

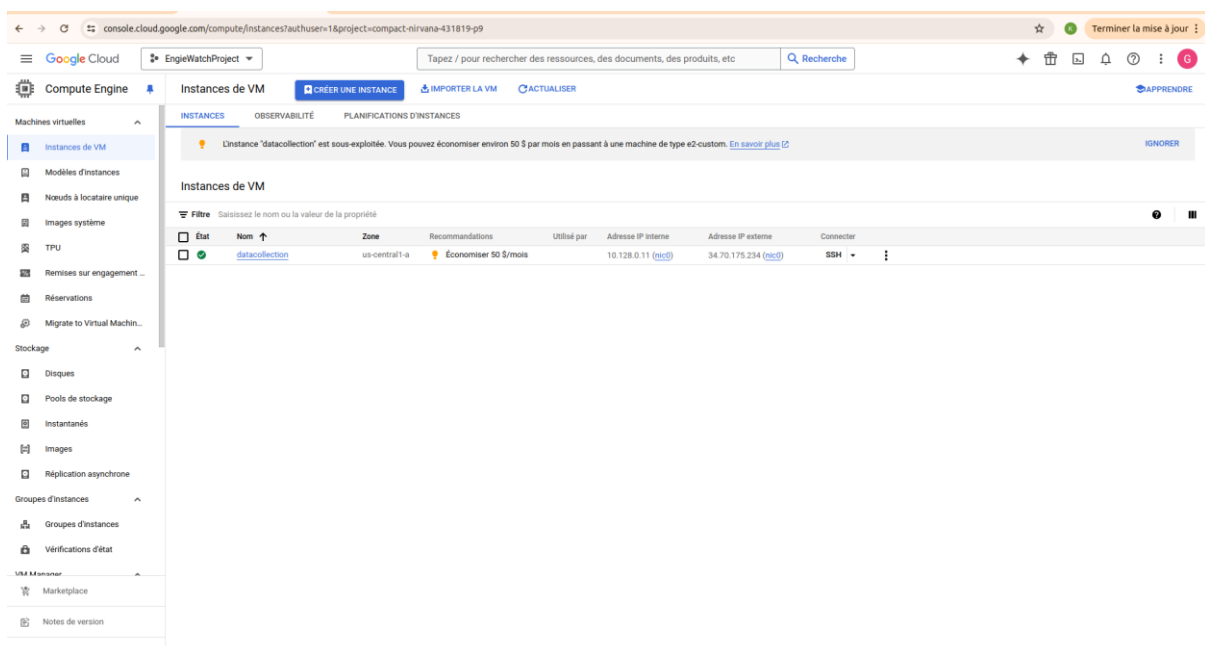
# Les etapes pour la mise en place du server Mlflow pour l'équipe

Nous avons opter pour une solution on-premise en utilisant GCP.

## Prerequis :

- **Un projet GCP configuré** avec un compte de facturation.
- **Une instance VM (Compute Engine)** configurée et fonctionnelle.
- **Un bucket GCS** où nos artefacts MLflow seront stocké.

## Etape 1 : Avoir un VM crée et active



## Etape 2 : Création d'un dossier Mlflow dans le bucket **engiestorgae** pour le projet engiewatch

- **Ajout d'autorisations IAM :**

Pour ce fait : Allez dans la section IAM & Admin dans la console GCP.

- Cliquez sur l'icône en forme de crayon à droite de ce compte de service pour modifier ses rôles.

