

# Création d'une machine virtuelle sur Google Cloud Platform et exécution planifiée de scripts Python

**Réalisé par:** Fehizoro RANDRIAMISANTA

**Date de soumission :** 03/07/2023

## Table des matières

1. Introduction.....	3
2. Comprendre le cloud computing et les machines virtuelles.....	3
a) Qu'est-ce que le cloud computing ? .....	3
b) Qu'est-ce qu'une machine virtuelle ? .....	4
c) Qu'est-ce que Compute Engine ?.....	4
3. Création d'une machine virtuelle sur Google Cloud Platform avec Compute Engine .....	5
d) Étapes pour créer une machine virtuelle sur GCP .....	5
e) Paramètres clés à configurer lors de la création d'une machine virtuelle sur GCP .....	8
f) Accès à une machine virtuelle sur GCP après sa création .....	12
4. Installation et configuration de l'environnement Python sur une machine virtuelle GCP	14
5. Utilisation d'un planificateur pour exécuter un script Python sur une machine virtuelle	15
g) Qu'est-ce qu'un planificateur ? .....	15
h) exécuter des scripts Python de manière planifiée .....	15
i. Création du crontab.....	15
ii. Suppression du crontab.....	16
i) Options de planification disponibles sur GCP .....	16
6. Vérification de l'exécution réussie d'un script Python sur une machine virtuelle .....	17
7. Avantages de l'utilisation de Google Cloud Platform pour l'exécution de scripts Python planifiés .....	17
8. Principaux services de Google Cloud Platform utilisés en conjonction avec une machine virtuelle et un script Python .....	18
9. Conclusion .....	18
10. Bibliographie .....	19

## 1. Introduction

Dans le contexte actuel où les entreprises doivent gérer d'importants volumes de données, elles recherchent des solutions efficaces pour répondre à leurs besoins en matière d'infrastructures informatiques. L'utilisation de machines virtuelles peut être une solution pertinente à ces défis. Google Cloud Platform, la plateforme de cloud computing de Google, offre un large éventail de services et de fonctionnalités, dont Compute Engine qui permet la création et la gestion de machines virtuelles.

Ce rapport vise à fournir une description détaillée des étapes nécessaires à la création d'une machine virtuelle sur Google Cloud Platform (GCP) en utilisant Compute Engine, ainsi que l'exécution planifiée d'un script Python sur cette machine virtuelle. Nous explorerons les paramètres clés de configuration, les méthodes d'accès à la machine virtuelle, l'installation et la configuration de l'environnement Python, l'utilisation d'un planificateur pour exécuter des scripts Python de manière planifiée, les options de planification disponibles, la vérification des résultats et les avantages de l'utilisation de GCP pour l'exécution de scripts Python planifiés.

## 2. Comprendre le cloud computing et les machines virtuelles

Avant d'aborder le sujet de ce rapport de manière approfondie, il est essentiel de bien comprendre certaines notions telles que le cloud computing et les machines virtuelles afin de faciliter la compréhension des explications ultérieures.

### a) Qu'est-ce que le cloud computing ?

Le cloud computing, ou l'informatique en nuage, est un concept qui permet d'accéder à des ressources informatiques, telles que des serveurs, des bases de données, du stockage de données et des logiciels, via Internet. Au lieu d'avoir ces ressources physiquement sur l'ordinateur ou dans l'entreprise, on peut les louer à distance auprès d'un fournisseur de services cloud. En d'autres termes, si on a besoin d'un ordinateur très puissant pour exécuter une application ou stocker de grandes quantités de données, au lieu d'acheter et de gérer cet ordinateur nous-même, on peut simplement se connecter à un service cloud et utiliser cet ordinateur virtuel à distance. On ne se soucie donc pas de l'infrastructure sous-jacente ni de la maintenance, car tout cela est géré par le fournisseur de services cloud. Notons que l'infrastructure, dans le contexte du cloud computing, fait référence à l'ensemble des ressources informatiques nécessaires pour prendre en charge le fonctionnement d'une application ou d'un service. Cela inclut les serveurs, le stockage, les réseaux, les systèmes d'exploitation, les bases de données et d'autres composants nécessaires à l'exécution et à la gestion des applications.

Les avantages du cloud computing sont nombreux. Tout d'abord, il offre une grande flexibilité, car on peut augmenter ou diminuer nos ressources selon nos besoins, ce qui permet d'économiser de l'argent en ne payant que pour ce qu'on utilise réellement. Deuxièmement, il offre une accessibilité mondiale, car nous pouvons accéder à nos ressources depuis n'importe quel endroit avec une connexion Internet.

Il existe trois principaux types de service de cloud computing :

- **Infrastructure as a Service (IaaS)** : Avec IaaS, les utilisateurs ont un contrôle élevé sur l'infrastructure sous-jacente. Ils peuvent créer et gérer des machines virtuelles, installer des logiciels et configurer l'ensemble de l'infrastructure (exemple : Compute Engine de Google Cloud Platform (GCP)).
- **PaaS (Platform as a Service)** : Cela offre une plateforme de développement et d'exécution d'applications. Les utilisateurs peuvent développer, tester et déployer des applications sans se soucier de l'infrastructure sous-jacente (exemple : Google App Engine) .  
Le PaaS est particulièrement utile pour les développeurs qui veulent se concentrer sur le développement d'applications sans se soucier de l'infrastructure.
- **SaaS (Software as a Service)** : Le SaaS fournit des applications prêtes à l'emploi via Internet. Les utilisateurs accèdent à ces applications via un navigateur web ou une interface client léger sans avoir à gérer l'infrastructure ou les couches logicielles sous-jacentes (exemple : Microsoft Office 365). Le fournisseur de services cloud gère et maintient l'ensemble de l'infrastructure, y compris les serveurs, les bases de données et les mises à jour logicielles. Le SaaS convient aux utilisateurs qui veulent simplement utiliser des applications sans se soucier de l'infrastructure ni de la gestion logicielle. Donc, avec PaaS, on développe notre propre application en utilisant une plateforme alors qu'avec SaaS, on utilise juste une application prête à l'emploi

#### b) Qu'est-ce qu'une machine virtuelle ?

Une machine virtuelle est une simulation logicielle d'un ordinateur physique. Elle permet d'exécuter plusieurs systèmes d'exploitation et applications indépendamment sur un même matériel physique. En d'autres termes, une machine virtuelle crée un environnement isolé et autonome à l'intérieur d'un serveur physique, offrant ainsi une flexibilité et une efficacité accrues dans la gestion des ressources informatiques.

