2 - Serveur AGIMUS-NG

Si vous utilisez actuellement la version 2 d'Agimus-NG, vous pouvez utiliser la page Migration des données issues de la v2 vers la v7 pour vous aider à migrer vos données

Installations préalables

Agimus-NG utilise Elasticsearch, Logstash. Pour en savoir plus sur l'installation, consulter la page Installations requises sur le serveur Agimus-NG ou cliquez sur le bouton ci-dessous.

approve ÉTAPE PRÉALABLE : Installation ELK

Cette documentation donne des exemples de configuration qui permettent d'enrichir les logs applicatifs grâce aux attributs LDAP suivants :

```
'eduPersonPrimaryAffiliation', 'supannEntiteAffectationPrincipale', 'supannEtuInscription', 'supannEtuSectEurDisciplinaire'
```

Vous êtes bien entendu libre de le modifier/remplacer/supprimer. Il suffit d'effectuer la modification dans chacun des fichiers de paramétrage décrit ci-dessous.

Mise en place Agimus-NG

Pour mettre en place Agimus-NG, il faut :

- récupérer les sources du projet sur github
- paramétrer les fichiers suivants (vous pouvez paramétrer au fur et à mesure de vos besoins) :
  - copie du fichier config-sample.py en config.py. Utilisé dans les traitements python
  - daily_batch.sh à lancer quotidien par cron pour effectuer les traitements
  - check_plugin_ldap.sh changer l'adresse de contact pour être prévenu en cas de mise à jour de logstash
  - traitements spécifiques à certains logs : scripts/moodle/*.py, scripts/traitement-ez*.sh, scripts/cron_stats_nc.sh, scripts/rocketchat-stats.sh
  - Configurations logstash appelant le ldap ou des bases externes
    - modulesBase/inputs/LDAP
    - gen_mappings/aff/0_LDAP
    - gen_mappings/vet-niveau/0_LDAP
    - gen_mappings/moodle_categories/0_jdbc
    - gen_mappings/vdi/0_LDAP
    - gen_mappings/vet/0_LDAP
    - gen_mappings/moodle_users/0_jdbc
    - moodle-from-db/0_jdbc
  - Copie du fichier frontal/config/config-sample.php en config.php pour le paramétrage du frontal
- Rapatrier les logs à traiter en vous aidant de la documentation ci-dessous
- Traiter ces logs grâce aux configurations logstash fournies que vous pourrez adapter
- Créer vos graphiques dans kibana pour visualiser les données générées

Rapatriement des logs

Pour lancer le traitement, il faut préalablement rapatrier les logs à traiter sur le serveur Agimus-NG

approve ÉTAPE SUIVANTE : Rapatrier les logs à traiter

Script quotidien

Vos données de log vont être enrichies et enregistrées dans elasticsearch grâce au script de traitement quotidien **daily_batch.sh**

Vous avez commencé à le paramétrer à l'étape précédente. Chaque traitement est indiqué dans les commentaires du fichier de script. Modifiez le suivant les besoins de votre établissement pour supprimer/ajouter les blocs correspondants aux traitements souhaités.

Nous allons voir et tester ci-dessous quelques traitements fournis par défaut dans le fichier.

Le fichier daily_batch.sh utilise une variable CONF_PATH qui est par défaut égale à $BUILD_HOME"/logstash/". Si vous lancez des traitements en dehors de ce batch, définissez préalablement la valeur de CONF_PATH :

```
export CONF_PATH="/opt/agimus/logstash"
```

Vous pouvez également remplacer la valeur par défaut (/tmp) dans les fichiers de configuration du répertoire logstash :

Remplacer /opt/agimus-ng/logstash par le chemin de votre répertoire logstash

```
cd /opt/agimus-ng/logstash
grep -rEl "CONF_PATH:'?/tmp" | xargs sed -E -i "s#CONF_PATH:'?/tmp'?#CONF_PATH:'/opt/agimus-ng
/logstash'#"
```

Traitement du LDAP

Les informations qui vont enrichir les logs sont extraites quotidiennement du ldap et intégrer à Elasticsearch. A cette étape, vous devriez avoir déjà vérifié le bon fonctionnement de la récupération des informations ldap avec logstash en utilisant le fichier test-logstash.conf.

