INSTALLATION ET CONFIGURATION DE LA STACK ELK

Introduction

La stack ELK peut être **installée à l'aide d'une variété de méthodes** et sur un **large éventail de systèmes d'exploitation** et d'**environnements différents**. Vous pouvez **installer ELK localement**, sur le cloud, à l'aide de Docker et de systèmes de gestion de configuration comme Ansible, Puppet et Chef. La pile peut être également installée à l'aide de votre gestionnaire de paquets ou manuellement depuis les binaires officiels. Voici le lien pour la page officielle des multiples méthodes d'installation d'ELK.

De nombreuses étapes d'installation sont similaires d'un environnement à l'autre et comme nous ne pouvons pas couvrir tous les différents scénarios, je vais vous fournir un exemple d' **installation de tous les composants de la pile Elasticsearch, Logstash, Kibana sous une seule machine Linux** à l'aide du gestionnaire de paquets APT et UUM afin de posséder et de gérer les dernières versions des composants.

> **Information**
>
> Pour ceux travaillant sur une machine Windows, veuillez créer une machine virtuelle afin de poursuivre du mieux ce cours.

À savoir

Lors de l'installation d'ELK, vous devez utiliser la même version sur l'ensemble de la pile. Par exemple, si vous utilisez Elasticsearch 7.8.0 alors Kibana doit être aussi sous sa version 7.8.0 et même pour pour Logstash en version 7.8.0.

Si vous mettez à niveau une installation existante, consultez la page officielle de Mise à niveau de la pile ELK.

Ordre d'installation

Pour installer les produits ELK il est recommandé de respecter l'ordre suivant afin de garantir que les composants dont chaque produit dépend sont résolus :

1. Elasticsearch
2. Kibana
3. Logstash

Installation

Elasticsearch Installation

Tout d'abord, vous devez ajouter la clé de signature d'Elastic pour que le package téléchargé puisse être vérifié (ignorez cette étape si vous avez déjà installé des packages d'Elastic):

Sous la famille debian :

```
wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -
```

Sous la famille redhat :

```
rpm --import https://artifacts.elastic.co/GPG-KEY-elasticsearch
```

Si vous êtes sur une machine de la famille de Debian, vous devez installer le paquet `apt-transport-https` :

```
sudo apt-get install apt-transport-https
```

L'étape suivante consiste à ajouter le dépôt Elasticsearch sur votre système :

Sous la famille debian :

```
echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee -a /e
```

Sous la famille redhat, créez un fichier et nommé le par exemple `elasticsearch.repo` dans le répertoire `/etc/yum.repos.d/`, contenant:

```
[elasticsearch]
name=Elasticsearch repository for 7.x packages
baseurl=https://artifacts.elastic.co/packages/7.x/yum
gpgcheck=1
gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
enabled=0
autorefresh=1
type=rpm-md
```

Il ne vous reste plus qu'à mettre à jour vos référentiels et installer Elasticsearch:

Sous la famille debian :

```
sudo apt-get update -y && sudo apt-get install elasticsearch
```

Sous la famille redhat :

```
sudo yum install --enablerepo=elasticsearch elasticsearch
```

Configuration

Par défaut à son lancement Elasticsearch consomme 1go de mémoire de la JVM (machine virtuelle java), si votre machine n'est pas assez puissante vous pouvez modifier les valeurs `Xms` et `Xmx` situé dans le fichier **/etc/elasticsearch/jvm.options** pour une consommation réduite:

```
# Avant (1go)
-Xms1g
-Xmx1g

# Après (512 mo)
-Xms512mo
-Xmx512mo
```

Les configurations Elasticsearch sont effectuées à l'aide du fichier de configuration **/etc/elasticsearch/elasticsearch.yml** qui vous permet de configurer les paramètres généraux comme par exemple le nom du nœud, ainsi que les paramètres réseau comme par exemple l'hôte et le port, l'emplacement des données stockées, la mémoire, les fichiers de logs, etc... Pour ce cours nous laisserons la configuration par défaut.

Lancement et test

Pour exécuter Elasticsearch, utilisez la commande suivante (l'initialisation peut prendre un peu de temps) :

```
sudo systemctl start elasticsearch
```

Si jamais vous rencontrez des problèmes d'initialisation, veuillez vérifier les logs du service elasticsearch à l'aide de la commande suivante :

```
sudo journalctl -f -u elasticsearch
```

Pour confirmer que tout fonctionne comme prévu, pointez votre commande curl ou votre navigateur sur l'adresse http://localhost:9200, et vous devriez voir quelque chose comme la sortie suivante :

```
curl localhost:9200
```

Résultat :

```
{
  "name" : "hatim-linux",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "U1zH_yFqTUmdMRckR1gHLQ",
  "version" : {
    "number" : "7.8.0",
    "build_flavor" : "default",
    "build_type" : "deb",
    "build_hash" : "757314695644ea9a1dc2fecd26d1a43856725e65",
    "build_date" : "2020-06-14T19:35:50.234439Z",
    "build_snapshot" : false,
    "lucene_version" : "8.5.1",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
```

Pour initialiser le service à chaque démarrage de la machine, lancez la commande suivante :

```
sudo systemctl enable elasticsearch
```

kibana Installation

Puisque nous avons déjà défini le dépôt dans le système, tout ce que nous avons à faire pour installer kibana est d'exécuter la commande suivante:

Sous la famille debian :

```
sudo apt-get install kibana
```

Sous la famille redhat :

```
sudo yum install kibana
```

configuration

Le fichier de configuration de kibana se retrouve dans **/etc/kibana/kibana.yml** . Si jamais vous avez modifié avec ce fichier, assurez-vous juste que la configuration kibana possède les bonnes informations pour communiquer avec Elasticsearch :

```
elasticsearch.hosts: ["http://localhost:9200"]
```

