



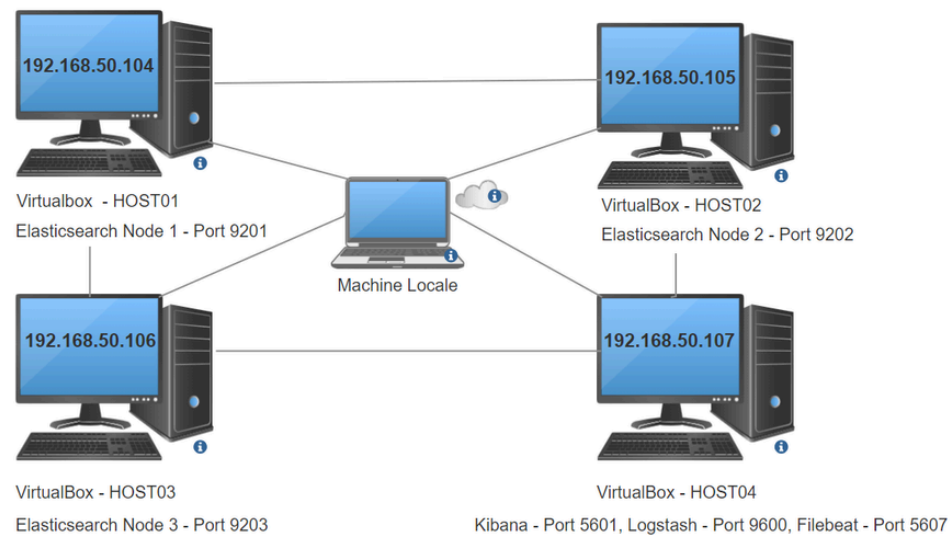
TP - Monitoring des logs d'une application Spring Boot avec Filebeat, Logstash, Elasticsearch et Kibana

Réalisé par **Mr. Iyanou Eraste AKANDE**, Ingénieur des données Telecom à Synaptique Maghreb, Ingénieur Certifié Elasticsearch.

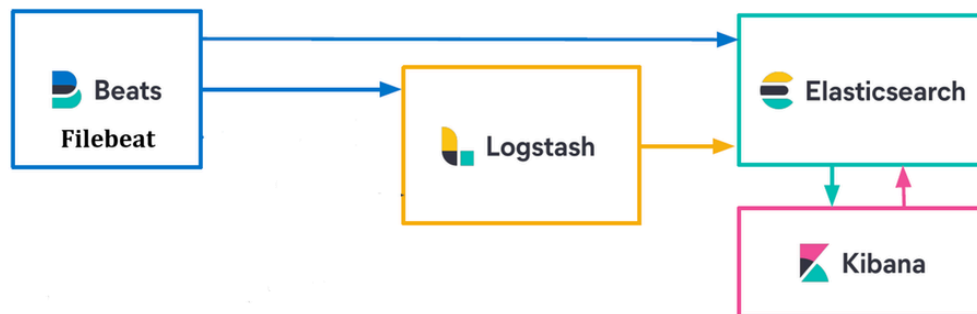
eraste.akande@gmail.com

Etape 1: Architecture

Les machines virtuelles



Les applications



Etape 2: Installation et configuration des machines virtuelles

- Téléchargez et installez la version de Virtual Box compatible à votre système d'exploitation sur le site <https://www.virtualbox.org/wiki/Downloads>.
- Dans certains cas pour les utilisateurs Windows, vous devriez installer au préalable Visual Studio C++ Redistributable à partir du site <https://learn.microsoft.com/en-us/cpp/windows/latest-supported-vc-redist?view=msvc-170>.

- Téléchargez et installez la version de Vagrant compatible à votre système d'exploitation à partir du site https://developer.hashicorp.com/vagrant/install?product_intent=vagrant.
- Téléchargez le fichier nommé Vagrantfile depuis ce répertoire GitHub <https://github.com/iyanou/elastic-training-lab>.
- Placez le fichier Vagrantfile dans un dossier et exécutez ce qui suit depuis cet emplacement

```
1 cd $vagrant_file_repository
2 vagrant up
```

- Se connecter sur chaque machine et vérifier qu'on arrive à faire un ping vers les autres adresses IP

```
1 ping -c 10 HOST01
2 ping -c 10 HOST02
3 ping -c 10 HOST03
4 ping -c 10 HOST04
```

