Sources & liens utiles

- Grafikart - Tutoriel Kibana. Ce tutoriel vidéo de 40min apprend à installer et configurer Vagrant et la suite ELK puis à utiliser Kibana. Il est passé sous silence la mise en place d'un site web jouet sous nginx mais une autre vidéo illustre la manoeuvre.
- Documentation de la suite elastic
- Documentation logstash pour les piplines de parsing de logs
- GrokDebugger pour vérifier que les règles de parsing issus du point précédent sont correctes et adaptées aux lignes de logs fournies.

Indications personnelles

***NOTE** : dans cette section nous référerons à l'environnement de travail Vagrant par 'la VM', mais il faut garder à l'esprit que Vagrant est en réalité une sur-couche pour la création et la configuration des environnements de développement virtuel. Notamment, il est possible d'utiliser le logiciel de gestion de VM de notre choix; dans notre cas nous avons choisi VirtualBox.*

Le site jouet est atteignable sur `localhost` remplace la page d'accueil par défaut de nginx, et Kibana est atteignable sur `stats.localhost`.

Installation & configuration

- Les paquets nécessaires pour reproduire le projet sont listés ci-dessous. Attention, la version des éléments de la suite ELK est *6.1.2* dans la vidéo et nous conseillons d'utiliser cette version disponible sur ce lien car les versions ultérieures ont des comportements différents tant au niveau de la configuration que de la manipulation.
  - `nginx`
  - `apache2-utils`
  - `elasticsearch` (version *6.1.2*)
  - `logstash` (version *6.1.2*)
  - `kibana` (version *6.1.2*)
  - `firefox` (ou tout autre navigateur)
  - `nano` (ou tout autre éditeur de texte)
- Les fichiers de configuration des différents paquets ainsi que le site jouet utilisé pour le projet sont disponibles dans le dossier `root_vagrant` de ce dépôt. L'architecture de dossiers est conservée pour indiquer où copier les fichiers.
- Le fichier de logs utilisé pour analyse par la suite ELK est `access.log` et est à placer dans la VM dans `/home/vagrant/casi_webapp/`. Pour simuler une arrivée de log par interaction d'un site web quelconque il suffit d'ajouter de l'information dans le fichier pendant que la suite ELK tourne.

***NOTE** : l'intérêt du paquet `apache2-utils` est de pouvoir définir un couple ID-password pour pouvoir accéder à l'application Kibana. Cette étape doit être reproduite en lançant la commande `sudo htpasswd -c /etc/nginx/htpasswd.users tuto` afin de créer un fichier `/etc/nginx/htpasswd.users` référencé dans `/etc/nginx/sites-available/kibana`. L'identifiant sera alors 'tuto' et le mot de passe celui choisi par l'utilisateur.*

Vagrant

- Les commandes de manipulation de Vagrant se lancent via terminal à l'emplacement du fichier de configuration `Vagrantfile`. Commandes utiles :

    - `vagrant up` : démarre la VM en utilisant le fichier de configuration `Vagrantfile` présent dans le dossier courant
    - `vagrant halt` : ferme la VM
    - `vagrant ssh` : accéder à la VM démarrée
    - `vagrant destroy` : détruire la VM
    - `vagrant package` : compresse la VM en un fichier partageable et décompressable (attention, le fichier généré est volumineux [quelques Go])

- Pour partager des fichiers entre la machine hôte et la VM de Vagrant, il suffit de déposer les fichiers souhaités à l'emplacement du `Vagrantfile` puisque cet emplacement est monté dans la VM au point `/vagrant`.

- Il est possible d'accéder à Kibana depuis l'extérieur de la VM, mais il faut s'assurer que les port forwarding sont en place dans le fichier de configuration Vagrant. Autrement, toujours dans ce fichier de configuration, il est possible de forwarder les applications graphiques gérées par X11 en ajoutant les lignes suivantes :

```
config.ssh.forward_agent = true
config.ssh.forward_x11 = true
```

- Eteindre la VM n'a pas d'incidence sur la conservation des données.

ELK

- Si après plusieurs installations d'elasticsearch, le service ne se lance plus correctement (i.e. erreur de lancement indiquée par `sudo journalctl --unit elasticsearch`), c'est peut-être dû à un conflit de node entre les versions. Dans ce cas tenter : `sudo rm -rf /var/lib/elasticsearch/nodes/` puis relancer le service.

- Autour de 20:55 dans la vidéo, il faut :

    - vérifier que les logs sont correctement parsés en demandant de lister des index transformés de quelques logs **sans les enregistrer** :

    ```
    sudo /usr/share/logstash/bin/logstash --debug --path.settings /etc/logstash/ -f /etc/logstash/conf.d/01-l
    ```

    - créer les index transformés à partir des logs en remplissant le fichier de log après avoir lancé cette commande (l'argument `--debug` est optionnel et ralentit le processus) :

    ```
    sudo /usr/share/logstash/bin/logstash --debug --path.settings /etc/logstash/ -f /etc/logstash/conf.d/01-l
    ```

- Une fois Kibana lancé, rajouter des données dans le fichier de log devrait avoir pour conséquence de mettre à jour les visualiseurs des dashboards mais nous ne l'avons pas constaté dans nos tests. Notre hypothèse est que les index ne sont pas toujours re-parsés par elasticsearch à chaque mise à jour du fichier surveillé (renseigné dans `/etc/logstash/conf.d/01-local-dev.conf` dans le champ `file` de `input`).