**Autorisations pour le projet "EngieWatchProject"**

Ces autorisations concernent ce projet et toutes ses ressources. [En savoir plus](#)

☐ Inclure les attributions de rôles fournies par Google

**AFFICHER PAR COMPTE PRINCIPAL** **AFFICHER PAR RÔLE**

**+ ACCORDER L'ACCÈS** **- SUPPRIMER L'ACCÈS**

**Filtre** Saisissez le nom ou la valeur de la propriété

Type	Principal	Nom	Rôle	Insights sur la sécurité
<input type="checkbox"/>	267114771716-compute@developer.gserviceaccount.com	Compute Engine default service account	Éditeur	8759/8768 autorisations en excès
<input type="checkbox"/>	gbegrace2018@gmail.com	Grâce Gbe	Propriétaire	9795/9992 autorisations en excès

- Dans la section "Ajouter un autre rôle", recherchez et sélectionnez l'un des rôles suivants en fonction de vos besoins :
  - **Storage Object Admin** : Donne un accès complet à tous les objets dans les buckets. Idéal pour administrer les objets dans GCS.
  - **Storage Object Creator** : Permet de créer des objets dans les buckets. Idéal si vous ne voulez permettre que l'écriture d'objets, sans les droits de suppression ou de modification.

**Modifier l'accès à "EngieWatchProject"**

Compte principal: 267114771716-compute@developer.gserviceaccount.com

Projet: EngieWatchProject

**Attribuer des rôles**

Les rôles sont composés d'ensembles d'autorisations et déterminent ce que le compte principal peut faire avec cette ressource. [En savoir plus](#)

Rôle: Éditeur

Condition IAM (facultatif)

+ AJOUTER UNE CONDITION IAM

Rôle: Créateur d'objets Storage

Condition IAM (facultatif)

+ AJOUTER UNE CONDITION IAM

+ AJOUTER UN AUTRE RÔLE

**ENREGISTRER** **TESTER LES MODIFICATIONS** **ANNULER**

**Autorisations pour le projet "EngieWatchProject"**

Ces autorisations concernent ce projet et toutes ses ressources. [En savoir plus](#)

☐ Inclure les attributions de rôles fournies par Google

**AFFICHER PAR COMPTE PRINCIPAL** **AFFICHER PAR RÔLE**

**+ ACCORDER L'ACCÈS** **- SUPPRIMER L'ACCÈS**

**Filtre** Saisissez le nom ou la valeur de la propriété

Type	Principal	Nom	Rôle	Insights sur la sécurité
<input type="checkbox"/>	267114771716-compute@developer.gserviceaccount.com	Compute Engine default service account	Créateur d'objets Storage	8759/8768 autorisations en excès
<input type="checkbox"/>	gbegrace2018@gmail.com	Grâce Gbe	Propriétaire	9795/9992 autorisations en excès

Cela permettra à MLflow d'écrire des artefacts dans ce bucket.

Etape 3 : Connection à notre VM via SSH.