Etape 3: Installation et configuration des applications [↗](#)

1. Les noeuds Elasticsearch

HOST01 [↗](#)

- Se connecter au HOST01 depuis votre machine en utilisant Vagrant

```
1 cd $vagrant_file_repository
2 vagrant ssh HOST01
```

- Création du dossier d'installation

```
1 sudo mkdir -p /opt/training
```

- Téléchargement de Elasticsearch 8.13.0

```
1 cd /opt/training
2 sudo wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-8.13.0-linux-x86_64.tar.gz
```

- Décompression de Elasticsearch 8.13.0

```
1 sudo tar -xzf elasticsearch-8.13.0-linux-x86_64.tar.gz
2 sudo rm elasticsearch-8.13.0-linux-x86_64.tar.gz
```

- Modifier le fichier de configuration `elasticsearch.yml` avec le contenu du fichier `es_node01.yml` situé dans le répertoire GitHub <https://github.com/iyanou/elastic-training-lab>.

```
1 cd /opt/training/elasticsearch-8.13.0
2 sudo rm config/elasticsearch.yml
3 sudo nano config/elasticsearch.yml
```

- Mettre à jour le paramètre `vm.max_map_count`

```
1 sudo grep vm.max_map_count /etc/sysctl.conf
2 grep 'vm.max_map_count=262144' /etc/sysctl.conf || sudo echo 'vm.max_map_count=262144' | sudo tee -a /etc/sysctl.conf
3 sudo sysctl -p
```

- Donner les permissions nécessaires à l'utilisateur sur le dossier elasticsearch

```
1 sudo chown -R vagrant:vagrant /opt/training/elasticsearch-8.13.0
```

- Générer les certificats de connexion elasticsearch en utilisant les fichiers contenus dans le répertoire GitHub <https://github.com/iyanou/elastic-training-lab/certificates>.

```
1 sudo mkdir -p config/certificates/ca
2 sudo mkdir -p config/certificates/node01
3 sudo nano config/certificates/node01/node01.key
4 sudo nano config/certificates/node01/node01.crt
5 sudo nano config/certificates/ca/ca.crt
```

- i** Voici comment tous les certificats ont été générés en utilisant le fichier `instances.yml` sur le répertoire GitHub <https://github.com/iyanou/elastic-training-lab/>.

```
1 cd /opt/training/elasticsearch-8.13.0
2 sudo mkdir /opt/training/elasticsearch-8.13.0/config/certificates
3 sudo nano config/certificates/instances.yml
4 sudo bin/elasticsearch-certutil ca --silent --pem -out config/certificates/ca.zip
5 sudo apt install unzip
6 sudo unzip -o config/certificates/ca.zip -d config/certificates
7 sudo bin/elasticsearch-certutil cert --silent --pem -out config/certificates/bundle.zip --in config/certi
8 sudo unzip -o config/certificates/bundle.zip -d config/certificates
9 sudo rm config/certificates/bundle.zip
10 sudo rm config/certificates/ca.zip
```

- Autoriser les ports sur le pare-feu

```
1 sudo ufw enable
2 sudo ufw allow 9201/tcp
3 sudo ufw allow 9301/tcp
4 sudo ufw status
```

- Démarrer le nœud Elasticsearch comme un daemon et vérifier les logs

```
1 ./bin/elasticsearch -d -p pid
```

- i** Pour arrêter le daemon elasticsearch `pkill -F pid`

HOST02 [↗](#)

- Se connecter au HOST02 depuis votre machine en utilisant Vagrant

```
1 cd $vagrant_file_repository
2 vagrant ssh HOST02
```

- Création du dossier d'installation

```
1 sudo mkdir -p /opt/training
```

- Téléchargement de Elasticsearch 8.13.0

```
1 cd /opt/training
2 sudo wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-8.13.0-linux-x86_64.tar.gz
```

- Décompression de Elasticsearch 8.13.0

```
1 sudo tar -xzf elasticsearch-8.13.0-linux-x86_64.tar.gz
2 sudo rm elasticsearch-8.13.0-linux-x86_64.tar.gz
```

- Modifier le fichier de configuration `elasticsearch.yml` avec le contenu du fichier `es_node02.yml` situé dans le répertoire GitHub <https://github.com/ianou/elastic-training-lab>.

```
1 cd /opt/training/elasticsearch-8.13.0
2 sudo rm config/elasticsearch.yml
3 sudo nano config/elasticsearch.yml
```

