Prise en main AWS

F. Diakhaté, R. Fihue

Date: 08/11/2019

Aperçu et Objectifs

Ce TP va vous permettre de manipuler les principaux composants d'une solution de cloud IaaS en se basant sur les produits d'Amazon Web Services. L'offre de produits AWS est pléthorique et ne se limite pas à des composants relevant strictement de l'IaaS, nous n'allons donc balayer qu'une petite partie des produits disponibles qui représentent le coeur de toute offre IaaS: l'instanciation de réseaux et de machines virtuelles, l'allocation d'espaces de stockages de type bloc ou objet. Nous verrons aussi que des composants logiciels de plus haut niveau peuvent être déployés pour faciliter la mise en oeuvre d'infrastructures virtuelles.

Prise en main

Connectez-vous à la console AWS à l'adresse https://aws.amazon.com/. Prenez quelques minutes pour parcourir les différents services disponibles dans l'onglet *Services* en haut à gauche. N'hésitez pas à consulter l'aide qui est généralement bien conçue. Notez la présence de l'onglet *Region* en haut à droite: la plupart des services AWS sont localisés par région, vous allez donc choisir dans quelles région vous instanciez chaque ressource. Pour certaines ressources vous pourrez avoir un contrôle plus fin en spécifiant la zone de disponibilité souhaitée (*availability zone*) ou en utilisant des groupes de placement afin, respectivement, d'améliorer la tolérance aux pannes (en allouant des ressources dans des datacenters différents) ou de maximiser la performance (en allouant plusieurs ressources colocalisées).

Ce TP utilise la console en langue anglaise, vous pouvez configurer la langue de la console en bas à gauche.

Ces services étant payants il est important de pouvoir suivre sa consommation. Le service *Billing* vous donne des informations sur les coûts engendrés. Le service *CloudWatch* permet de surveiller le bon fonctionnement de votre infrastructure mais aussi les coûts associés.

Utilisez le service *CloudWatch* pour mettre en place une alerte dès que vos dépenses dépassent un certain montant, par exemple 10$. Pour cela, après avoir sélectionné ce service, cliquez sur *Billing* à gauche, puis sur le bouton *Create Alarm* et envoyez vous un E-mail en cas de dépassement via une notification SNS (Simple Notification Service).

S3

Simple Storage Service (S3) est le service de stockage objet de l'offre AWS. L'interface de ce service a la particularité d'être multi-region (global) mais vous aurez à choisir une région lorsque vous créerez les *Bucket* qui vont accueillir vos données.

- Créez un bucket dans la région N.Virginia (laissez initialement les paramètres par défaut)
- Uploadez un fichier par exemple: http://bit.ly/2PToa0D
- Rendez le fichier public et consultez-le depuis un navigateur et/ou avec la commande curl
- Consultez les méta-données du fichier et les possibilités de versionnage des objets ainsi que les différentes classes de stockage disponibles.

Cloudfront

Cloudfront est le service CDN (Content Delivery Network) d'AWS. Vos données S3 étant stockées aux Etats-Unis, un CDN va permettre d'accélérer les performances pour tous les accès depuis d'autres régions en mettant automatiquement en cache les données dans serveurs répartis dans le monde.

- Dans la console Cloudfront, créez une distribution ayant pour origine votre bucket S3 (laissez les paramètres de cache par défaut)
- Mesurez la latence d'accès à vos données en passant par le CDN ou en accédant directement à votre bucket

EC2

Amazon EC2 (**E**lastic **C**ompute **C**loud) est le principal service permettant d'instancier des machines virtuelles.

A partir de la console EC2, dans la région Paris, utilisez le bouton Launch instance pour démarrer une nouvelle VM. Choisissez le modèle Amazon Linux 2. Renseignez-vous sur les tarifs des différents types

d'instance proposés. Utilisez une instance t2.micro pour cet exemple. Passez directement à la dernière étape en cliquant sur review and launch.

L'interface vous demandera la génération d'une paire de clés afin de vous connecter à cette instance. Conservez bien votre clé privée en sûreté (`chmod 0400 %VOTRE_CLE_PEM%`). Vous pouvez aussi importer une clé publique existante au préalable en cliquant sur Key Pairs à gauche dans la console EC2.

Connectez vous à votre VM en suivant les instructions données en cliquant sur **Connexion** dans votre liste d'instances. La ligne de commande associée doit ressembler à :

```
$ ssh -i example_key.pem ec2-user@ec2-????.eu-west-3.compute.amazonaws.com
```

Étudiez dans la console les différentes

Une fois dans votre VM, lancez un serveur web :

```
$ sudo yum install -y httpd lynx
$ sudo systemctl enable httpd
$ sudo systemctl start httpd
```

C'est normal que cela ne marche pas, les politiques de sécurité par défaut n'autorise que le port ssh (tcp/22) sur vos machines. Il faut modifier le groupe de sécurité associé à votre VM.