Notre vm s'appelle ***datacollection***, elle a déjà été créée par moi avant

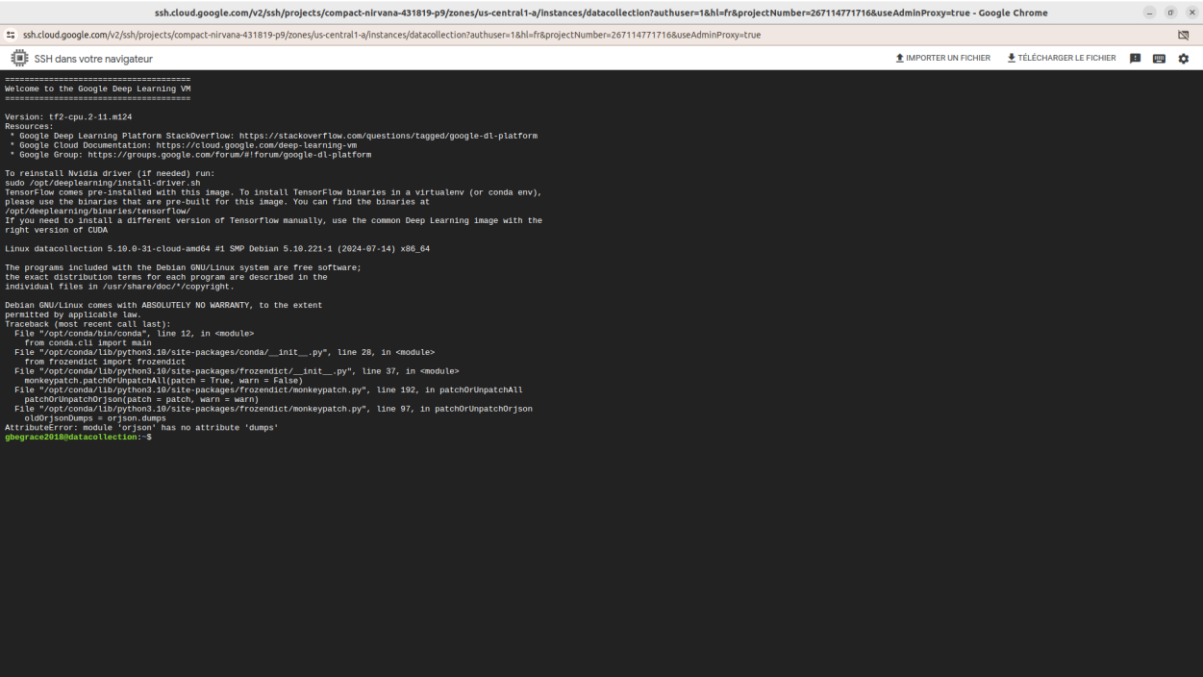
Instances de VM

Filtre

Saisissez le nom ou la valeur de la propriété

<input type="checkbox"/>	État	Nom <span>↑</span>	Zone	Recommandations	Utilisé par	Adresse IP interne	Adresse IP externe	Connecter
<input type="checkbox"/>		<a href="#">datacollection</a>	us-central1-a	Économiser 50 \$/mois		10.128.0.11 ( <a href="#">nic0</a> )	34.70.175.234 ( <a href="#">nic0</a> )	SSH <div></div>

Cliquez sur le bouton "SSH" à côté de la VM que vous avez choisie. Cela ouvrira une session SSH dans votre navigateur pour vous connecter directement à la VM.



Etape 4 : Créer une base de données SQL lite

Installer SQLite3:

sudo apt-get update

sudo apt-get install sqlite3

Voir la version : sqlite3 --version

Créer la base de donnée : sqlite3 mlflow.db

Suivez les étapes suivantes sur votre terminal :

