

# Mastère SIO



## Projet fil rouge IaaS-Cloud

Laetiscia Pirbay

03/04/2020

## Table des matières

1. Environnement de déploiement : .....	2
2. Rappel des objectifs .....	2
3. Installation serverless / redéploiement .....	2
3.1. Pré-requis environnement .....	2
3.2. Récupération du projet : .....	3
3.3. Installation / configuration.....	3
3.4. Utilisation .....	3
3.5. Résultat attendu .....	3
4. Archivage du fichier dans S3 .....	4
5. Suppression du fichier au bout d'un an – théorie .....	4

## 1. Environnement de déploiement :

Composant	Version
OS	FreeBSD freebsd 11.2-STABLE
Python	Python 3.7.6
Instance	t2.micro

## 2. Rappel des objectifs

Critère	Description
Base	Document descriptif de l'installation et utilisation (voir ci-dessous)
Fichiers de test	Type de fichiers supportés : <ul style="list-style-type: none"> <li>• Text</li> <li>• Csv</li> <li>• Image</li> </ul> Un fichier de test de chaque type est contenu dans le git
> 4 arguments variables retournés	En plus du contenu au format json, le script renvoie les 3 métadonnées suivantes : <ul style="list-style-type: none"> <li>• mimetype : type de fichier</li> <li>• name : nom du fichier avec extension</li> <li>• Size : taille du fichier en byte</li> </ul>
Serverless	Le service est disponible sur l'url : <a href="https://27gckgi4wk.execute-api.eu-west-1.amazonaws.com/dev/toto">https://27gckgi4wk.execute-api.eu-west-1.amazonaws.com/dev/toto</a>
Archivage du fichier dans S3	Les fichiers et leurs métadonnées associées sont archivés dans un bucket. Les fichiers peuvent être chargés depuis le bucket sur l'url du bucket/nom_du_fichier.json Eg : <a href="https://bucket-projet-fil-rouge.s3-eu-west-1.amazonaws.com/coverage.json">https://bucket-projet-fil-rouge.s3-eu-west-1.amazonaws.com/coverage.json</a>
Suppression du fichier au bout d'un an – théorie	La configuration est faite avec la GUI AWS mais n'a pas été testée en configurant des délais plus courts. Les étapes de configurations sont détaillées plus bas
Suppression du fichier au bout d'un an – pratique	Non testé
Utilisez un service d'AWS tel que Rekognition ou Transcribe pour enrichir le retour de votre API	Non livré

## 3. Installation serverless / redéploiement

### 3.1. Prérequis environnement

Composant	Version
OS	FreeBSD freebsd 11.2-STABLE
Python	Python 3.7.6

### 3.2. Récupération du projet :

Le projet est disponible sur github à l'adresse :

<https://github.com/djangit/projet-fil-rouge-iaas/tree/final>

Commande pour cloner la branche :

```
$ git clone -b final https://github.com/djangit/projet-fil-rouge-iaas.git
```

### 3.3. Installation / configuration

- Si besoin, installer le framework serverless :

```
$ sudo pkg install node
$ sudo pkg install npm
$ sudo npm install serverless -g
```

- Lancement

```
$ cd projet-fil-rouge-iaas/
$ . ./venv/bin/activate
(venv) $ sls deploy
```

### 3.4. Utilisation

- Endpoint :

<https://27gckgi4wk.execute-api.eu-west-1.amazonaws.com/dev/toto>

- Requête :

Fichier texte	curl -X POST "https://27gckgi4wk.execute-api.eu-west-1.amazonaws.com/dev/toto" -H "accept: application/json" -H "Content-Type: multipart/form-data" -F "file=@VocabulaireOntoPizza.txt"
Fichier csv	curl -X POST "https://27gckgi4wk.execute-api.eu-west-1.amazonaws.com/dev/toto" -H "accept: application/json" -H "Content-Type: multipart/form-data" -F "file=@ecole.csv"
Image	curl -X POST "https://27gckgi4wk.execute-api.eu-west-1.amazonaws.com/dev/toto" -H "accept: application/json" -H "Content-Type: multipart/form-data" -F "file=@coverage.jpg"