Dans la description de votre instance vous trouverez un lien vers le groupe de sécurité associé.

Ajoutez une règle entrante qui autorise le port HTTP pour tout le monde puis réessayez d'y accéder.

Repérez dans l'interface que votre VM a une IP publique et un IP privée. Quelle IP est affectée à l'interface réseau de votre VM (utilisez la commande **ip**) ?

Essayez d'ajouter un périphérique bloc supplémentaire et vérifiez que vous le voyez apparaître dans votre VM avec la commande **lsblk**.

IAM et ligne de commande

Le service **IAM** (Identity and Access Management) permet de définir des utilisateurs et des groupes et de définir très précisément les permissions qui leur sont attribuées via des politiques (*policy*). Le compte initial avec lequel vous vous loggez à la console est équivalent à un compte root ayant tous les droits. Il est aussi à noter que les ressources créées, par exemple une VM, peuvent elles même se voir attribuer des permissions. Pour cela, il faut associer la ressource à un rôle qui est lui même associé à des politiques.

Connectez vous sur le dashboard **IAM**. Créez un nouvel utilisateur avec accès ligne de commande. Donnez lui tous les droits d'administration sur votre compte AWS (politique **AdministratorAccess**).

Cet utilisateur devra avoir un accès sans limite à AWS (policy **AdministratorAccess**).

Installez la CLI AWS sur votre poste si ce n'est pas déjà fait et configurez-la à l'aide de la commande **aws configure**. Entrez les détails d'authentification de votre utilisateur CLI. La région correspondant à Paris est **eu-west-3**.

Listez l'état de vos instances : `aws ec2 describe-instance-status`

Puis lancez une seconde VM en adaptant les paramètres de la commande suivante :

```
aws ec2 run-instances --image-id <ami> --count 1 --instance-type t2.micro
--key-name <keypair> --security-groups <security group>
```

Tentez de vous y connecter.

Créez un rôle ayant la même politique et attribuez-le à une de vos VMs. Vérifiez que vous pouvez faire fonctionner la CLI aws depuis cette VM sans avoir à rentrer d'identifiants.

Utilisez la CLI aws pour copier le fichier présent dans votre bucket S3 vers le répertoire /var/www/html de votre VM puis vérifiez que vous y accédez à travers son serveur http.

VPC

Nous avons pour l'instant créé nos VMs en utilisant le réseau virtuel par défaut d'AWS qui est préconfiguré avec des sous-réseaux vous offrant une connectivité vers l'extérieur et des IPs publiques. Pour bien comprendre le principe de fonctionnement des différents composants réseaux nous allons créer un nouveau VPC dans lequel nous déploierons deux types de VMs. Des VMs représentant des noeuds de login, accessible de l'extérieur en SSH et ayant des IPs publiques et des VMs représentant des noeuds de calcul ayant uniquement des IPs privées, et accessible seulement depuis les noeuds de logins en SSH et ICMP (ping). À des fins de haute disponibilité, nous répartirons des VMs de chaque type dans deux zones de disponibilité.

Dans la console VPC:

- Cliquez à gauche sur *Your VPCs* puis *Create VPC* (n'utilisez pas le *wizard*). Choisissez une plage d'adresses IPs privées (IPv4) pour votre VPC (gardez le mode tenancy à *default*).
- Cliquez à gauche sur *Subnets* et créez 4 subnets pour les login et compute dans les 2 zones de disponibilités.

- Cliquez à gauche sur *Internet Gateways* et ajoutez un routeur entre Internet et votre VPC
- Cliquez à nouveau sur *Subnets* set activez l'attribution d'IP publiques pour les subnets de login
- Cliquez sur *Route tables* et repérez la table de routage par défaut associée à vos sous-réseaux. Elle ne permet de communiquer qu'entre les IPs internes au VPC. Créez une nouvelle table de routage pour votre VPC et ajoutez une route par défaut (0.0.0.0/0) via le routeur Internet que vous venez de créer.

Créez maintenant une VM dans chaque subnet et vérifiez que vous obtenez bien la connectivité souhaitée entre Internet et les noeuds de logins, puis entre les noeuds de login et les noeuds de calcul. Il vous faudra pour cela configurer des security groups appropriés.

Vos noeuds de calcul n'ont aucun accès vers l'extérieur, il est donc difficile, par exemple, d'installer des paquets depuis vos noeuds de calcul. Mettez en place des passerelles NAT (*NAT Gateways*) entre les subnets de noeuds de calcul et Internet pour leur donner un accès sortant à internet sans leur attribuer d'IP publiques.

EFS

Vos VMs n'ont pas d'espace de stockage partagé comme on pourrait l'attendre d'un cluster de calcul type HPC. Le service EFS permet d'instancier un serveur NFS géré par AWS sans avoir à se préoccuper de la configuration du serveur ou du dimensionnement du stockage.

Utilisez EFS pour créer un espace de stockage réseau hautement disponible et montez-le dans l'ensemble de vos VMs.