**Installer les dépendances**

- Vérifier que Python et MLflow sont installés sur votre VM. Sinon
- Copier le code *sudo apt-get update*
- Puis *sudo apt-get install -y python3-pip*
- Enfin *pip3 install mlflow*

```

monkeypatch.patchOrUnpatchAll(patch = True, warn = False)
File "<opt/conda/lib/python3.10/site-packages/frozendict/monkeypatch.py", line 192, in patchOrUnpatchAll
  patchOrUnpatchOrJson(patch = patch, warn = warn)
File "<opt/conda/lib/python3.10/site-packages/frozendict/monkeypatch.py", line 97, in patchOrUnpatchOrJson
  oldToJsonDumps = orjson.dumps
AttributeError: module 'orjson' has no attribute 'dumps'
$ pip3 install mlflow
$ sudo apt-get update
$ sudo apt-get install -y python3-pip
Hit:1 https://packages.cloud.google.com/apt/gcpfuse-bullseye InRelease
Hit:2 https://download.docker.com/linux/debian/bullseye InRelease
Hit:3 https://nvidia.github.io/libnvidia-container/stable/debian10/amd64 InRelease
Hit:4 https://packages.cloud.google.com/apt/google-compute-engine-bullseye-stable InRelease
Hit:5 https://nvidia.github.io/nvidia-container-runtime/stable/debian10/amd64 InRelease
Hit:6 https://nvidia.github.io/nvidia-docker/debian10/amd64 InRelease
Hit:7 https://packages.cloud.google.com/apt/cloud-sdk-bullseye InRelease
Hit:8 https://packages.cloud.google.com/apt/google-fastsocket InRelease
Hit:9 https://deb.debian.org/debian/bullseye InRelease
Hit:10 https://deb.debian.org/debian-security/bullseye-security InRelease
Get:11 https://deb.debian.org/debian/bullseye-updates InRelease [44.1 kB]
Get:12 https://deb.debian.org/debian/bullseye-backports InRelease [49.0 kB]
Fetched 93.0 kB in 1s (65.8 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
python3-pip is already the newest version (20.3.4-4deb11u1).
0 upgraded, 0 newly installed, 0 to remove and 7 not upgraded.
Collecting mlflow
  Downloading mlflow-2.16.0-py3-none-any.whl (26.6 MB)
    26.6 MB 100.0 MB/s
Collecting gunicorn==24
  Downloading gunicorn-24.0.0-py3-none-any.whl (85 kB)
    85 kB 5.3 MB/s
Collecting docker>=4.0.0
  Downloading docker-7.1.0-py3-none-any.whl (147 kB)
    147 kB 67.2 MB/s
Collecting graphene==3.3
  Downloading graphene-3.3.0-py3-none-any.whl (128 kB)
    128 kB 86.0 MB/s
Collecting mlflow-skinny==2.16.0
  Downloading mlflow-skinny-2.16.0-py3-none-any.whl (5.6 MB)
    5.6 MB 50.1 MB/s
Collecting alembic==1.10.6
  Downloading alembic-1.10.6-py3-none-any.whl (232 kB)
    232 kB 88.0 MB/s
Collecting markdown==3.3
  Downloading markdown-3.3-py3-none-any.whl (106 kB)
    106 kB 63.1 MB/s
Collecting scipy==1.13.1
  Downloading scipy-1.13.1-cp39-cp39-manylinux_2_17_x86_64_manylinux2014_x86_64.whl (38.6 MB)
    38.6 MB 4.9 MB/s
Collecting pandas==2.2.2
  Downloading pandas-2.2.2-cp39-cp39-manylinux_2_17_x86_64_manylinux2014_x86_64.whl (13.1 MB)
    13.1 MB 56.4 MB/s
Collecting numpy==2.0.2
  Downloading numpy-2.0.2-cp39-cp39-manylinux_2_17_x86_64_manylinux2014_x86_64.whl (19.5 MB)
    19.5 MB 55.6 MB/s
Collecting matplotlib==3.9.2
  Downloading matplotlib-3.9.2-cp39-cp39-manylinux_2_17_x86_64_manylinux2014_x86_64.whl (8.3 MB)
    8.3 MB 53.5 MB/s
Collecting scikit-learn==1.5.1
  Downloading scikit-learn-1.5.1-cp39-cp39-manylinux_2_17_x86_64_manylinux2014_x86_64.whl (13.4 MB)
    13.4 MB 53.5 MB/s

```

## Création d'une section tmux :

*tmux new -s mlflow\_session*

N.B: Pour y entrer apres faire *tmux attach -t mlflow\_session*

```
ssh.cloud.google.com/v2/ssh/projects/compact-nirvana-431819-p9/zones/us-central1-a/instances/datacollection?authuser=1&hl=fr&projectNumber=267114771716&useAdminProxy=true - Google Chrome
ssh.cloud.google.com/v2/ssh/projects/compact-nirvana-431819-p9/zones/us-central1-a/instances/datacollection?authuser=1&hl=fr&projectNumber=267114771716&useAdminProxy=true
SSH dans votre navigateur
IMPORTER UN FICHIER
TÉLÉCHARGER LE FICHIER
Traceback (most recent call last):
  File "/opt/conda/bin/conda", line 12, in <module>
    from conda.cli import main
  File "/opt/conda/lib/python3.10/site-packages/conda/_init_.py", line 28, in <module>
    from frozendict import frozendict
  File "/opt/conda/lib/python3.10/site-packages/frozendict/_init_.py", line 37, in <module>
    monkeypatch.patchOrUnpatchAll(patch = True, warn = False)
  File "/opt/conda/lib/python3.10/site-packages/frozendict/monkeypatch.py", line 182, in patchOrUnpatchAll
    patchOrUnpatchOrjson(patch = patch, warn = warn)
  File "/opt/conda/lib/python3.10/site-packages/frozendict/monkeypatch.py", line 97, in patchOrUnpatchOrjson
    orjson.dumps = orjson.dumps
AttributeError: module 'orjson' has no attribute 'dumps'
gmg@cloudshell:~$
```

