

# AWS – IAM & EC2

Le cloud, ou bien comme disent nos cousins québécois “nuage infogéré”.

## Introduction du sujet

---

Dans ce projet, vous allez voir comment utiliser les potentialités du cloud, plus précisément celles de AWS pour apprendre à administrer une infrastructure cloud.

Il sera impératif de documenter chacune des étapes qui vous ramèneront à la réalisation des Jobs.

## Job 0

---

Commencer par se créer un [compte AWS](#) !



# Job 1

---

La première étape consistera à voir le service AWS IAM :

**faites vos recherches dessus et notez-les dans votre documentation.**

Maintenant que vous avez vu et compris ce à quoi ce service sert, il est important de noter quelque chose. Il y a une règle de sécurité qui s'appelle le principe du moindre privilège qui est important de respecter tout au long de ce projet, faites vos recherches dessus et notez-le dans votre documentation.

**Vous devez donc créer 6 utilisateurs qui seront les suivants :**

premièrement votre utilisateur IAM avec lequel vous vous connecterez pour éviter d'utiliser le compte root, utilisez votre nom et prénom simplement (prénom\_nom), et ensuite :

- **Jeff Bezos ( admin )**
- **Elon Musk ( admin )**
- **Mark Zuckerberg ( utilisateur simple sans droit )**
- **Steve Jobs ( utilisateur simple sans droit )**
- **Bill Gates ( utilisateur simple sans droit )**

Maintenant switchez de compte, utilisez uniquement votre IAM User personnel.



## Job 2

---

Maintenant que vous avez implémenté vos IAM Users, il serait bien de créer des groupes et de les répartir avec une architecture cohérente, les groupes ainsi que leurs permissions seront les suivants :

- ***Developers ( AdministratorAccess )***
- ***Audit team ( IAMFullAccess )***
- ***Operations ( IAMReadOnlyAccess )***

placez Mark Zuckerberg dans le groupe "Developers", Steve Jobs dans "Audit Team" et Bill Gates dans "Operations".

## Job 3

---

Parlons sécurité et politique de mot de passe, sur AWS, il est vivement recommandé d'attribuer des politiques de mots de passes pour ses utilisateurs, Votre mission est la suivante, mettez en place une politique de mot de passe qui sera celle-ci :

- ***minimum 12 caractères***
- ***le mot de passe doit inclure un caractère en majuscule***
- ***le mot de passe doit inclure un caractère en minuscule***
- ***le mot de passe doit inclure un nombre***
- ***le mot de passe doit inclure un caractère spécial (!?. etc...)***



Mais au-delà de ça, il est **VIVEMENT** recommandé d'activer la **MFA ( Multiple Factor Authentication )**, autrement dit la double authentification et au minimum sur le compte root.

**Activez donc la double authentification sur le compte root** et servez vous d'un outil comme "Google Authenticator" ou bien "Authy"

justifier via des explications et des captures d'écran toutes les étapes que vous avez suivies.

## Job 4

---

Maintenant que vous avez vos utilisateurs, vos groupes et que la sécurité y est implémentée, vous allez voir Comment faire de l'administration AWS autrement que sur la UI du site web de ce dernier avec des AWS Access Key et la CLI AWS :

Utilisez vos AWS Access Keys pour accéder de manière sécurisée aux services AWS via la CLI qui vous permettra d'interagir avec les services AWS directement à partir d'un terminal.

Commencez par télécharger la CLI AWS 2 pour vous connecter à cette dernière.



## Job 5

---

En prévision des futurs besoins en terme d'infrastructure, vous savez en tant qu'expert cloud que vous devez créer un **IAM Role**, faites vos recherches dessus et notez-la pour la présenter à vos équipes qui feront partie du groupe "Developers" qui ne sont pas des Administrateurs d'infrastructure cloud ( cela vous sera demandé lors de votre passage à l'oral ).

Une fois le travail de recherche effectué, créez un IAM Role qui s'appellera "**DemoForEC2**" auquel vous attacherez la permission "**IAMReadOnlyAccess**".

