**SAE 1.03 - INSTALLATION POSTE DE DÉVELOPPEMENT**

**SUJET 4**

Documentation technique

ROBERT Louan

GAUTIER Lowell

SABY Aubin

GLANDY Arthus

I2

**Sommaire**

Introduction 3

Avant de commencer 3

Fonctionnement 3

Utilisation 4

Mise en place 4

Mise en route 4

Arrêt 4

Détails des scripts Transfert.php, ExtractionDonnee.php, CreationGraph.sh 6

Transfert.php 6

Premier temps 6

Second temps 6

Troisième temps 6

ExtractionDonnee.php 7

Généralité 7

Pour les fichiers CSV 7

Pour les fichiers JSON 7

Pour les fichiers APACHE 7

Pour le tri 7

Pour le comptage 7

CreationGraph.sh 9

Premier temps 9

Second temps 9

Troisième temps 9

# Introduction

## Avant de commencer

Avant de lire cette documentation, assurez-vous d’avoir bien téléchargé les 7 scripts nécessaires au fonctionnement de l’application.

Les scripts sont

* Installation.bash
* Recuperation.bash
* Insertion.bash
* Desinstallation.bash
* CreationGraph.sh
* ExtractionDonnee.php
* TransfertFichier.php

Nous expliquons le rôle de chacun plus loin.

De plus, vous aurez besoin de Docker pour exécuter les scripts dans des conteneurs.

Assurez-vous d’avoir les droits d’exécution sur tous les scripts.

## Fonctionnement

Notre application sert à la création de graphiques à partir de fichiers de logs aux formats json, csv ou apache. Les autres extensions ne sont pas encore supportées.

Plus précisément, les fichiers de logs sont d’abord triés selon leurs extensions dans trois dossiers, CSV, JSON et APACHE.

Ensuite, lors d’une deuxième étape exécutée dans un second conteneur, les IP et les dates de connexions sont extraits de chaque fichier et écrits dans un fichier texte temporaire. Toujours dans le même conteneur, les données sont triées sur la date et le nombre de connexions par jour est inscrite dans un fichier de données définitif.

Finalement, dans un troisième conteneur, un graphique de connexion par jour est généré grâce au fichier de données définitif en utilisant Gnuplot.

# Utilisation

## Mise en place

Pour pouvoir commencer à utiliser l’application, exécutez le script Installation.bash. Ce script met à jour les images qui seront nécessaires aux conteneurs, créé un volume et copie les scripts dedans et finalement créé les conteneurs en exécutant les scripts à l‘intérieur.

Après avoir exécuté Installation.bash, un nouveau dossier apparaît dans votre répertoire de travail, Fichiers\_de\_travail. Collez tous vos fichiers de logs dans ce dossier.

ATTENTION :

Vos noms de fichiers ne doivent pas contenir de « . » hormis celui de l’extension.

Vous collerez vos fichiers sans les mettre dans un sous-dossier. Ils doivent tous être au même niveau de l’arborescence.

## Mise en route

Pour lancer le traitement de vos fichiers, exécutez le script Insertion.bash.

NB.

Pour observer le traitement de vos fichiers, utilisez trois terminaux différents dans lesquels vous taperez les commandes (une commande par terminal):

docker container logs -f Transmetteur

docker container logs -f Extracteur

docker container logs -f Gnuplot

Pour récupérer votre graphique, exécutez le script Recuperation.bash.

Un nouveau dossier apparaît, Graph. Dedans vous retrouverez votre graphique ainsi qu’un dossier qui contient des fichiers de traitement intermédiaires.

## Arrêt

Pour arrêter et désinstaller l’application, exécutez le script Desinstallation.bash. Ce script arrête et supprime les conteneurs qui ont été créés et lancés. Il supprime aussi le volume créé.

NB.

Si vous voulez générer un nouveau graphique avec de nouveaux fichiers, vous devez d’abord exécuter le script Desinstallation.bash puis Installation.bash. Remplacez ensuite le contenu du dossier Fichiers\_de\_travail et reprenez à l’étape « Mise en route ».

# Détails des scripts Transfert.php, ExtractionDonnee.php, CreationGraph.sh

## Transfert.php

### Premier temps

Dans un premier temps, le script créé les dossiers Separation, Separation/CSV, Separation/APACHE et Separation/JSON qui correspondent aux extensions des fichiers supportés.

### Second temps

Dans un second temps, le script transfère les fichiers dans les dossiers correspondant à leurs extensions.

### Troisième temps

Finalement, pour éviter de traiter plusieurs fois les mêmes données, le script renomme le dossier Fichiers\_de\_travail en finC1.

## ExtractionDonnee.php

(C’est là que le grand bricolage commence)

### Généralité

Le but de ce script est la récupération des IP et des dates de connexions, le tri de ces informations et le comptage de connexions par jour.

A la fin du script, le dossier finC1 est renommé en finC2 pour éviter le traitement des mêmes fichiers.

### Pour les fichiers CSV

Nous avons traité les fichiers CSV ligne par ligne grâce à des structures de type foreach. Cela nous a permis de diviser les lignes en plusieurs sections.

Nous avons ensuite sélectionné les sections contenant les informations sur les IP et la date de connexions que nous avons formaté avant de les écrire dans un fichier de données temporaire.

### Pour les fichiers JSON

Nous avons décodé les fichiers JSON a l’aide de jq en un array contenant 2 arrays contenant les connexions. Cela nous a permis de facilement sélectionner chaque objet du JSON. Ensuite, nous avons sélectionné les objets « date » et « ip ». Le format de la date séparait jour, mois, année par des / et précisait l’heure et l’horaire, alors nous avons enlevé la précision et remplacé les / par des -.

### Pour les fichiers APACHE

### Pour le tri

Concernant le tri des données sur la date de connexion, nous avons utilisé un tri à bulles. Les données sont toujours écrites sur un fichier de données temporaire.

### Pour le comptage

Finalement, nous avons pris la décision de compter le nombre d’occurrences de chaque date.

Nous avions à l’origine prévu de compter chaque connexion d’utilisateurs différents (d’où la nécessité des IP) mais le peu de données disponibles nous ont paru insuffisantes.

Les données écrites sur le fichier de données définitif sont sous le format suivant

Numéro\_de\_ligne nombre\_de\_connexions date

Exemple :

1 4 19-01-2020

2 5 20-01-2020

3 3 21-01-2020

## CreationGraph.sh

### Premier temps

Dans un premier temps, le script créé un fichier de configuration gnuScript qui contient toutes les informations de configuration nécessaires à Gnuplot.

### Second temps

Le script appelle la commande gnuplot et fournie en paramètre le fichier de configuration gnuScript. Le graphique est généré.

### Troisième temps

Le script supprime le fichier de configuration et renomme le dossier finC2 en Graph.