Grâce aux machines virtuelles, les entreprises peuvent optimiser l'utilisation de leurs infrastructures et réduire les coûts en consolidant plusieurs machines virtuelles sur un seul serveur physique. De plus, elles facilitent le déploiement rapide de nouvelles instances, l'évolutivité des ressources et la gestion efficace des applications.

Compute Engine est par exemple largement utilisé par les entreprises pour créer des machines virtuelles en raison de sa flexibilité, de sa performance fiable, de sa disponibilité élevée et de son intégration avec d'autres services de Google Cloud Platform. Il offre une solution puissante pour répondre aux besoins de déploiement et de gestion des machines virtuelles dans un environnement cloud.

#### c) Qu'est-ce que Compute Engine ?

Compute Engine est l'un des services phares de Google Cloud Platform (GCP). Prenons d'abord le temps d'expliquer un peu ce que c'est exactement GCP. GCP est une plateforme de services cloud fournie par Google. Elle offre un ensemble de services et d'outils (le stockage de données, le calcul, les bases de données, l'intelligence artificielle, l'analyse de données,...)

permettant aux entreprises de créer, déployer et gérer leurs applications et leurs données dans un environnement cloud sécurisé . Compute Engine est donc l'un des services proposés par GCP et il offre la possibilité de créer et de gérer des machines virtuelles (VM) à la demande. Il constitue une solution puissante pour exécuter diverses charges de travail, des applications web aux charges de calcul intensives.

Avec Compute Engine, on peut sélectionner la configuration de la machine virtuelle en termes de puissance de calcul, de mémoire, de stockage et de réseau, afin de répondre aux besoins spécifiques de notre application.

### 3. Création d'une machine virtuelle sur Google Cloud Platform avec Compute Engine

#### d) Étapes pour créer une machine virtuelle sur GCP

Voici donc les étapes à suivre pour créer une machine virtuelle sur Google Cloud Platform :

1. Il faut tout d'abord créer un compte Google en allant sur la page principale de Google
2. Une fois le compte créé, on peut aller sur la console Google Cloud Platform (la Console Google Cloud Platform est une interface basée sur le Web fournie par Google pour gérer et administrer les services et les ressources de Google Cloud Platform) et nous connecter avec les identifiants Google. La console ressemble à ceci :

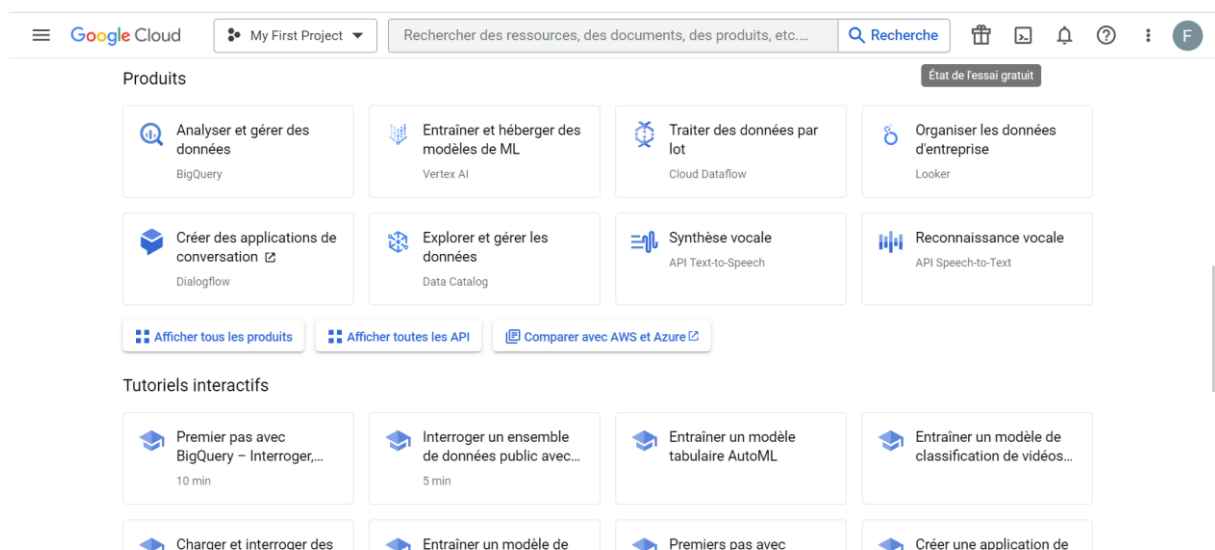
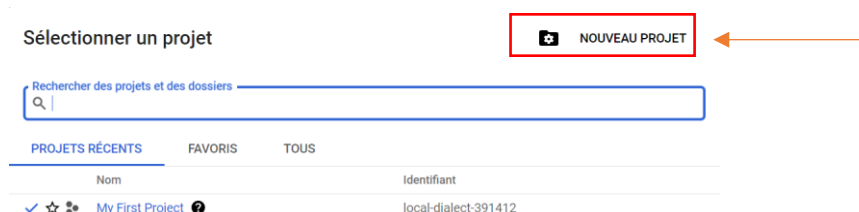
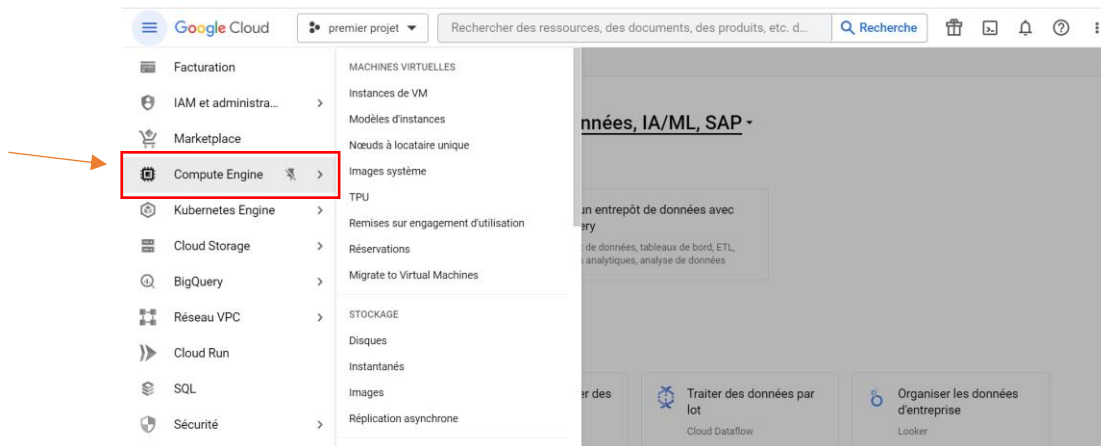


