Fournier Jérémy

Documentation agrégateur RSS

SAE Mettre en place une solution informatique

IUT Réseaux et Télécommunications
2021-2022

Sommaire

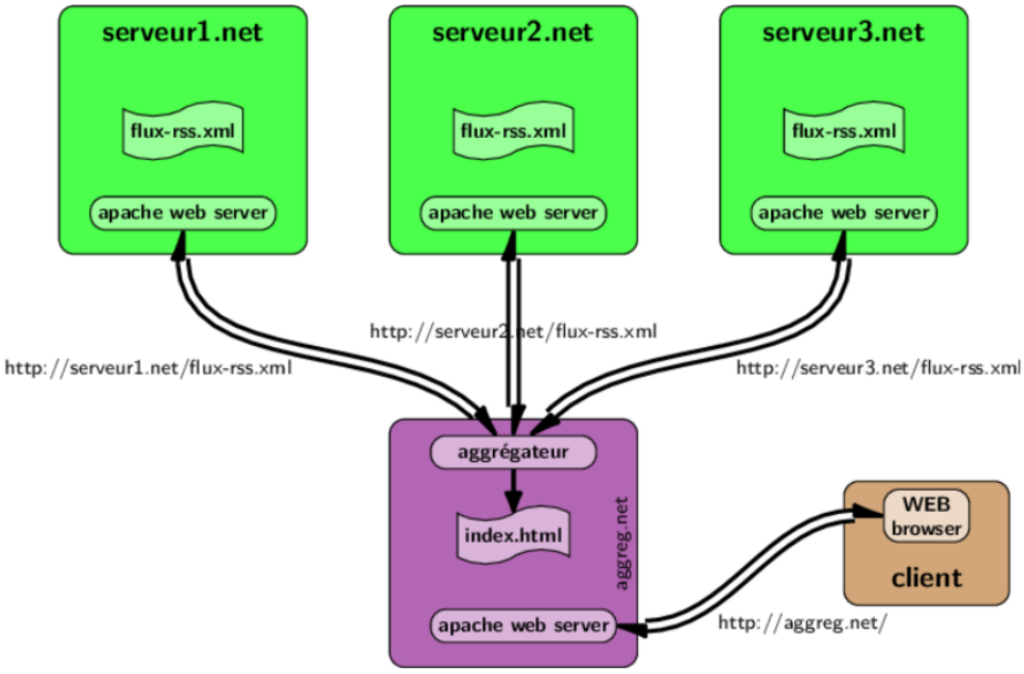
1)Contexte et environnement de test

Présentation : Le but de cette SAE est de présenter un site WEB sur une machine ayant traité et mis en forme au préalable les flux RSS d'un certain nombre de serveurs donnés

Contexte :
Afin de produire un programme fonctionnant dans n'importe quel environnement, nous avons mis en place un environnement classique avec plusieurs serveurs où seront stockés les flux RSS, une machine aggreg où le programme principal s'exécutera avec un fichier de configuration où l'on renseignera un certain nombre d'information

Nous avons besoin:
-3 serveurs avec un générateur de flux RSS "simu-site" installé avec l'utilisateur "simu"
-1 agrégateur avec "aggreg.py" et un fichier de configuration "conf.yaml" avec l'utilisateur "aggreg"
-Un client avec interface graphique

Schéma explicatif :



Explication : Les 3 serveurs génèrent de façon automatique des flux RSS avec un nom précis dans un emplacement précis accessible depuis l'extérieur via un URL. La machine aggreg va récupérer ces flux RSS les traiter et produire en sortie deux fichiers :index.html et feed.css où dans index.html on retrouve les flux triés et mis en forme. Le client quant à lui est juste d'accéder au serveur web de agreg et observer le rendu final

2)Fichier de configuration conf.yaml

Il s'agit du fichier racine du projet où le programme va venir utiliser ses différents renseignements.