- Mettre à jour le paramètre `vm.max_map_count`

```
1 sudo grep vm.max_map_count /etc/sysctl.conf
2 grep 'vm.max_map_count=262144' /etc/sysctl.conf || sudo echo 'vm.max_map_count=262144' | sudo tee -a /etc/sysctl.conf
3 sudo sysctl -p
```

- Donner les permissions nécessaires à l'utilisateur sur le dossier `elasticsearch`

```
1 sudo chown -R vagrant:vagrant /opt/training/elasticsearch-8.13.0
```

- Générer les certificats de connexion elasticsearch en utilisant les fichiers contenus dans le répertoire GitHub <https://github.com/ianou/elastic-training-lab/certificates>.

```
1 sudo mkdir -p config/certificates/ca
2 sudo mkdir -p config/certificates/node02
3 sudo nano config/certificates/node02/node02.key
4 sudo nano config/certificates/node02/node02.crt
5 sudo nano config/certificates/ca/ca.crt
```

- Autoriser les ports sur le pare-feu

```
1 sudo ufw enable
2 sudo ufw allow 9202/tcp
3 sudo ufw allow 9302/tcp
4 sudo ufw status
```

- Démarrer le nœud Elasticsearch comme un daemon et vérifier les logs

```
1 ./bin/elasticsearch -d -p pid
```

HOST03

- Se connecter au HOST03 depuis votre machine en utilisant Vagrant

```
1 cd $vagrant_file_repository
2 vagrant ssh HOST03
```

- Création du dossier d'installation

```
1 sudo mkdir -p /opt/training
```

- Téléchargement de Elasticsearch 8.13.0

```
1 cd /opt/training
2 sudo wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-8.13.0-linux-x86_64.tar.gz
```

- Décompression de Elasticsearch 8.13.0

```
1 sudo tar -xzf elasticsearch-8.13.0-linux-x86_64.tar.gz
2 sudo rm elasticsearch-8.13.0-linux-x86_64.tar.gz
```

- Modifier le fichier de configuration `elasticsearch.yml` avec le contenu du fichier `es_node03.yml` situé dans le répertoire GitHub <https://github.com/ianou/elastic-training-lab>.

```
1 cd /opt/training/elasticsearch-8.13.0
2 sudo rm config/elasticsearch.yml
3 sudo nano config/elasticsearch.yml
```

- Mettre à jour le paramètre `vm.max_map_count`

```
1 sudo grep vm.max_map_count /etc/sysctl.conf
2 grep 'vm.max_map_count=262144' /etc/sysctl.conf || sudo echo 'vm.max_map_count=262144' | sudo tee -a /etc/sysctl.conf
3 sudo sysctl -p
```

- Donner les permissions nécessaires à l'utilisateur sur le dossier elasticsearch

```
1 sudo chown -R vagrant:vagrant /opt/training/elasticsearch-8.13.0
```

- Générer les certificats de connexion elasticsearch en utilisant les fichiers contenus dans le répertoire GitHub <https://github.com/ianou/elastic-training-lab/certificates> .

```
1 sudo mkdir -p config/certificates/ca
2 sudo mkdir -p config/certificates/node03
3 sudo nano config/certificates/node03/node03.key
4 sudo nano config/certificates/node03/node03.crt
5 sudo nano config/certificates/ca/ca.crt
```

- Autoriser les ports sur le pare-feu

```
1 sudo ufw enable
2 sudo ufw allow 9203/tcp
3 sudo ufw allow 9303/tcp
4 sudo ufw status
```

- Démarrer le nœud Elasticsearch comme un daemon et vérifier les logs

```
1 ./bin/elasticsearch -d -p pid
```

Connexion au Cluster Elasticsearch

- Générer le mot de passe de l'utilisateur `elastic` et `kibana_system` en utilisant l'un des hosts et sauvegarder ces mots de passe

```
1 cd /opt/training/elasticsearch-8.13.0
2 bin/elasticsearch-reset-password -u elastic
3 bin/elasticsearch-reset-password -u kibana_system
```

- Sur votre machine local, aller sur votre navigateur et accéder au lien https://192.168.50.104:9201/_cluster/health?pretty pour vérifier que le cluster est en bon état (statut "green").
- Sur votre machine local, aller sur votre navigateur et accéder au lien https://192.168.50.104:9201/_cat/nodes?pretty pour vérifier que les 3 nœuds ont rejoint le cluster.