Figure 1 Console Google Cloud Platform

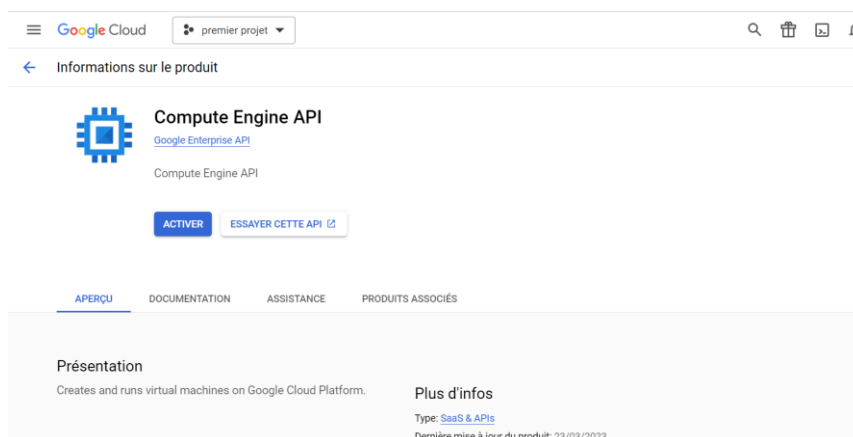
3. On peut maintenant créer un nouveau projet en cliquant sur « nouveau projet »



- On donne un nom à notre projet et on choisit une zone si on a une organisation. Dans le contexte de GCP, une organisation est une entité qui regroupe plusieurs projets sous une structure hiérarchique. Les organisations sont généralement utilisées dans le cadre de déploiements d'entreprise où plusieurs équipes ou départements ont leurs propres projets.
- On peut maintenant accéder à Compute Engine en le sélectionnant dans le menu de navigation ou on recherche « Compute Engine » dans la barre de recherche en haut de la page.



- On clique sur « Instances de VM » pour commencer le processus de création. Chaque fois qu'on crée une instance de VM, on crée essentiellement une copie individuelle et exécutable d'une machine virtuelle avec ses propres spécifications et configurations.
- Il faut activer l'API Compute Engine sur Google Cloud Platform (GCP) afin de pouvoir créer une machine virtuelle (VM) en utilisant le service Compute Engine. L'activation de l'API est nécessaire car elle permet d'accéder et d'utiliser les fonctionnalités spécifiques fournies par le service Compute Engine. Une API (Interface de programmation d'application) est un ensemble de protocoles et de règles qui permettent à différentes applications informatiques de communiquer entre elles. Ici, l'API Compute Engine est une interface qui permet aux développeurs et aux utilisateurs d'interagir avec le service Compute Engine.



8. Cela fait, on arrive sur cette page et on peut sélectionner une option pour créer une instance

← Créer une instance CODE ÉQUIVALENT

Pour créer une instance de VM, sélectionnez l'une de ces options :

- Nouvelle instance de VM**  
Créer entièrement une instance de VM unique
- Nouvelle instance de VM à partir d'un modèle  
Créer une instance de VM unique à partir d'un modèle existant
- Nouvelle instance de VM à partir d'une image système  
Créer une instance de VM unique à partir d'une image système existante
- Marketplace  
Déployer une solution prête à l'emploi sur une instance de VM

**NOM** \*  
instance-1

**GÉRER LES TAGS ET LES ÉTIQUETTES.**

**Région \***  
us-west4 (Las Vegas)  
La sélection d'une région est définitive

**Zone \***  
us-west4-b  
La zone est définitive

**Configuration de la machine**

☒ Usage général ☐ Optimisé pour le calcul ☐ Mémoire optimisée ☐ GPU

Types de machines pour les charges de travail courantes permettant d'optimiser les coûts et la flexibilité

Série  
E2

**Estimation mensuelle**  
**28,65 \$US**  
Soit un coût horaire d'environ 0,04 \$US  
Vous payez ce que vous consommez : facturation à la seconde, sans frais initiaux

Élément	Estimation mensuelle
2 vCPU + 4 GB memory	27,55 \$US
Disque persistant avec équilibrage, d'une taille de 10 Go	1,10 \$US
<b>Total</b>	<b>28,65 \$US</b>

[Tarifs de Compute Engine](#)

[L'ESS](#)

**CRÉER** **ANNULER** CODE ÉQUIVALENT

- **Nouvelle instance de VM** : Cette option nous permet de créer une nouvelle machine virtuelle à partir de zéro en spécifiant les détails de configuration tels que le type de machine, la région, la zone, le système d'exploitation, etc. On a un contrôle complet sur la configuration de la machine virtuelle.
- **Nouvelle instance de VM à partir d'un modèle** : Cette option nous permet de créer une machine virtuelle en utilisant un modèle préconfiguré qu'on a créé précédemment. Les modèles peuvent contenir des configurations prédéfinies, des applications installées, des paramètres réseau, etc. On peut réutiliser ces modèles pour simplifier le processus de création de nouvelles machines virtuelles.
- **Nouvelle instance de VM à partir d'une image système** : Avec cette option, on peut créer une machine virtuelle en utilisant une image système prédéfinie fournie par Google Cloud Platform. Les images système contiennent des systèmes d'exploitation populaires tels que Debian, Ubuntu, Windows, etc. On peut choisir l'image système appropriée et personnaliser la configuration de la machine virtuelle.
- **Marketplace** : Cette option nous permet d'accéder à la marketplace de Google Cloud Platform, où on choisit des images préconfigurées et des solutions logicielles prêtes à l'emploi provenant de partenaires et de développeurs tiers. On peut sélectionner une image à partir de la marketplace pour créer une machine virtuelle avec des applications ou des configurations spécifiques déjà préinstallées.