## Job 6

---

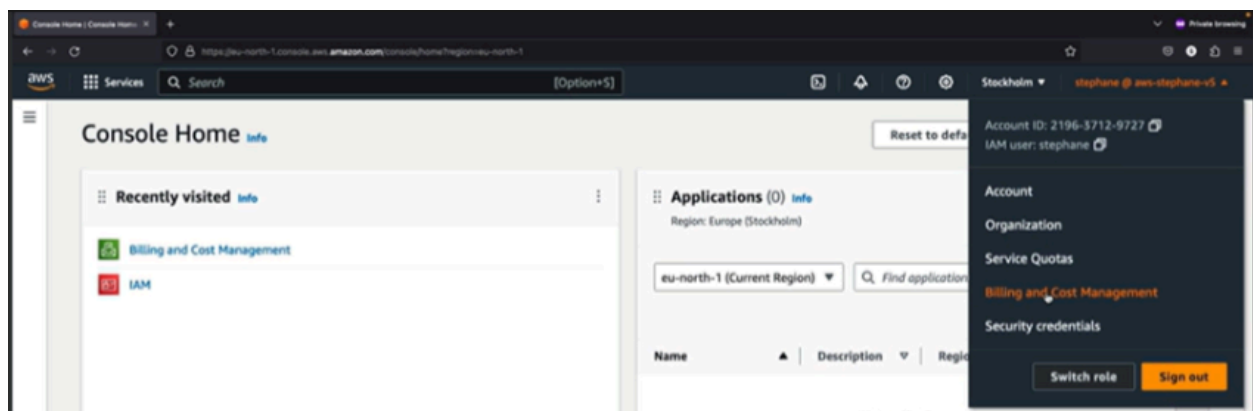
En termes de sécurité également, en tant que chef de l'infrastructure cloud, il est important de surveiller les va-et-vient et de savoir qui s'est connecté à quel moment, pour ça il y a **AWS Credentials Report**, générez le et enregistrez-le.



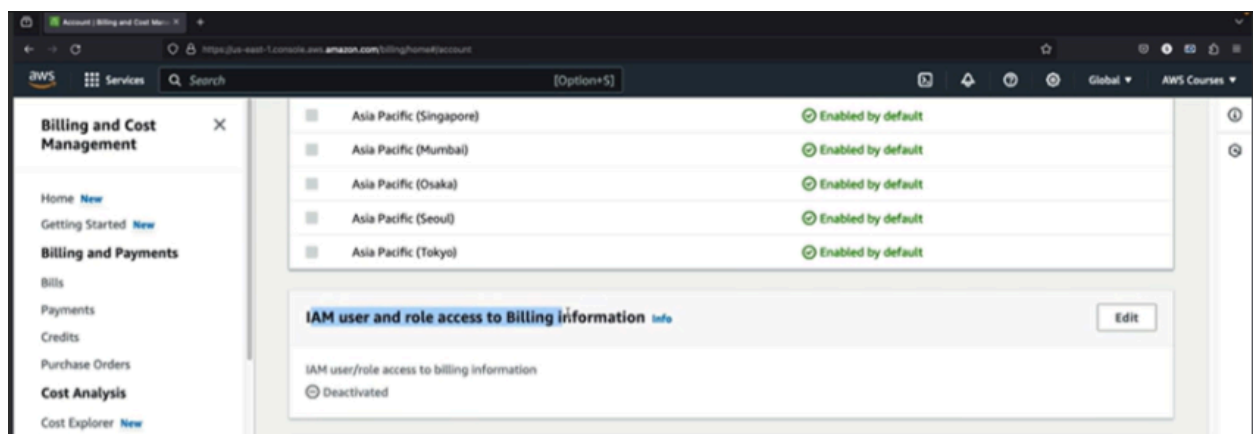
## Job 7

Maintenant que vous avez mis en place la base de vos utilisateurs, de groupes et d'IAM Roles en implémentant les bonnes règles, **IL EST IMPORTANT DE METTRE DES POLITIQUES DE BUDGET POUR ÉVITER DE PAYER POUR DES SIMPLES TEST SUR AWS, POUR LE FAIRE IL SUFFIT DE L'ACTIVER**, ( le projet va utiliser uniquement la partie “free-tiers” de AWS pour qu’il n’y ait aucune facturation ) et suivre les étapes suivantes :

cliquez sur “Billing and Cost Management” :