 Au lieu d'aller sur le navigateur, on peut faire les mêmes vérifications depuis l'un des hosts avec les commandes suivantes :

```
1 curl --cacert /opt/training/elasticsearch-8.13.0/config/certificates/ca/ca.crt -u elastic:$elastic_password
2 curl --cacert /opt/training/elasticsearch-8.13.0/config/certificates/ca/ca.crt -u elastic:$elastic_password
```

2. Le serveur Kibana

 La configuration de Kibana se fera sur le HOST04

- Se connecter au HOST04 depuis votre machine en utilisant Vagrant

```
1 cd $vagrant_file_repository
2 vagrant ssh HOST04
```

- Création du dossier d'installation

```
1 sudo mkdir -p /opt/training
```

- Téléchargement de Kibana 8.13.0

```
1 cd /opt/training
2 sudo wget https://artifacts.elastic.co/downloads/kibana/kibana-8.13.0-amd64.deb
```

- Décompression de Kibana 8.13.0

```
1 sudo dpkg -i kibana-8.13.0-amd64.deb
2 sudo rm kibana-8.13.0-amd64.deb
```

- Modifier le fichier de configuration `kibana.yml` avec le contenu du fichier `kibana.yml` situé dans le répertoire GitHub <https://github.com/ianou/elastic-training-lab>. Ne pas oublier de mettre à jour le mot de passe du user `kibana_system`.

```
1 sudo rm /etc/kibana/kibana.yml
2 sudo nano /etc/kibana/kibana.yml
```

- Déposer le certificat d'autorité de Elasticsearch sur le HOST04. Copier le certificat depuis le répertoire GitHub <https://github.com/ianou/elastic-training-lab/certificates>.

```
1 sudo mkdir -p /opt/training/ca
2 sudo nano /opt/training/ca/ca.crt
```

- Autoriser les ports sur le pare-feu

```
1 sudo ufw enable
2 sudo ufw allow 5601/tcp
3 sudo ufw status
```

- Démarrer Kibana

```
1 sudo /bin/systemctl daemon-reload
2 sudo systemctl enable kibana.service
3 sudo systemctl start kibana.service
4 sudo systemctl status kibana.service
```

 Pour arrêter le service Kibana, exécutez `sudo systemctl stop kibana.service`.

- Afficher les logs du service Kibana

```
1 sudo journalctl --unit=kibana.service -n 100 --no-pager
```

- Sur votre machine aller sur l'interface de Kibana <http://192.168.50.107:5601> et connecter vous avec le user `elastic` et son mot de passe généré précédemment.

3. L'ETL Logstash

La configuration de Logstash se fera sur le HOST04

- Se connecter au HOST04 depuis votre machine en utilisant Vagrant

```
1 cd $vagrant_file_repository
2 vagrant ssh HOST04
```

- Téléchargement de Logstash 8.13.0

```
1 cd /opt/training
2 sudo wget https://artifacts.elastic.co/downloads/logstash/logstash-8.13.0-amd64.deb
```

- Décompression de Logstash 8.13.0

```
1 sudo dpkg -i logstash-8.13.0-amd64.deb
2 sudo rm logstash-8.13.0-amd64.deb
```

- Modifier le fichier de configuration `logstash.yml` avec le contenu du fichier `logstash.yml` situé dans le répertoire GitHub <https://github.com/ianou/elastic-training-lab>.

```
1 sudo rm /etc/logstash/logstash.yml
2 sudo nano /etc/logstash/logstash.yml
```

- Autoriser les ports sur le pare-feu

```
1 sudo ufw enable
2 sudo ufw allow 9600/tcp
3 sudo ufw allow 5044/tcp
4 sudo ufw status
```

- Donner les permissions nécessaires à Logstash

```
1 sudo chmod 755 /usr/share/logstash/data
2 sudo chown -R logstash:logstash /usr/share/logstash/data
```

- Démarrer Logstash

```
1 sudo /bin/systemctl daemon-reload
2 sudo systemctl enable logstash.service
3 sudo systemctl start logstash.service
4 sudo systemctl status logstash.service
```

 Pour arrêter le service Logstash, exécuter `sudo systemctl stop logstash.service`.