Pour ce rapport, on va créer une machine virtuelle à partir de zéro donc on va prendre l'option « Nouvelle instance de VM »

La partie « Estimation mensuelle » à droite de l'écran est une estimation des coûts mensuels associés à l'utilisation de cette machine virtuelle. Cette estimation nous donne une idée approximative du montant qu'on peut attendre à payer pour l'utilisation de la machine virtuelle sur une base mensuelle. Cette estimation dépend de plusieurs facteurs tels que le type de la machine virtuelle, la durée d'utilisation, les ressources associées, la région et la zone,...

## e) Paramètres clés à configurer lors de la création d'une machine virtuelle sur GCP

On va donc voir en détail les paramètres à prendre en compte lors de la création de la machine virtuelle.

### - Nom :

Nom \*  
instance-1

c'est le nom qu'on donne à la machine virtuelle pour l'identifier. Le nom de l'instance doit être unique à l'intérieur du projet GCP. Il doit commencer par une lettre, contenir uniquement des lettres minuscules, des chiffres ou des tirets (-), et ne doit pas dépasser 63 caractères.

### - Région et zone :

Région \*  
us-west4 (Las Vegas)

Zone \*  
us-west4-b

La sélection d'une région est définitive

La zone est définitive

C'est la région et la zone dans laquelle la machine virtuelle sera déployée. Cela détermine l'emplacement physique où la machine virtuelle sera hébergée et exécutée. La région représente un emplacement géographique spécifique, tel que us-central1 (États-Unis, centre), europe-west1 (Europe, ouest), etc. La zone est une sous-région à l'intérieur de la région, par exemple us-central1-a, europe-west1-b, etc. Le choix de la région et de la zone dépend de la disponibilité des ressources, de la latence et des réglementations régionales.

### - Série :

#### Configuration de la machine

☒ Usage général ☐ Optimisé pour le calcul ☐ Mémoire optimisée

Types de machines pour les charges de travail courantes permettant d'optimiser les coûts et la flexibilité

Série  
N2

Ce paramètre fait référence à la famille de machines virtuelles disponible dans Google Cloud Platform. Chaque série est conçue pour répondre à des besoins spécifiques en termes de performances, de capacités de calcul et de coûts. Chaque série de machines virtuelles offre un ensemble de ressources matérielles et de spécifications différentes. Par exemple, la série E2 est optimisée pour les charges de travail générales, tandis que la série N2 est axée sur les charges de travail nécessitant une accélération matérielle pour les tâches de calcul intensif.

### - Type de machine :

#### Type de machine

Choisissez un type de machine avec des quantités prédéfinies de vCPU et de mémoire adaptées à la plupart des charges de travail. Vous pouvez aussi créer une machine personnalisée pour les besoins spécifiques de votre charge de travail. [En savoir plus](#)

PRÉDÉFINI PERSONNALISÉ

Filtrer les tailles des instances

Cœur partagé	n1-standard-1 1 processeurs virtuels, 3,75 Go de mémoire
Standard	n1-standard-2 2 processeurs virtuels, 7,5 Go de mémoire
Haute capacité de mémoire	n1-standard-4 4 processeurs virtuels, 15 Go de mémoire
Haute capacité de processeur	n1-standard-8 8 processeurs virtuels, 30 Go de mémoire



Les séries de machines, telles que les séries E, N, C, etc., regroupent des types de machines similaires en termes de performances et de caractéristiques. Ce paramètre détermine les ressources allouées à la machine virtuelle, telles que le nombre de CPU, la mémoire RAM, la capacité de stockage, etc.

Lorsque qu'on choisit un type de machine prédéfini, il nous donne une configuration par défaut qui est adaptée à la série de machines choisie. Cependant, on a la flexibilité de modifier ces paramètres selon nos besoins spécifiques. On peut augmenter ou réduire le nombre de cœurs de processeur virtuels (vCPU) et spécifier la quantité de mémoire (RAM) désirée en fonction de nos exigences de performance et de capacité.

Cela permet de créer des machines virtuelles personnalisées qui correspondent exactement aux besoins de l'application, en ajustant les ressources de calcul pour optimiser les performances et les coûts.

- **Service Confidential VMs :**



Ce paramètre permet de garantir un niveau élevé de confidentialité pour les charges de travail sensibles. Lorsque ce paramètre est activé, la machine virtuelle est exécutée sur un matériel spécialement configuré et sécurisé au sein des centres de données de Google Cloud. Cette configuration matérielle spécifique utilise des technologies de chiffrement et d'isolation pour protéger les données et les processus s'exécutant sur la machine virtuelle. L'utilisation des VMs confidentielles peut être appropriée pour les charges de travail nécessitant un niveau élevé de confidentialité, telles que le traitement de données sensibles ou réglementées, les applications sécurisées, les clés de chiffrement, etc. Notons que des coûts supplémentaires peuvent être appliqués si ce paramètre est activé.

- **Disque de démarrage :**

**Disque de démarrage**

Nom	instance-1
Type	Nouveau disque persistant avec équilibrage
Taille	10 Go
Type de licence	Gratuit
Image	Debian GNU/Linux 11 (bullseye)

MODIFIER

Disque de démarrage

Sélectionnez une image ou un instantané pour créer un disque de démarrage, ou associez un disque existant. Vous ne trouvez pas ce que vous recherchez ? Explorez des centaines de solutions de VM dans [Marketplace](#)

IMAGES PUBLIQUES

IMAGES PERSONNALISÉES

INSTANTANÉS

INSTANTANÉS D'ARCHIVE

DISQUES EXISTANTS

Système d'exploitation  
Debian

Version \*  
Debian GNU/Linux 11 (bullseye)

x86\_64, amd64 built on 20230615, supports Shielded VM features

Type de disque de démarrage \*  
Disque persistant avec équilibrage

COMPARER LES TYPES DE DISQUES

Taille (Go) \*  
10

SÉLECTIONNER

ANNULER

Ce paramètre fait référence au disque qui sera utilisé comme disque principal pour le système d'exploitation de la machine virtuelle. Un disque de démarrage contient le système d'exploitation et les fichiers nécessaires pour lancer la machine virtuelle.