### 3.5. Résultat attendu

La réponse au format json comprend 2 parties :

- File Data : contenu du fichier
- File MetaData : métadonnées du fichier incluant son nom, sa taille et son mimetype

```
{ "File
Data": "attrs==19.3.0\n boto3==1.12.11\n botocore==1.15.11\n Click==7.0\n
docutils==0.15.2\n filetype==1.0.6\n flasgger==0.9.4\n Flask==1.1.1\n fla
sk-swagger-ui==3.25.0\n importlib-
metadata==1.6.0\n itsdangerous==1.1.0\n Jinja2==2.11.1\n jmespath==0.9.5
```

## PROJET FIL ROUGE - IAAS-CLOUD

```
\njsonschema==3.2.0\nMarkupSafe==1.1.1\nmistune==0.8.4\npyrsistent==0.16.0\npython-dateutil==2.8.1\npython-dotenv==0.11.0\nPyYAML==5.3\ns3transfer==0.3.3\nsix==1.14.0\nurllib3==1.21.1\nWerkzeug==1.0.0\nzip==3.1.0\n", "File\nMetadata": {"name": "requirements.txt", "size": 0, "type": "text/plain"}}\n(venv) $
```

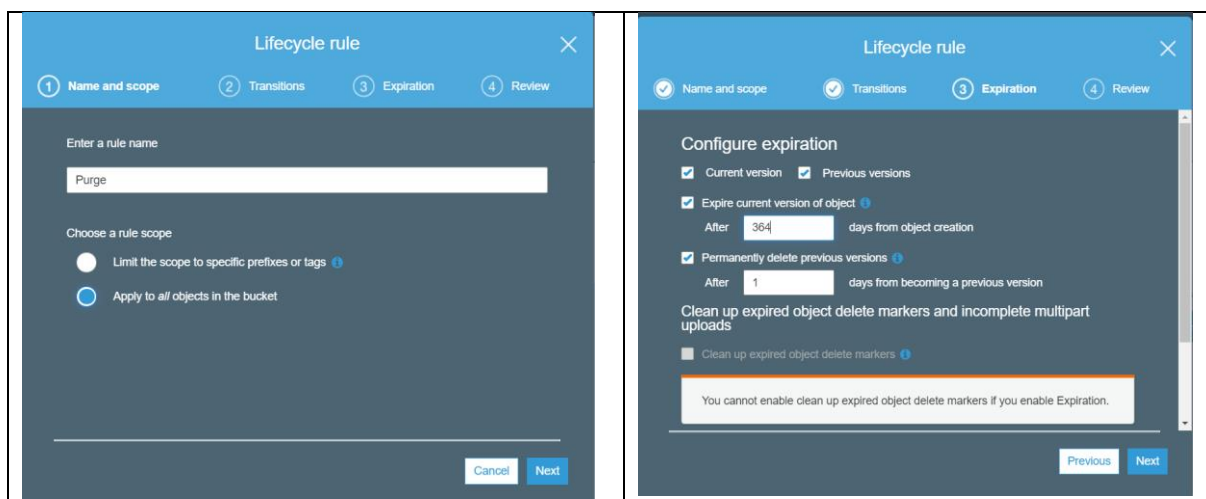
### 4. Archivage du fichier dans S3

Le fichier est sauvegardé en format json sur le bucket bucket-projet-fil-rouge

### 5. Suppression du fichier au bout d'un an – théorie

J'ai configuré la suppression automatique des fichiers du bucket avec la GUI AWS :

Sélectionner le bucket ->management-> lifecycle



Par manque de temps, je n'ai pas pu vérifier le bon fonctionnement de cette configuration (avec des délais plus courts).