Le script de traitement quotidien **daily_batch.sh** contient les commandes ci-dessous pour :

- supprimer des données de l'index ldap dans elasticsearch
- récupérer des nouvelles données du ldap
- enregistrer des statistiques ldap dans l'index ldap-stat

POUR TESTER

```
curl -XDELETE 'http://localhost:9200/ldap/' # A ne pas faire la première fois car cette commande
supprime l'index ldap
export CONF_PATH="./logstash"; scripts/logstash -f logstash/import-ldap     # import des données
définies dans logstash/modulesBase/inputs/LDAP
python scripts/ldap-agg.py      # Génération de l'index ldap-stat qui décompte les populations par
regroupement d'attributs
```

Ces commandes vont générer dans votre elasticsearch :

- un index ldap, contenant l'ensemble de vos fiches avec les attributs traités
- un index ldap-stat qui comptabilisent les différentes valeurs ainsi que leur nombre d'occurrence pour chacun des attributs traités

Pour aller plus loin

Nom des attributs issus du ldap

Les attributs issus du ldap seront utilisables avec une casse en minuscule dans logstash même si le nom a une autre casse dans ldap.

Par exemple, `supannEntiteAffectationPrincipale` deviendra `supannentiteaffectationprincipale`

Traitement du fichier trace.log issu du serveur CAS

Le contenu du fichier trace.log est enregistré dans elasticsearch dans l'index trace qui sera interrogé lors du traitement des logs afin de faire le lien entre un log et l'utilisateur l'ayant généré.

POUR TESTER

Il faut lancer la commande suivante pour traiter le fichier trace.log rapatrié dans le répertoire /data/in/*date_du_jour*

```
logstash -f logstash/trace < /data/in/$date/trace.log
```

Cette commande va générer dans votre elasticsearch un index trace, contenant l'ensemble des associations login<->trace

Traitement des logs issus du CAS

Vous pouvez tester l'action de traitement qui est faite dans le script quotidien :

Il faut lancer la commande suivante pour traiter les logs esup rapatriés dans le répertoire /data/in/*date_du_jour*

```
zcat /data/in/$date/serviceStats.log.gz | logstash -f logstash/casRequest
```

Cette commande va générer dans votre elasticsearch un index ag-casrequest-YYYY.MM contenant les logs contenant les appels à votre CAS. Ces documents (chaque document est une ligne de log) contiendront les informations supplémentaires issues de votre ldap.

Autres traitements

Il existe plusieurs autres exemples dans le fichier de traitement quotidien ainsi que des exemples de configuration dans le dossier logstash. Inspirez-vous en et n'hésitez pas à poser vos questions sur la liste de diffusion esup-utilisateurs

Points d'attention

Certains traitements ont des pré-requis. Vérifiez les points suivants si vous rencontrez des soucis lors de l'import de vos données :

- Avez-vous paramétré correctement la variable d'environnement CONF_PATH ? (cf la remarque de la section script_quotidien)
- Avez-vous généré ou créé les fichiers de mappings nécessaires à l'enrichissement des données ? Principalement utilisés par moodle, il vous faudra, au choix :
  - lancer les enrichissements du dossier gen_mappings pour générer les fichiers ou
  - désactiver l'enrichissement en commentant tout le bloc *translate {…}* de votre fichier de traitement

Traitement quotidien dans la crontab

Afin d'alimenter ElasticSearch quotidiennement, il faut lancer le script daily_batch.sh chaque jour, une fois les logs de la veille rapatriés. Pour cela, ajouter l'appel au script dans la crontab de l'utilisateur exécutant le traitement.

```
10 0 * * * /opt/agimus-ng/scripts/daily_batch.sh | logger -t AGIMUS
```

approve ÉTAPE SUIVANTE : Visualisation des résultats