GCP propose une large gamme de systèmes d'exploitation parmi lesquels on peut choisir, tels que Linux (par exemple, Ubuntu, Debian, CentOS) ou Windows Server. Chaque système d'exploitation a ses propres caractéristiques, fonctionnalités et exigences spécifiques.

Le système d'exploitation est le logiciel de base qui gère les ressources matérielles d'un ordinateur et fournit des fonctionnalités essentielles telles que la gestion des fichiers, des processus, des périphériques, etc. Il agit en tant qu'interface entre le matériel et les logiciels.

D'autre part, une distribution est une version spécifique d'un système d'exploitation qui est généralement personnalisée et préconfigurée avec certains logiciels, utilitaires et pilotes supplémentaires. Une distribution peut inclure des choix spécifiques de logiciels, des outils de gestion du package, une interface utilisateur spécifique, des configurations par défaut, etc.

Pour les systèmes d'exploitation Linux, on a le choix entre plusieurs distributions populaires telles que Ubuntu, Debian, CentOS, Red Hat Enterprise Linux (RHEL), SUSE Linux, etc. Chacune de ces distributions a différentes versions disponibles, nous permettant de choisir celle qui correspond le mieux à nos besoins. Notons que contrairement à Linux où il existe de nombreuses distributions populaires, le système d'exploitation Windows est généralement disponible en tant que version unique fournie par Microsoft mais on peut sélectionner différentes versions du système d'exploitation, telles que Windows Server, Windows 10, Windows 11, etc.

On peut aussi spécifier la taille du disque utilisé comme disque de démarrage pour la machine virtuelle. Plus la taille du disque de démarrage est grande, plus on dispose d'espace de stockage disponible pour installer des applications, des données et d'autres fichiers sur la machine virtuelle. Il est possible de redimensionner le disque de démarrage ultérieurement si nécessaire, mais cela peut nécessiter des étapes supplémentaires et une interruption temporaire de la machine virtuelle.

## - Identité et accès à l'API :

### Identité et accès à l'API ?

#### Comptes de service ?

Compte de service

Compute Engine default service account

▼

Vous devez définir le rôle "Utilisateur du compte de service" (roles/iam.serviceAccountUser) pour les utilisateurs qui souhaitent accéder aux VM avec ce compte de service. [En savoir plus](#)

#### Niveaux d'accès ?

- ☒ Autoriser l'accès par défaut
- ☐ Autoriser l'accès complet à l'ensemble des APIs Cloud
- ☐ Définir l'accès pour chaque API

ce paramètre concerne la gestion des autorisations et des rôles pour l'instance de la machine virtuelle. Lorsqu'on configure cette option, on peut définir les autorisations spécifiques accordées à l'instance et spécifier les rôles qui lui sont attribués.

On a trois options pour définir les niveaux d'accès :


**Autoriser l'accès par défaut :** Cette option accorde à l'instance de la machine virtuelle un accès limité par défaut aux ressources et aux API de Google Cloud Platform. Cela signifie que l'instance aura un ensemble restreint de privilèges par défaut et devra obtenir des autorisations supplémentaires pour accéder à des ressources spécifiques ou utiliser certaines API. Cette option est généralement utilisée si on souhaite limiter les privilèges de l'instance pour des raisons de sécurité.


**Autoriser l'accès complet à l'ensemble des APIs Cloud :** Cette option accorde à l'instance un accès complet à toutes les API de Google Cloud Platform. Cela signifie que l'instance aura des privilèges étendus pour interagir avec toutes les ressources et tous les services disponibles sur GCP. Cette option est souvent utilisée si on a besoin d'une instance avec des privilèges élevés pour effectuer des tâches d'administration ou accéder à des ressources sensibles.

**Définir l'accès pour chaque API :** Cette option nous permet de spécifier les autorisations pour chaque API individuellement. On peut définir des autorisations granulaires en choisissant les API spécifiques auxquelles l'instance aura accès et en configurant les privilèges correspondants. Cela donne un contrôle précis sur les autorisations accordées à l'instance pour chaque API.

## - Sécurité :

**Accès à la VM**  
Gérez la façon dont les utilisateurs se connectent à la VM

 Par défaut, lorsque vous vous connectez à une VM via cette console ou via gcloud, vos clés SSH sont générées automatiquement. [En savoir plus](#)

☐ Contrôler l'accès aux VM à l'aide des autorisations IAM   
Associez l'accès à la VM au rôle IAM de l'utilisateur. Cette opération active OS Login. [En savoir plus](#)

☐ Exiger la validation en deux étapes  
Exiger une deuxième authentification des utilisateurs [En savoir plus](#)

☐ Bloquer les clés SSH au niveau du projet  
Si vous cochez cette case, vous rendez cette instance inaccessible pour les clés SSH au niveau du projet. [En savoir plus](#)

**Ajouter des clés SSH générées manuellement**  
Ajoutez vos propres clés pour accéder à la VM via un outil tiers. Vous ne pouvez pas utiliser ces clés lorsque l'accès basé sur IAM (via OS Login) est activé. [En savoir plus](#)

[+ AJOUTER UN ÉLÉMENT](#)

Dans cette partie, on a la possibilité de gérer la façon dont les utilisateurs se connectent à la machine virtuelle. Par défaut, lorsqu'on se connecte à une machine virtuelle via la console GCP, nos clés SSH sont générées automatiquement. L'option d'ajouter des clés

SSH manuellement permet de fournir une flexibilité supplémentaire dans la gestion des clés d'authentification SSH. Par exemple, si on dispose déjà d'une paire de clés SSH qu'on utilise pour se connecter à d'autres systèmes ou machines, on peut les ajouter manuellement à la machine virtuelle pour une authentification cohérente. On peut aussi ajouter des clés ssh manuellement pour permettre à plusieurs utilisateurs d'accéder à la machine virtuelle. Seuls les utilisateurs dont les clés publiques ont été ajoutées à la liste des clés autorisées pourront accéder à la machine virtuelle.

Une fois qu'on a configuré les paramètres de la machine virtuelle, on peut cliquer sur « Créer » pour lancer le processus de création (GCP va vérifier les paramètres et allouer les ressources nécessaires pour la machine virtuelle).

il est possible de modifier les paramètres d'une machine virtuelle après l'avoir créée, comme la taille de la machine, le stockage, les options d'authentifications et d'accès,... Certains paramètres sont définis de manière permanente et ne peuvent pas être modifiés après la création de la machine virtuelle, comme le type de machine, le système d'exploitation, la région et la zone de déploiement,...