Voici sa syntaxe :

```
destination: /var/www/aggreg/index.html
rss-name: rss.xml
sources:
- http://serveur1.net/
- http://serveur2.net/
- http://serveur3.net/
tri-chrono: True
```

-Destination : chemin où la page web sera généré, le fichier css est fait pour être créer à côté de lui
-rss-name : indique le nom du flux RSS généré, selon votre commande simu-site il peut changer, dans notre cas il restera rss.xml
-sources: URL d'accès aux serveurs où l'on pourra trouver le rss-name, il y a autant de sources que de serveurs
-Tri-chrono : Indique si les événements seront triés de façon chronologique ou non sur la page (True pour oui , False pour non)

Tous les champs sont obligatoires et seul le "sources" peut prendre plusieurs éléments.

Son emplacement peut être variable, soit directement à côté du programme principal ou dans un répertoire de configuration du type /etc. Il suffit juste de modifier une ligne dans le programme afin d'indiquer son emplacement précis.

3)Programme aggreg.py et son fonctionnement

Il y a 4 fonctions principales dans notre programme :
-**check_url** qui vérifie si un URL est accessible grâce au module requests
-**charge_urls** qui renvoie un flux RSS chargé à partir d'URLs
-**fusion_flux** qui renvoie une liste d'événements triés à partir des flux RSS
-**genere_html** qui renvoie deux fichiers : index.html et feed.css avec les événements intégrés et mis en page de façon responsive et ainsi accessible via un navigateur. L'emplacement d'index.html dépend du fichier conf.yaml et feed.css est fait pour se créer automatiquement à côté de lui
-**conf_yaml** qui lis le fichier de paramètres qui défini les chemins utiles aux fonctions.
-**main** où l'on va lancer toutes nos fonctions en ayant définis les options nécessaires du fichier config.yaml

Une fois ce programme exécuté, il génère deux fichiers en sortie sur votre machine aggreg:
-**index.html** dans l'emplacement indiqué dans le conf.yaml dans destination qui est la page web avec les flux triés par le programme
-**feed.css** dans l'emplacement indiqué dans le conf.yaml qui est le fichier de style de index.html améliorant la visibilité en divisant la page en 3 colonnes responsives. A gauche les événements "MINOR", au milieu "MAJOR" et à droite "CRITICAL"

❗ Le programme fonctionne uniquement à partir du fichier conf.yaml, sans lui rien ne marche. Il faut donc vérifier deux choses au préalable pour que tout fonctionne correctement :
-Bien avoir le fichier "conf.yaml" avec la bonne syntaxe

```
destination: /var/www/aggreg/index.html
rss-name: rss.xml
sources:
- http://serveur1.net/
- http://serveur2.net/
- http://serveur3.net/
tri-chrono: True
```

-Bien le placer dans le répertoire renseigné dans la fonction conf_yaml, si vous voulez le placer dans un autre emplacement, il faut impérativement changer une ligne de code dans la fonction "conf_yaml" en renseignant le bon chemin absolu du fichier.

```
with open("/home/aggreg/conf.yaml","r") as conf:
```

❗ Pour plus d'interrogation, le programme est entièrement commenté et décrit chaque étape d'exécution sur mon etulab :
https://etulab.univ-amu.fr/f21202056/sae203

4)Améliorations possibles et limites du programme

Notre programme fonctionne s' il est bien paramétré au préalable, mais on peut vite voir certaines faiblesses que l'on pourrait optimiser ou même créer, par exemple :

-Si différents serveurs possèdent des noms de flux RSS différents qu'ils puissent les renseigner sans faire tout crasher. Pour cela, un système de liste comme pour les sources peut être mis en place où le programme peut déterminer quel fichier avec le bon nom est présent sur chaque serveur.

-Un petit graphique sur la page HTML avec le nombre de catégories d'événements afin de bien visualiser l'état général. Il suffirait de compter chaque catégorie présente et d'en faire une moyenne pour chaque. De nombreux modules permettent d'en générer sous python.

-Faire en sorte que le programme soit capable de lire le fichier /etc/hosts pour voir quels sont les serveurs enregistrés et qu'il les ajoute dans le champ sources de conf.yaml. Après reste à voir comment blacklister certaines machines qui ne souhaitent pas se retrouver dedans.

-Si le flux RSS n'est pas complet, qu'il puisse ignorer certaines options qu'il cherche de base afin de visualiser quand même les événements s' il manque des choses