Nous sommes dans le tmux :

**Lancez le serveur MLflow** dans la session tmux :

```
python3 -m mlflow server \
--backend-store-uri sqlite:///mlflow.db \
--default-artifact-root gs://engiedatastorage/mlflow_experiment \
--host 0.0.0.0 --port 5000
```

Cela démarre le serveur MLflow. Il sera accessible sur le port 5000 de l'adresse IP publique de votre VM.

```
ssh.cloud.google.com/v2/ssh/projects/compact-nirvana-431819-p9/zones/us-central1-a/instances/datacollection?authuser=1&hl=fr&projectNumber=267114771716&useAdminProxy=true - Google Chrome
ssh.cloud.google.com/v2/ssh/projects/compact-nirvana-431819-p9/zones/us-central1-a/instances/datacollection?authuser=1&hl=fr&projectNumber=267114771716&useAdminProxy=true
SSH dans votre navigateur
Importer un fichier Télécharger le fichier
Traceback (most recent call last):
  File "/opt/conda/bin/conda", line 12, in <module>
    from conda.cli import main
  File "/opt/conda/lib/python3.10/site-packages/conda/_init_.py", line 28, in <module>
    from frozendict import frozendict
  File "/opt/conda/lib/python3.10/site-packages/frozendict/_init_.py", line 37, in <module>
    monkeypatch_patchOrUnpatchAll(patch = True, warn = False)
  File "/opt/conda/lib/python3.10/site-packages/frozendict/monkeypatch.py", line 182, in patchOrUnpatchAll
    patchOrUnpatchOrJson(patch = patch, warn = warn)
  File "/opt/conda/lib/python3.10/site-packages/frozendict/monkeypatch.py", line 97, in patchOrUnpatchOrJson
    objOrJsonDumps = orjson.dumps
AttributeError: module 'orjson' has no attribute 'dumps'
geograce@mlflowstore:~$ python3 -m mlflow server -l
--default-artifact-root: /mlflow
--backend-store-uri: sqlite:///mlflow.db
--host: 0.0.0.0 --port: 5000
2024/08/30 11:38:51 INFO mlflow.store.db.utils: Creating initial MLflow database tables...
2024/08/30 11:38:51 INFO mlflow.store.db.utils: Updating database tables
INFO [alembic.runtime.migration] Context impl SQLiteImpl.
INFO [alembic.runtime.migration] Will assume non-transactional DDL.
INFO [alembic.runtime.migration] Running upgrade 451aeb310d3 -> 451aeb310d3, add metric step
INFO [alembic.runtime.migration] Running upgrade 451aeb310d3 -> 96e94c40722, migrate user column to tags
INFO [alembic.runtime.migration] Running upgrade 96e94c40722 -> 181f10493468, allow nulls for metric values
INFO [alembic.runtime.migration] Running upgrade 181f10493468 -> df5be9277c5e, Add Experiment Tags Table
INFO [alembic.runtime.migration] Running upgrade df5be9277c5e -> 7ac759974a08, Update run tags with larger limit
INFO [alembic.runtime.migration] Running upgrade 7ac759974a08 -> 89d4b8295536, create latest metrics table
INFO [alembic.runtime.migration] Running upgrade 89d4b8295536 -> 2b4d017a5e9b, Migration completed
INFO [alembic.runtime.migration] Running upgrade 2b4d017a5e9b -> 2b4d017a5e9b, add model registry tables to db
INFO [alembic.runtime.migration] Running upgrade 2b4d017a5e9b -> cf6240cc0731, Add registered models and model_versions tables to database.