#### f) [Accès à une machine virtuelle sur GCP après sa création](#)

Après avoir créé une machine virtuelle sur GCP, on peut y accéder de différentes manières. Voici les méthodes courantes pour y accéder :

- **Connexion SSH** : c'est la méthode la plus couramment utilisée pour accéder à une machine virtuelle sur GCP. La connexion SSH (Secure Shell) est un protocole de communication sécurisé qui permet d'établir une connexion sécurisée entre deux machines, généralement un client et un serveur distant. Elle est couramment utilisée pour accéder à des machines à distance et les gérer de manière sécurisée. La connexion SSH utilise des paires de clés cryptographiques, composées d'une clé publique et d'une clé privée. La clé publique est placée sur le serveur distant, tandis que la clé privée est stockée sur notre client (ordinateur ou périphérique à partir duquel on se connecte au serveur distant). Lors de la connexion SSH, le client utilise sa clé privée pour s'authentifier auprès du serveur en prouvant qu'il possède la clé correspondante.

Voici les étapes à suivre :

1. On ouvre le terminal de notre machine pour générer la paire de clé ssh. Les distributions Linux disposent généralement d'un terminal natif (par exemple, Terminal sur Ubuntu) qui prend en charge les commandes de ligne de commande, y compris celles nécessaires pour la génération des clés SSH. Sur Windows, on peut par exemple utiliser Git Bash pour le faire. Git Bash est un émulateur de terminal pour Windows qui fournit une interface en ligne de commande (CLI) permettant d'utiliser Git et d'exécuter des commandes Unix. Il offre une expérience de ligne

de commande similaire à celle de Linux ou macOS, ce qui facilite l'utilisation de Git et d'autres outils en ligne de commande sur un environnement Windows.

2. On exécute la commande suivante pour générer la paire de clés SSH : `ssh-keygen -t rsa`
3. La commande `ssh-keygen` générera ensuite une paire de clés SSH : une clé privée (qui sera enregistrée dans le fichier spécifié) et une clé publique (qui aura l'extension `.pub` et sera enregistrée dans le même répertoire)  
Une fois la paire de clés générée, vous pouvez afficher la clé publique en exécutant la commande suivante :  
`cat ~/.ssh/id_rsa.pub`
4. On copie la clé publique affichée et on la colle dans la section « Clé SSH » de Compute Engine en ajoutant une clé ssh.

#### Clés SSH

Ces clés n'autorisent l'accès qu'à cette instance, contrairement aux clés SSH à l'échelle du projet. [En savoir plus](#)

☐ Bloquer les clés SSH au niveau du projet

Si vous cochez cette case, vous rendez cette instance inaccessible pour les clés SSH à niveau du projet. [En savoir plus](#)

Ci SSH 1 \*

ssh-rsa AA

Saisissez la clé SSH publique

Ci SSH 2 \*

5. Maintenant, la machine virtuelle est configurée pour accepter la connexion SSH en utilisant la paire de clés générée.
6. On peut maintenant se connecter à la machine virtuelle en utilisant le nom d'utilisateur, la clé privée et l'adresse IP de la machine virtuelle en tapant la commande : `ssh -i /chemin_vers_la_clé-privée/id_rsa utilisateur@adresse_IP`

Notons que l'adresse IP ici, c'est l'adresse IP externe de la machine virtuelle, celle qui est publique et visible depuis Internet. Elle est attribuée à la machine virtuelle par le fournisseur de services Internet (FSI) et permet à la machine virtuelle d'être accessible depuis Internet.

- **Console Google Cloud Platform** : On peut aussi accéder à la machine virtuelle directement depuis la Console GCP en utilisant le bouton "SSH" fourni. Il suffit de cliquer sur ce bouton pour qu'on soit connecté à la machine virtuelle.

#### Instances de VM

Filtre Saisissez le nom ou la valeur de la propriété

<input type="checkbox"/>	État	Nom ↑	Zone	Recommandations	Utilisé par
<input type="checkbox"/>	✓	<a href="#">Instance-1</a>	us-west4-b		



Quand on utilise le bouton SSH sur Compute Engine, une interface web ou une application de terminal spécifique s'ouvre directement dans le navigateur, permettant d'accéder à la machine virtuelle. Cette méthode est souvent plus conviviale pour les utilisateurs moins familiers avec les commandes SSH et le terminal. On n'a pas besoin de connaître l'adresse IP de la machine virtuelle ou de taper manuellement la commande SSH.

Avec cette méthode, GCP récupère une clé publique temporaire. Cette clé publique est ensuite utilisée pour établir une connexion SSH sécurisée avec la machine virtuelle.

Il est important de noter que les clés SSH générées de manière temporaire ne sont valables que pour la session SSH actuelle. Lorsqu'on se déconnecte de la machine virtuelle, ces clés sont supprimées et de nouvelles clés seront générées lors de la prochaine connexion. Si on souhaite une authentification plus permanente, il est recommandé d'ajouter manuellement une clé SSH à la machine virtuelle.

#### 4. Installation et configuration de l'environnement Python sur une machine virtuelle GCP

L'installation et la configuration de l'environnement Python sur une machine virtuelle GCP permettent de créer un environnement de développement dédié à Python, ce qui permet d'exécuter et de développer des applications Python spécifiques. En ayant un environnement Python bien configuré, on peut développer, tester et exécuter les applications Python en toute confiance et efficacité sur votre machine virtuelle. L'utilisation d'un environnement Python sur une machine virtuelle permet de tirer parti de la puissance et des ressources de la machine virtuelle, ce qui peut améliorer les performances et la capacité de traitement des applications Python, en particulier pour des tâches exigeantes en termes de calcul ou de traitement de données.

Voici donc les étapes à suivre pour installer et configurer l'environnement python sur la machine virtuelle :

1. On se connecte à la machine virtuelle en utilisant l'une des méthodes décrites précédemment.
2. On vérifie si python est déjà installé en exécutant la commande suivante dans le terminal : **python3 --version**
3. Si Python est déjà installé, la version sera affichée. Sinon, on doit procéder à l'installation. Pour installer Python, on exécute la commande suivante dans le terminal : **sudo apt install python3**
4. On peut maintenant exécuter du code Python en utilisant l'interpréteur interactif de Python par exemple. Pour se faire, il suffit de taper « python3 » dans le terminal.
5. Si on veut exécuter un script python, on utilise la commande : **python mon\_script.py**

6. On peut écrire du code python plus complexe en utilisant un éditeur de texte comme « nano » en tapant la commande nano. On appuie sur CTRL + O pour enregistrer le script.