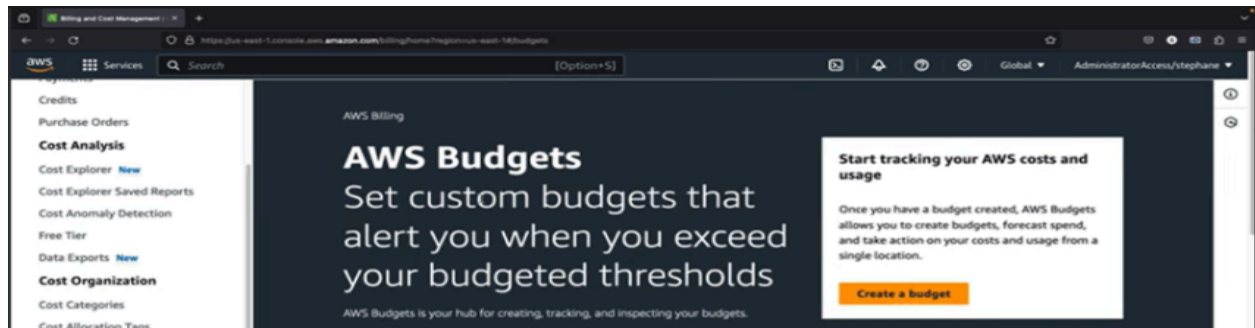


cliquez sur “Edit” :

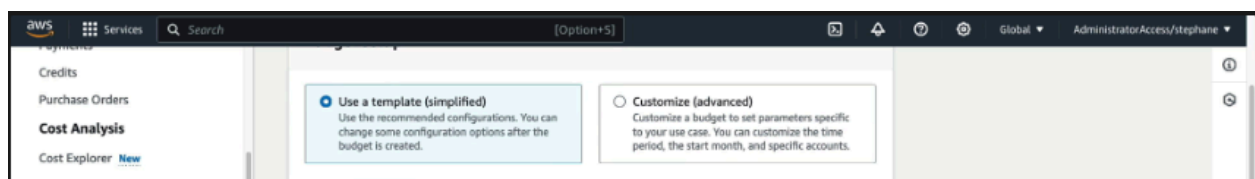




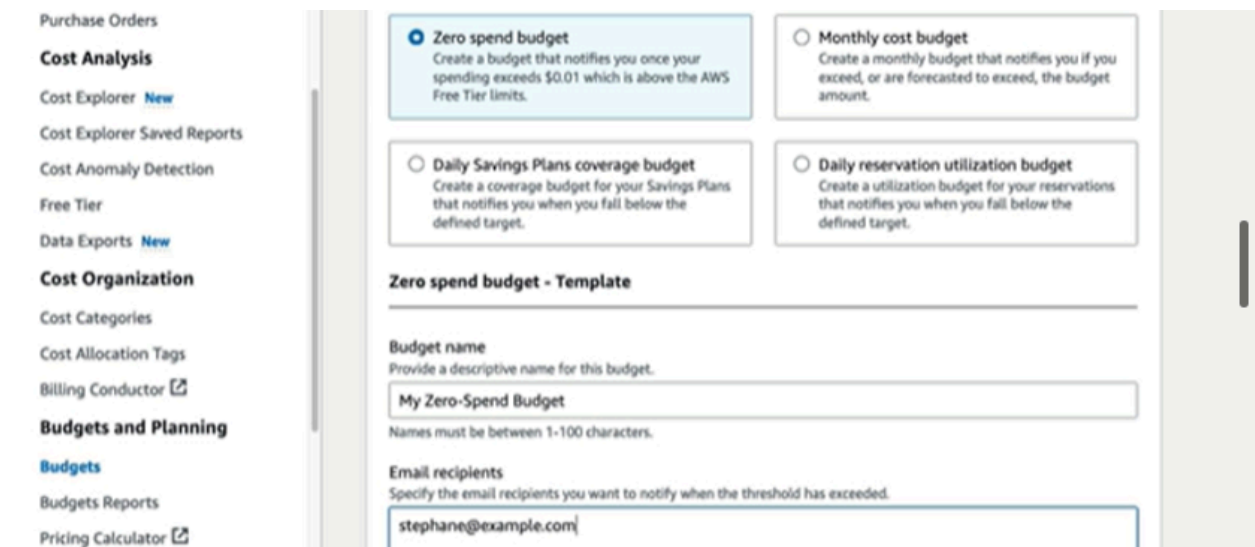
cliquez sur "Create a budget" :



