.Budgets.Model.Spend]>
TimePeriod	: Amazon.Budgets.Model.TimePeriod
TimeUnit	: MONTHLY
BudgetLimit	: Amazon.Budgets.Model.Spend
BudgetName	: BSG_AWS_SAA02_Example2

Lab 1 using Terraform (Windows)

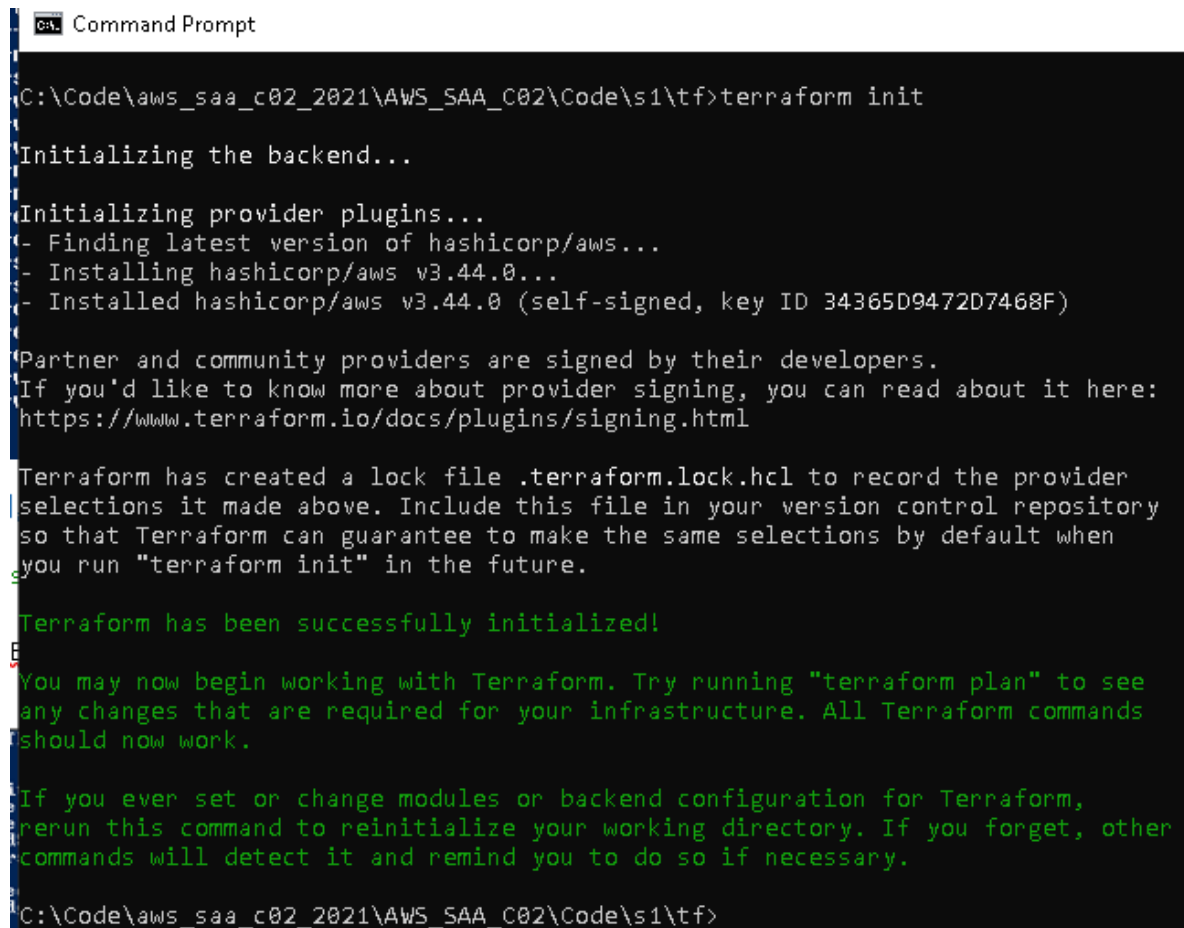
You must download the GIT repository on a folder, then you have to configure \$HOME/aws/credentials from the Access Key and Secret Access Key for instance for Identify current user (Command Line), similar to have this file:

The screenshot shows a Notepad++ window titled "C:\Users\Francisco\aws\credentials - Notepad++". The window contains the following text:

```
1 [default]
2 aws_access_key_id=AKIA3FYCIHB5CSU2D5I
3 aws_secret_access_key=9Ohyd2BbRM4oNuZcbzHIUdiTGYN3xBx2+z1IpM2O
```

Then, we have to execute on the same folder that TF files:

```
terraform init
```



```
Command Prompt
C:\Code\aws_saa_c02_2021\AWS_SAA_C02\Code\s1\tf>terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v3.44.0...
- Installed hashicorp/aws v3.44.0 (self-signed, key ID 34365D9472D7468F)

Partner and community providers are signed by their developers.
If you'd like to know more about provider signing, you can read about it here:
https://www.terraform.io/docs/plugins/signing.html

Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.

C:\Code\aws_saa_c02_2021\AWS_SAA_C02\Code\s1\tf>
```

```
terraform plan
```

```
C:\Code\aws_saa_c02_2021\AWS_SAA_C02\Code\s1\tf>terraform plan
```

```
An execution plan has been generated and is shown below.  
Resource actions are indicated with the following symbols:
```

```
Terraform will perform the following actions:
```

```
Plan: 0 to add, 0 to change, 0 to destroy.
```

```
Changes to Outputs:
```

```
+ account_id = "768312754627"  
+ caller_arn = "arn:aws:iam::768312754627:user/test"  
+ caller_user = "AIDA3FYCYIHBRZG6UXFVM"
```

```
-----  
Note: You didn't specify an "-out" parameter to save this plan, so Terraform  
can't guarantee that exactly these actions will be performed if  
"terraform apply" is subsequently run.
```

```
terraform apply
```

```
C:\Code\aws_saa_c02_2021\AWS_SAA_C02\Code\s1\tf>terraform apply
```

```
An execution plan has been generated and is shown below.  
Resource actions are indicated with the following symbols:
```

```
Terraform will perform the following actions:
```

```
Plan: 0 to add, 0 to change, 0 to destroy.
```

```
Changes to Outputs:
```

```
+ account_id = "768312754627"  
+ caller_arn = "arn:aws:iam::768312754627:user/test"  
+ caller_user = "AIDA3FYCYIHBRZG6UXFVM"
```

```
Do you want to perform these actions?
```

```
Terraform will perform the actions described above.  
Only 'yes' will be accepted to approve.
```

```
Enter a value: yes write yes
```

```
Apply complete! Resources: 0 added, 0 changed, 0 destroyed.
```

```
Outputs:
```

```
account_id = "768312754627"  
caller_arn = "arn:aws:iam::768312754627:user/test"  
caller_user = "AIDA3FYCYIHBRZG6UXFVM"
```

Evidence to send.

To have a review, the student must send some screenshots to instructor email:

1. Screenshot of confirmed subscription of Billing Alarm. You must create using Web Management Console on section [Stablish Billing Alarm](#).
2. Detailed Budget Information.