## 5. Utilisation d'un planificateur pour exécuter un script Python sur une machine virtuelle

### g) Qu'est-ce qu'un planificateur ?

Un planificateur est un outil qui permet de programmer l'exécution de tâches de manière automatisée et régulière. Il permet de définir des horaires, des intervalles ou des conditions spécifiques pour exécuter des scripts ou des commandes.

### h) exécuter des scripts Python de manière planifiée

Pour exécuter un script Python de manière planifiée sur une machine virtuelle, on peut utiliser un planificateur de tâches tel que cron. Cron est un utilitaire de planification de tâches disponible dans les systèmes d'exploitation de type Unix, tels que Linux. Il permet d'exécuter des tâches automatiquement à des moments spécifiés, qu'il s'agisse d'exécuter des scripts, de lancer des commandes système, d'envoyer des emails, etc.

#### i. Création du crontab

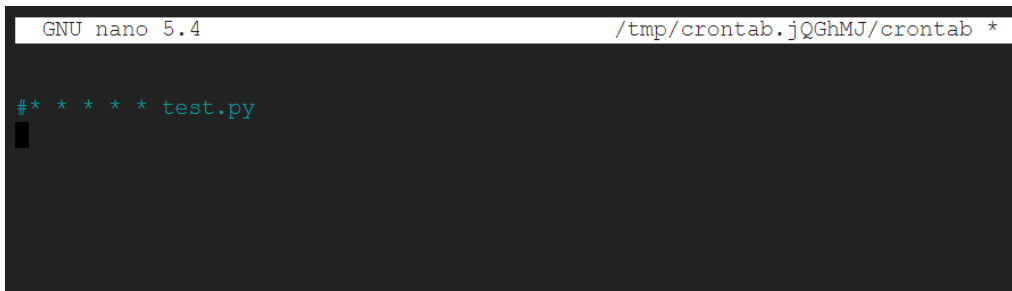
Le fonctionnement de Cron est basé sur des fichiers de configuration appelés "crontab" qui contiennent les instructions sur les tâches à exécuter et leur fréquence d'exécution (le crontab peut contenir plusieurs tâches). Chaque utilisateur peut avoir son propre fichier crontab pour définir ses propres tâches planifiées. Lorsqu'on exécute la commande **crontab -e**, cela ouvre le fichier de configuration de cron (crontab) associé à l'utilisateur dans un éditeur de texte. Les entrées dans un fichier crontab suivent une syntaxe spécifique, généralement composée de cinq champs (minutes, heures, jours du mois, mois, jours de la semaine), représentés par cinq astérisques suivis de la commande à exécuter. Par exemple, une ligne dans un fichier

crontab pourrait ressembler à ceci : **0 8 \* \* \* /chemin/vers/mon\_script.py**. Cela signifie que le script "mon\_script.py" sera exécuté tous les jours à 8h00.

#### ii. Suppression du crontab

Pour arrêter l'exécution des tâches planifiées dans le crontab, on utilise la commande : **crontab -r**.

Cela supprime toutes les tâches planifiées pour l'utilisateur actuel. Si on souhaite désactiver seulement certaines tâches, on peut utiliser la commande **crontab -e** pour éditer le crontab et commente en ajoutant un **#** au début de la ligne d'une tâche.



```
GNU nano 5.4 /tmp/crontab.jQGhMJ/crontab *
# * * * * * test.py
```

#### i) Options de planification disponibles sur GCP

Lors de l'utilisation d'un planificateur sur GCP pour exécuter un script Python, on a plusieurs options de planification disponibles :

- **Cron** : Cron est un outil de planification largement utilisé sur les systèmes Unix/Linux. On peut utiliser Cron pour définir des tâches récurrentes en utilisant une syntaxe spécifique pour la planification du temps. On peut utiliser Cron pour exécuter un script Python à des intervalles spécifiques, tels que toutes les minutes, toutes les heures, tous les jours, etc. C'est celle qu'on a développé précédemment.
- **Cloud Scheduler** : Cloud Scheduler est un service de planification entièrement géré fourni par Google Cloud. On peut configurer des tâches planifiées à l'aide d'une interface Web conviviale. On peut définir des tâches pour qu'elles s'exécutent à des moments spécifiques, à des intervalles réguliers ou en utilisant une expression cron.
- **Cloud Functions** : Cloud Functions est un service de calcul sans serveur sur GCP. On peut créer une fonction Cloud Function en utilisant Python et définir un déclencheur basé sur une planification. Par exemple, on peut configurer une fonction pour s'exécuter toutes les heures ou tous les jours.
- **Cloud Run** : Cloud Run est un service d'exécution de conteneurs entièrement géré sur GCP. On peut emballer un script Python dans un conteneur Docker, le déployer sur Cloud Run et configurer une planification pour exécuter le conteneur à des intervalles spécifiques.

Ces options de planification offrent une flexibilité pour exécuter des scripts Python sur GCP selon un calendrier spécifique ou à des intervalles réguliers, en fonction des besoins et de la complexité de l'application.



## 6. Vérification de l'exécution réussie d'un script Python sur une machine virtuelle

Pour vérifier si le script Python s'est exécuté avec succès sur une machine virtuelle GCP, on peut effectuer les actions suivantes :

- **Vérifier les journaux** : Les journaux (logs) du script peuvent fournir des informations sur son exécution. On peut consulter les journaux système ou les journaux d'application pour voir s'il y a des messages d'erreur ou des indicateurs de réussite.
- **Vérifier les sorties** : Si le script génère des sorties ou des résultats, on peut vérifier ces sorties pour s'assurer que le script s'est exécuté comme prévu. Par exemple, si le script crée un fichier de sortie, on peut vérifier si le fichier a été créé et si son contenu est conforme à ce qui était attendu. De même, si le script envoie un e-mail, on peut vérifier la boîte de réception spécifiée pour vous assurer que l'e-mail a été envoyé correctement.
- **Utiliser des notifications** : On peut configurer des notifications pour être averti lorsque notre script s'exécute ou rencontre des problèmes. Cela peut nous aider à surveiller l'exécution du script et à être informé des résultats ou des problèmes potentiels. Par exemple, on peut configurer des alertes par e-mail pour recevoir une notification chaque fois que notre script s'exécute avec succès ou rencontre une erreur. On peut également utiliser des services de notification en temps réel comme Google Cloud Pub/Sub pour recevoir des notifications push sur l'exécution du script.