- Afficher les logs du service Logstash

```
1 sudo journalctl --unit=logstash.service -n 100 --no-pager
```

- Aller sur l'interface de Logstash sur votre navigateur avec le lien <http://192.168.50.107:9600/?pretty> pour se rassurer que Logstash est démarré.

4. L'agent Filebeat

La configuration de Filebeat se fera sur le HOST04

- Se connecter au HOST04 depuis votre machine en utilisant Vagrant

```
1 cd $vagrant_file_repository
2 vagrant ssh HOST04
```

- Téléchargement de Filebeat 8.13.0

```
1 cd /opt/training
2 sudo wget https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-8.13.0-amd64.deb
```

- Décompression de Filebeat 8.13.0


```
1 sudo dpkg -i filebeat-8.13.0-amd64.deb
2 sudo rm filebeat-8.13.0-amd64.deb
```

- Autoriser les ports sur le pare-feu

```
1 sudo ufw enable
2 sudo ufw allow 5067/tcp
3 sudo ufw status
```

- Démarrer Filebeat


```
1 sudo /bin/systemctl daemon-reload
2 sudo systemctl enable filebeat.service
3 sudo systemctl start filebeat.service
4 sudo systemctl status filebeat.service
```

 Pour arrêter le service Filebeat, exécuter `sudo systemctl stop logstash.service`.


- Afficher les logs du service Filebeat

```
1 sudo journalctl --unit=filebeat.service -n 100 --no-pager
```

Etape 4: Introduction à la pile ELK [↗](#)

 Présentation des composantes de la pile ELK, le rôle de chaque composante ainsi que les différents cas d'utilisation → (PPT Introduction to Elastic Stack).

Etape 5: Introduction à Filebeat [↗](#)

 Explication du fonctionnement de Filebeat et présentation des différentes options et paramètres de configuration → (PPT Starting with Filebeat).

Etape 6: LAB 1 - Collecte des logs Spring Boot : De Filebeat vers Elasticsearch [↗](#)

 Ce LAB se déroulera sur le HOST04.

- Se connecter au HOST04 depuis votre machine en utilisant Vagrant

```
1 cd $vagrant_file_repository
2 vagrant ssh HOST04
```

- Création du fichier de logs Spring Boot en utilisant le fichier `spring-boot-app-logs.json` situé dans le répertoire GitHub <https://github.com/iyanou/elastic-training-lab>.

```
1 cd /opt/training
2 sudo mkdir -p /opt/training/app/logs
3 sudo nano /opt/training/app/logs/spring-boot-app-logs.json
```

- Configuration de `filebeat.yml` en utilisant le fichier `filebeat_lab1.yml` situé dans le répertoire GitHub <https://github.com/iyanou/elastic-training-lab>.

```
1 sudo rm /etc/filebeat/filebeat.yml
2 sudo nano /etc/filebeat/filebeat.yml
```

- Redémarrer le service Filebeat


```
1 sudo systemctl restart filebeat.service
```

- Afficher les logs du service Filebeat

```
1 sudo journalctl --unit=filebeat.service -n 100 --no-pager
```

- Aller sur Kibana, créer un data view nommé **"filebeat"** et aller dans la rubrique **"Discover"** pour vérifier que les logs ont été envoyés sur le cluster Elasticsearch.

Etape 7: Introduction à Logstash

-  Explication du fonctionnement de Logstash et présentation des différentes options et paramètres de configuration des pipelines Logstash → (PPT Starting with Logstash).

Etape 8: LAB 2 - Collecte des logs Spring Boot : De Filebeat vers Logstash et transfert vers Elasticsearch

-  Ce LAB se déroulera sur le HOST04.

- Se connecter au HOST04 depuis votre machine en utilisant Vagrant

```
1 cd $vagrant_file_repository
2 vagrant ssh HOST04
```

- Créer le pipeline Logstash pour le traitement des données venant de Filebeat. Utiliser le fichier `filebeat-pipeline.conf` situé dans le répertoire GitHub <https://github.com/iyanou/elastic-training-lab>.

```
1 sudo nano /etc/logstash/conf.d/filebeat-pipeline.conf
```

- Supprimer le data stream de Filebeat existant sur le cluster avec Dev Tools

```
1 DELETE /_data_stream/filebeat-8.13.0
```

- Redémarrer le service Logstash

```
1 sudo systemctl restart logstash.service
```