Lancement et test

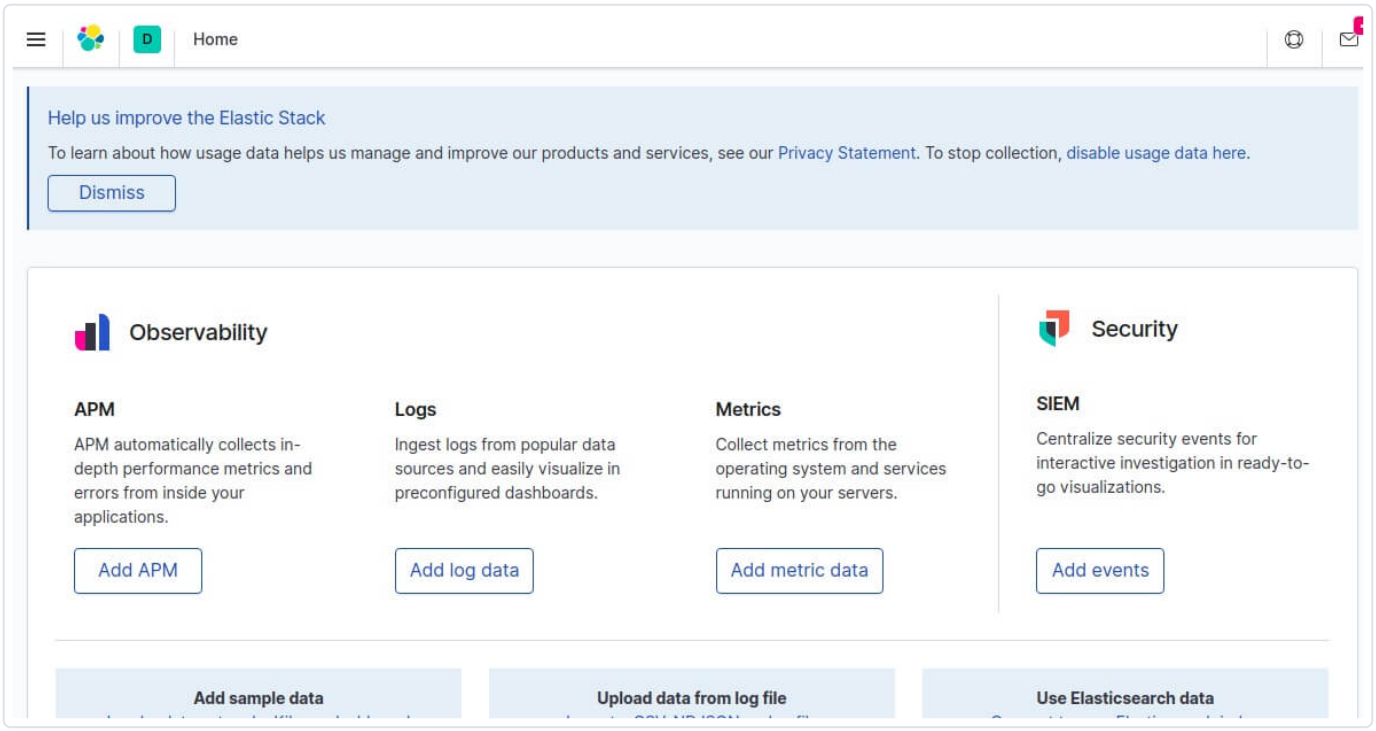
Voici la commande pour démarrer Kibana :

```
sudo systemctl start kibana
```

Si jamais vous rencontrez des problèmes d'initialisation, veuillez vérifier les logs du service kibana comme suit :

```
sudo journalctl -f -u kibana
```

Pour tester Kibana, ouvrez dans votre navigateur l'url http://localhost:5601 afin de voir la page d'accueil Kibana :



Pour initialiser le service Kibana à chaque démarrage de la machine, lancez la commande suivante :

```
sudo systemctl enable kibana
```

Logstash Installation

Logstash nécessite au minimum la version 8 de java pour fonctionner, nous allons donc commencer le processus de configuration de Logstash avec:

Sous la famille debian :

```
sudo apt-get install default-jre
```

Sous la famille redhat, java 8 :

```
yum install java-11-openjdk.x86_64

# ou java 8
yum install java-1.8.0-openjdk.x86_64
```

Enfin, vérifiez que java est installé:

```
java -version
```

Résultat :

```
openjdk version "11.0.7" 2020-04-14
...
```

Comme pour kibana, puisque nous avons déjà défini le dépôt dans le système, tout ce que nous avons à faire pour installer Logstash est d'exécuter:

Sous la famille debian :

```
sudo apt-get install logstash
```

Sous la famille redhat :

```
sudo yum install logstash
```

Configuration

Le fichier de configuration de Logstash est le suivant : **/etc/logstash/logstash.yml** et permet de configurer des paramètres généraux comme par exemple le nom du nœud, le port, le niveau des logs etc... Pour ce cours nous laisserons la configuration par défaut.

Lancement et test

Voici la commande pour démarrer logstash :

```
sudo systemctl start logstash
```

Si jamais vous rencontrez des problèmes d'initialisation, vérifiez les logs du service Logstash comme suit :

```
sudo journalctl -f -u logstash
```

Pour initialiser le service à chaque démarrage de la machine, lancez la commande suivante :

```
sudo systemctl enable logstash
```

Pour tester votre installation Logstash, vous devez configurer un pipeline de données. Nous aborderons cette partie dans le prochain chapitre.