## 7. Avantages de l'utilisation de Google Cloud Platform pour l'exécution de scripts Python planifiés

L'utilisation de GCP pour l'exécution de scripts Python planifiés offre plusieurs avantages par rapport à d'autres solutions :

- **Planification intégrée** : GCP propose des services de planification intégrés tels que Google Cloud Scheduler, qui permettent de programmer facilement l'exécution régulière des scripts Python. On peut spécifier des horaires précis, des intervalles de répétition et des déclencheurs personnalisés pour automatiser l'exécution de vos scripts selon votre planification.
- **Scalabilité élastique** : Avec GCP, on peut ajuster la taille et la capacité des ressources de calcul en fonction des besoins. Cela vous permet d'exécuter vos scripts Python planifiés de manière efficace, même lorsqu'ils nécessitent des ressources supplémentaires pour gérer une charge de travail plus importante.
- **Gestion centralisée** : GCP offre des outils de gestion centralisée qui permettent de configurer et de gérer facilement les scripts Python planifiés à partir d'une seule interface. On peut définir les horaires d'exécution, surveiller l'état des tâches planifiées et gérer les paramètres de configuration de manière centralisée.

- **Intégration avec d'autres services GCP** : On peut intégrer les scripts Python planifiés avec d'autres services GCP pour étendre leurs fonctionnalités. Par exemple, on peut stocker les résultats des scripts dans Google Cloud Storage, déclencher des actions supplémentaires en utilisant Google Cloud Pub/Sub, ou visualiser les statistiques d'exécution avec Google Cloud Monitoring.
- **Fiabilité et tolérance aux pannes** : GCP garantit la disponibilité et la fiabilité de ses services, ce qui est essentiel pour l'exécution planifiée des scripts Python. Les machines virtuelles et les services GCP sont conçus pour résister aux pannes matérielles et pour offrir une redondance des données, assurant ainsi une exécution stable et continue de vos scripts.

## 8. Principaux services de Google Cloud Platform utilisés en conjonction avec une machine virtuelle et un script Python

Lorsqu'on utilise une machine virtuelle sur GCP en conjonction avec un script Python, on peut tirer parti de plusieurs services pour améliorer les fonctionnalités et les performances de l'application. Voici quelques-uns des principaux services de GCP qu'on peut utiliser :

- **Cloud Storage** : Cloud Storage est un service de stockage objet dans le cloud qu'on peut utiliser pour stocker et récupérer les données utilisées par le script Python. On peut utiliser Cloud Storage pour stocker les fichiers d'entrée et de sortie, les données de référence, les fichiers de configuration, etc.
- **Cloud Pub/Sub** : Cloud Pub/Sub est un service de messagerie asynchrone qui permet la communication entre différentes parties de l'application. On peut utiliser Cloud Pub/Sub pour envoyer des messages et déclencher des actions entre le script Python et d'autres composants de l'infrastructure.
- **Cloud Functions** : Cloud Functions est un service de calcul sans serveur qui permet d'exécuter des scripts Python sous forme de fonctions événementielles. On peut utiliser Cloud Functions pour exécuter le script en réponse à des événements spécifiques, tels que l'ajout d'un fichier dans Cloud Storage ou la réception d'un message dans Cloud Pub/Sub.
- **Cloud Scheduler** : Comme on a déjà dit, Cloud Scheduler est un service de planification qui permet de programmer l'exécution planifiée du script Python. On peut définir des horaires précis, des intervalles de répétition et des déclencheurs personnalisés pour automatiser l'exécution du script.

## 9. Conclusion

En conclusion, ce rapport a détaillé le processus de création d'une machine virtuelle sur Google Cloud Platform (GCP) ainsi que l'exécution planifiée de scripts Python. En utilisant les fonctionnalités avancées de GCP, on peut bénéficier d'une solution puissante et complète pour l'exécution efficace de scripts Python planifiés.

L'utilisation de GCP offre plusieurs avantages, notamment une infrastructure hautement scalable, une disponibilité élevée, une sécurité renforcée et des services complémentaires intégrés tels que Cloud Storage, Cloud Pub/Sub, Cloud Functions ou encore Cloud Scheduler.

## 10. Bibliographie

- **[YouTube].** (2019). **Google Cloud: Creating a Compute Engine Virtual Machine** [Vidéo].  
[https://www.youtube.com/watch?v=BnzATg93QFY&list=PLgTTUdxnNfyQNKUTYyl-2CL\\_bXkfprTnk&index=2](https://www.youtube.com/watch?v=BnzATg93QFY&list=PLgTTUdxnNfyQNKUTYyl-2CL_bXkfprTnk&index=2)
- **Kloudle.** (2021). 5 Ways to Connect to Your GCP VM Instances Using SSH. Kloudle Academy. <https://kloudle.com/academy/5-ways-to-connect-to-your-gcp-vm-instances-using-ssh/>
- **ShevArezoBlog.** (2019). Héberger et exécuter un script Python sur Google Cloud Platform -Informatique générale. ShevArezoBlog. <https://shevarezo.fr/blog/heberger-et-executer-un-script-python-sur-google-cloud-platform-informatique-generale>
- **[Google Cloud Console].** (s.d.). Créer une instance de machine virtuelle [Page Web]. Google\_Cloud\_Console.  
<https://console.cloud.google.com/compute/instancesAdd?hl=fr&project=local-dialect-391412>
- **LeBigData.fr.** (s.d.). Google Cloud Platform : tout savoir sur la plateforme cloud de Google [Page Web]. LeBigData.fr. <https://www.lebigdata.fr/google-cloud-platform-tout-savoir>
- **Microsoft Azure Documentation.** (s.d.). Qu'est-ce que le cloud computing ?  
<https://azure.microsoft.com/fr-fr/resources/cloud-computing-dictionary/what-is-cloud-computing>