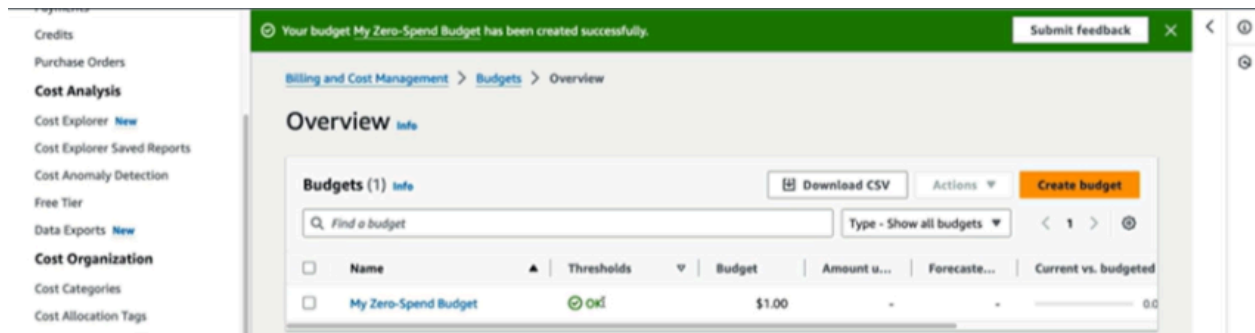
sélectionner "Use a template (simplified)" :



sélectionner "Zero spend budget" et renseigner un nom pour cette règle et également votre boîte mail pour recevoir les alertes dessus :



La politique budgétaire a bien été mise en place :



## Job 8

Le temps est venu d'aborder le service AWS le plus fréquemment utilisé, j'ai nommé **EC2 ( Elastic Compute Cloud )**, faites vos recherches dessus et documentez le sujet en répondant à ces questions :

**Qu'est ce que EC2 ?**

**Quelles sont ses options de configuration et les tailles disponibles ?**

**C'est quoi EC2 User Data et à quoi ça sert ?**

**Quelles sont les types d'instances qui existent ?**

## Job 9

Maintenant que vous avez documenté EC2, et que vous avez bien compris ce que ça comporte, vous devez vous lancer dans la création d'une instance EC2 qui devra respecter le cahier des charges qui vous a été donné pour l'environnement des développeurs de la société :

- Le nom de l'instance sera "**Serveur Web Dev**"
- L'image d'exécution sera "**Amazon Linux**"





- le type d'instance sera la plus petite ( renseignez vous bien )
- comme les développeurs devront y avoir accès depuis ssh, il faudra générer une key pair
- il faut que le trafic soit ouvert pour tout le monde sur les port 80 et 443
- comme storage, un disque gp2 de 8Go
- et dans la partie User Data, exécuter un script qui permet le déploiement d'un serveur web ( bash )

une fois tout cela fait, accédez à votre instance depuis un navigateur.

Bravo, vous venez de découvrir la puissance et la résilience du cloud.

Garder une trace de tout ce que vous venez de faire en capture d'écran et en documentant.

## Job 10

---

Les développeurs veulent avoir accès à l'instance via **SSH** qui est le moyen standard de se connecter aux instances EC2 Linux à partir d'un terminal.

Faites en sorte de vous connecter à votre instance.

## Job 11

---

**Attribuer à l'instance EC2 le IAM Role** que vous avez créé (cf. Job 5), et tapez la commande suivante dans votre instance en SSH pour voir si cela fonctionne :



```
aws iam list-users
```

**effectuer des captures d'écrans des étapes que vous avez suivies et documentez-les.**

## Job 12

---

Votre supérieur vous demande de lui faire un rapport sur l'ensemble des options d'achat EC2 qui existent pour pouvoir optimiser le coût de la facture, présenter lui les différentes possibilités et besoins en termes de performances, de coûts et de flexibilité dans une documentation.

## Job 13

---

Une fois que vous avez terminé tous les jobs précédents, **effectuez un snapshot du volume attaché à votre instance "Serveur web Dev"**, une fois cela fait, stopper et résilier l'instance que vous venez de créer.



## Compétences visées

---

- Administration réseau cloud
- Sécurité des systèmes d'information cloud
- Administration infrastructure cloud
- Gestion des accès
- Administration réseau

## Rendu

---

Présentation avec diapositive sur la totalité du projet, veille, concept et architecture.

## Base de connaissances

---

- [AWS IAM](#)
- [AWS EC2](#)
- [Budget Setup](#)
- [VPC Security Groups](#)
- [CLI AWS](#)