INFO [alembic.runtime.migration] Running upgrade cf6240cc0731 -> 6e6213401a0e, drop duplicate, killed constraint
INFO [alembic.runtime.migration] Running upgrade 6e6213401a0e -> 728d73865e8d, add registered model tags table
INFO [alembic.runtime.migration] Running upgrade 728d73865e8d -> 278da60202f1, add model version tags table
INFO [alembic.runtime.migration] Running upgrade 278da60202f1 -> 84291f48a231, add run link to model version
INFO [alembic.runtime.migration] Running upgrade 84291f48a231 -> a80ca7380d06, allow nulls for run_id
INFO [alembic.runtime.migration] Running upgrade a80ca7380d06 -> 39d1c3be5f05, add is_run_constraint_for_metrics_table_if_necessary
INFO [alembic.runtime.migration] Running upgrade 39d1c3be5f05 -> c48cb773b0b7, reset default value for is_run_in_metrics_table_for_mysql
INFO [alembic.runtime.migration] Running upgrade c48cb773b0b7 -> b0977e9b3c5, create index on run_uuid
INFO [alembic.runtime.migration] Running upgrade b0977e9b3c5 -> ac778090ac13, add deleted_time field to runs table
INFO [alembic.runtime.migration] Running upgrade ac778090ac13 -> cc1f7728345, change param value length to 500
INFO [alembic.runtime.migration] Running upgrade cc1f7728345 -> 9722a7f0f4d, Add creation time and last_update time to experiments table
INFO [alembic.runtime.migration] Running upgrade 9722a7f0f4d -> 3509859a5d39, Add Model Aliases table
INFO [alembic.runtime.migration] Running upgrade 3509859a5d39 -> 772a7d5fae7d, add datasets input tags tables
INFO [alembic.runtime.migration] Running upgrade 772a7d5fae7d -> 296e25a74d3e, increase max param val length from 500 to 8000
INFO [alembic.runtime.migration] Running upgrade 296e25a74d3e -> acf3f170cc7, add storage location field to model versions
INFO [alembic.runtime.migration] Running upgrade acf3f170cc7 -> 8d7405a0f9d4, add trace tables
INFO [alembic.runtime.migration] Running upgrade 8d7405a0f9d4 -> 5b0e3adcefc, add cascade deletion to trace tables foreign keys
INFO [alembic.runtime.migration] Running upgrade 5b0e3adcefc -> 4d0504737401, increase max dataset schema size
INFO [alembic.runtime.migration] Context impl SQLiteImpl.
INFO [alembic.runtime.migration] Will assume non-transactional DDL.
2024-08-30 11:38:52 +0000 [2904150] [INFO] Starting gunicorn 23.0.0
2024-08-30 11:38:52 +0000 [2904156] [INFO] Listening at: http://0.0.0.0:5000 (2904150)
2024-08-30 11:38:52 +0000 [2904156] [INFO] Using worker: sync
2024-08-30 11:38:52 +0000 [2904157] [INFO] Booting worker with pid: 2904157
2024-08-30 11:38:52 +0000 [2904161] [INFO] Booting worker with pid: 2904161
2024-08-30 11:38:52 +0000 [2904162] [INFO] Booting worker with pid: 2904162
2024-08-30 11:38:52 +0000 [2904163] [INFO] Booting worker with pid: 2904163
```

Détachez-vous de la session tmux sans arrêter MLflow :

Appuyez sur Ctrl + B, puis D. Cela vous ramènera à votre session SSH normale, tout en laissant MLflow en cours d'exécution dans le fond via tmux.