- Configuration de `filebeat.yml` en utilisant le fichier `filebeat_lab2.yml` situé dans le répertoire GitHub <https://github.com/ianou/elastic-training-lab>.

```
1 sudo rm /etc/filebeat/filebeat.yml
2 sudo nano /etc/filebeat/filebeat.yml
```

- Vider le registre de Filebeat

```
1 sudo rm -R /var/lib/filebeat/registry
2 sudo rm /var/lib/filebeat/filebeat.lock
3 sudo rm /var/lib/filebeat/meta.json
```

- Redémarrer le service Filebeat

```
1 sudo systemctl restart filebeat.service
```

- Afficher les logs du service Filebeat

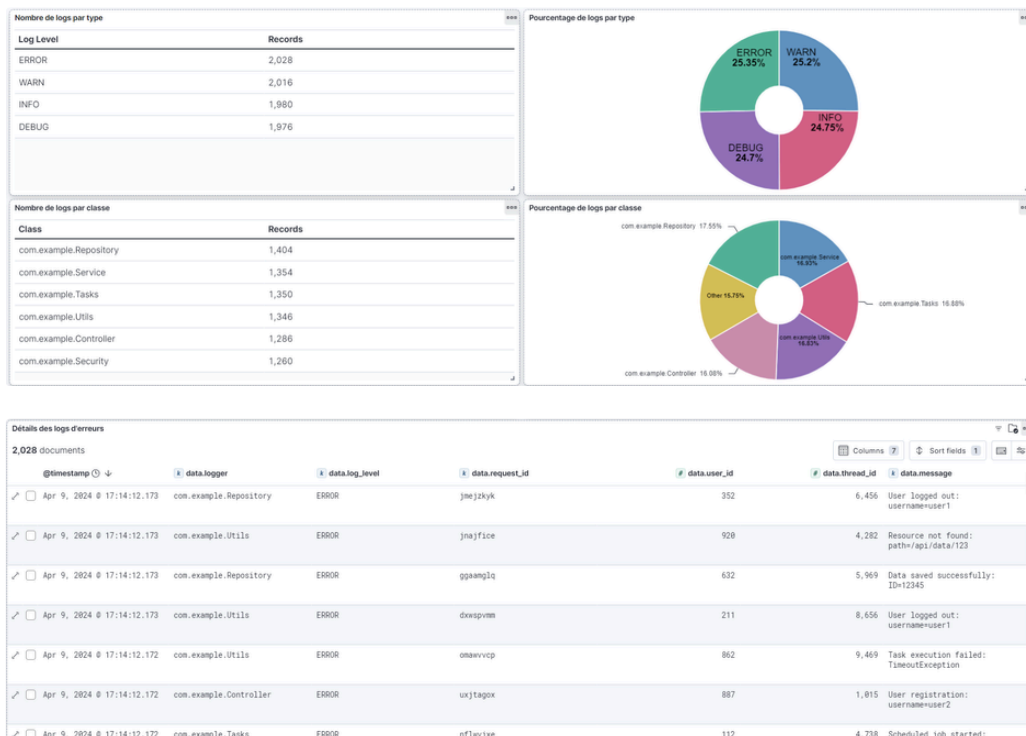
```
1 sudo journalctl --unit=filebeat.service -n 100 --no-pager
```

- Afficher les logs du service Logstash

```
1 sudo journalctl --unit=logstash.service -n 100 --no-pager
```

- Aller sur Kibana, choisir le data view nommé **"filebeat"** et aller dans la rubrique **"Discover"** pour vérifier que les logs ont été envoyés sur le cluster Elasticsearch.

Etape 9: Visualisation des Logs et création de Dashboards avec Kibana [↗](#)



Tester que le système est automatique

Sur le HOST04,

- Créer un nouveau fichier `spring-boot-app-logs-test.json` en utilisant le fichier `spring-boot-app-logs-test.json` situé dans le répertoire GitHub <https://github.com/iyanou/elastic-training-lab>.

```
1 sudo nano /opt/training/app/logs/spring-boot-app-logs-test.json
```

- Visualiser que les logs ont mis le Dashboard à jour.

Etape 10: Quelques concepts clés d'Elasticsearch

 Explication de quelques concepts importants liés à Elasticsearch et à son fonctionnement → (PPT Starting with Elasticsearch).