Vérifiez que MLflow fonctionne en visitant <http://<IP de la VM>:5000> dans un navigateur. Dans notre cas l'IP est : 34.70.175.234 qui est l'IP de ma VM *datacollection*

## A faire pour éviter un problème de port

Un problème arrivé était l'impossibilité d'accéder au site <http://34.70.175.234:5000>

Il faut donc faire les config pour que **les règles de pare-feu de Google Cloud** autorisent le trafic sur le port 5000.

Pour ce fait :

### Étape 1 : Accéder à "Pare-feu"

1. Dans la barre de recherche en haut, tapez simplement **"Pare-feu"**.
2. Vous devriez voir une option qui s'appelle **"Pare-feu"** sous la section **"Réseau VPC"**. Cliquez dessus.

### Étape 2 : Créer une nouvelle règle de pare-feu

3. Vous arriverez à la page des règles de pare-feu.
4. Cliquez sur le bouton **"Create firewall rule"** ou **"Créer une règle de pare-feu"** pour créer une nouvelle règle.

## Étape 3 : Configurer la règle de pare-feu

5. **Nom de la règle** : Donnez un nom comme allow-mlflow.
6. **Réseau** : Sélectionnez le réseau approprié, généralement le réseau par défaut.
7. **Priorité** : Laissez la priorité à 1000.
8. **Directions du trafic** : Choisissez "Ingress".
9. **Action sur la correspondance** : Sélectionnez "Allow".
10. **Cibles** : Sélectionnez "All instances in the network" (toutes les instances du réseau).
11. **Filtres source** : Sélectionnez "IP ranges" et entrez 0.0.0.0/0.
12. **Protocoles et ports** : Sélectionnez "Specified protocols and ports", puis "tcp" et entrez 5000.

## Étape 4 : Enregistrer la règle

13. Cliquez sur "Create" pour enregistrer la règle.

The screenshot displays the Google Cloud Firewall Manager interface. The main content area shows the 'Règles de pare-feu VPC' (VPC Firewall Rules) section. A table lists the configured rules:

Nom	Type	Cibles	Filtres	Protocoles/Ports	Action	Priorité	Réseau	Journaux	Nombre d'appels	Dernier a
allow-mlflow	Entrée	Appliquer	Plages	tcp:5000	Autoriser	1000	default	Désactivé	-	-
default-allow-icmp	Entrée	Appliquer	Plages	icmp	Autoriser	65534	default	Désactivé	-	-
default-allow-icmp	Entrée	Appliquer	Plages	tcp:0-65535 udp:0-65535 icmp	Autoriser	65534	default	Désactivé	-	-
default-allow-icmp	Entrée	Appliquer	Plages	tcp:3389	Autoriser	65534	default	Désactivé	-	-
default-allow-ssh	Entrée	Appliquer	Plages	tcp:22	Autoriser	65534	default	Désactivé	-	-

A notification at the bottom of the table states: "La règle de pare-feu 'allow-mlflow' a bien été créée." The right sidebar contains 'Recommandations personnalisées' (Personalized Recommendations) with links to various security guides, including 'Présentation de Cloud Next Generation Firewall', 'Types de stratégies de pare-feu', 'Créer une stratégie de pare-feu de réseau au niveau mondial', 'Configurer le service de prévention des intrusions', and 'Configurer l'inspection TLS'.

Une fois cette règle créée, essayez à nouveau d'accéder à votre serveur MLflow via <http://34.70.175.234:5000> dans votre navigateur. Cela devrait permettre à MLflow d'être accessible à partir de cette adresse IP.

## Expériences

Expériences de recherche

✓ Défaut

## Défaut

Donnez votre avis

Ajouter une description

Partager

Cours	Évaluation	Traces
metrics.mse < 1 et param.model = "tree"	Le temps créé	État : Actif
Ensembles de données		
Titre : Créé		
Colonnes		
Grouper par		
Nom de l'exécution	Créé	Ensemble de données
Durée	Source	Modèles



### Aucune course enregistrée

Aucune exécution n'a encore été enregistrée. Découvrez comment créer des exécutions d'entraînement de modèle ML dans cette expérience.

0 courses correspondantes