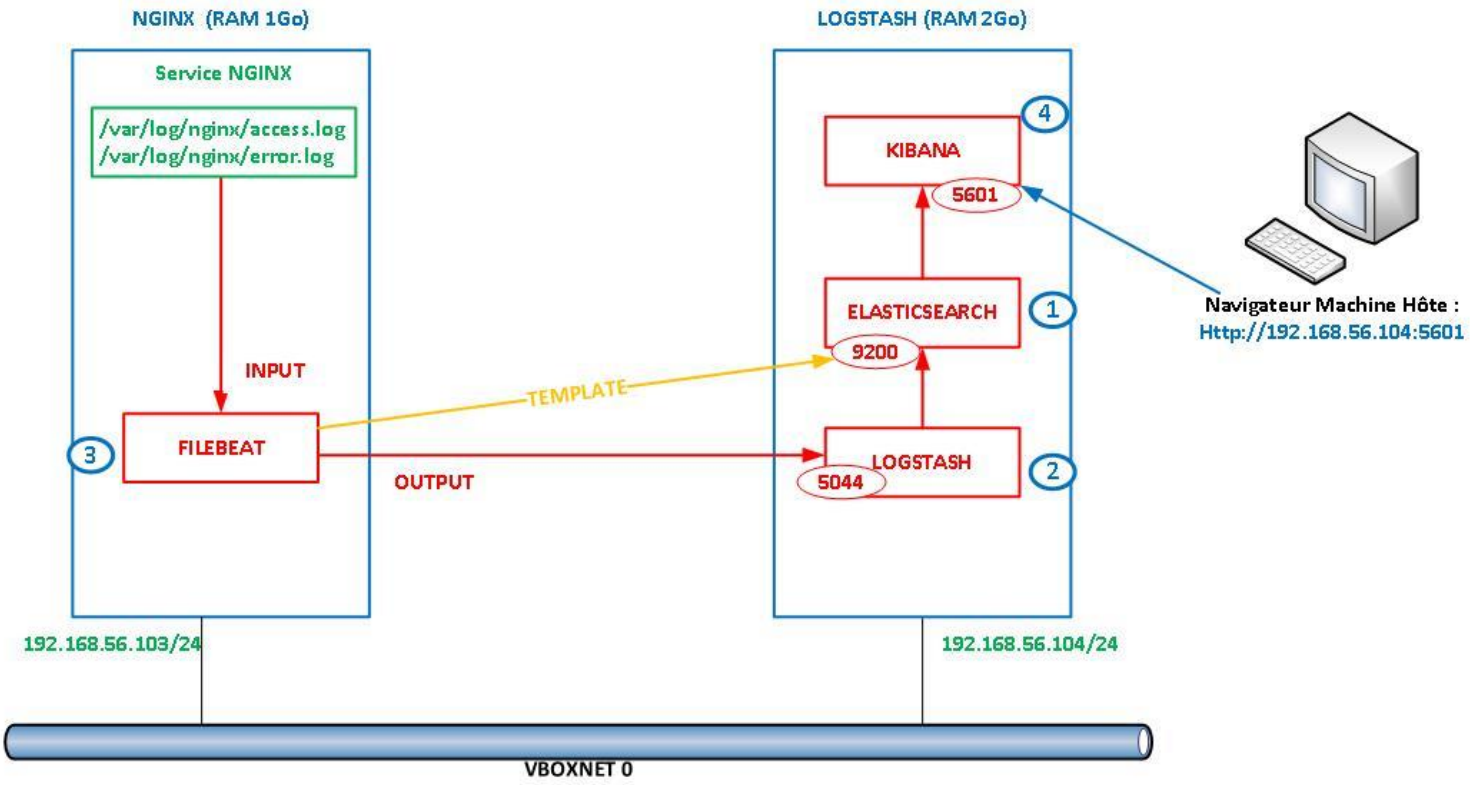
Prérequis :

2 VM type Debian ou Ubuntu Server

Sur Virtualbox :
2 cartes réseau :
- 1 NAT
- 1 Host-Only adapter sur vboxnet0 (adresse IP en 192.168.56.xx automatiquement alloués par Virtualbox))

Sur Proxmox :
1 carte réseau sur vmbr2 (adresse en 172.23.x.x au lieu de 192.168.56.xx)

Objectif final :



1. Préparation de 2 VMS :

Cloner la VM modèle en Linked-Clone vers :
→1 VM nommée Nginx, avec 1Go de RAM (pour Nginx et Filebeat)
→1 VM nommée Logstash avec 2Go de RAM (pour Logstash, Elasticsearch et Kibana)

1 fois les Vms démarées , vérifier les fichiers **/etc/network/interfaces** qui doivent contenir:

```
auto enp0s8
iface enp0s8 inet dhcp
```

/etc/hostname doit contenir le nom de la VM (respectivement nginx et logstash)

Rédémarrer les Vms et vérifier
- l'adresse IP obtenue grâce à la commande "ip addr"
- la communication entre les Vms
- la communication depuis le poste hôte vers les Vms

2. Installation de Nginx :

```
# apt-get isntall nginx
# systemctl status nginx.service
```

Le navigateur de la machine hôte doit être capable de se connecter en http sur la VM nginx
http://<@IP_VM_nginx>

3. Installation d'Elastic Search sur la VM Logstash:

Pour plus d'infos , se referrer à :
https://www.elastic.co/guide/en/elasticsearch/reference/current/deb.html

```
# wget http://www.utc.fr/~quetwilf/sr07/elk_repo.sh
# chmod 750 elk_repo.sh
#./elk_repo.sh
```

Ce script contient les lignes suivantes :

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

# apt-get install apt-transport-https
# apt-get install curl
# apt-get install openjdk-11-jdk

# echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-7.x.list

# apt-get update

# apt-get install elasticsearch
```

Ensuite active le service elasticsearch :

```
# systemctl enable elasticsearch
```

Editer le fichier **/etc/elasticsearch/elasticsearch.yml**
ajouter :

```
#network.host: localhost
network.bind_host: 0.0.0.0
```

Editer le fichier /etc/elasticsearch/jvm.options et remplacer :

```
-Xms1g
-Xmx1g
```

par

```
-Xms512m
-Xmx512m
```

```
# systemctl start elasticsearch
# systemctl status elasticsearch    => le status doit être active
```

En cas d'échec, analyser le démarrage du service elasticsearch :

```
# journalctl --unit elasticsearch
```

Vérifier le fonctionnement d'elasticsearch :

```
# curl localhost:9200
```

du texte au format JSON doit être renvoyé à l'écran :

```
{
  "name" : ……
…}
```

4. Installation de Logstash sur la VM Logstash:

```
# apt-get install logstash
```

Créer le fichier **/etc/logstash/conf.d/01-sr07.conf** :

Vous pouvez aussi le récupérer avec wget :

```
# cd /etc/logstash/conf.d
# wget http://www.utc.fr/~quetwilf/sr07/01-sr07.conf
```

```
input {
  beats {
    port => 5044
    host => "0.0.0.0"
  }
}
filter {
  if [fileset][module] == "nginx" {
    if [fileset][name] == "access" {
      grok {
        match => { "message" =>
["%{IPORHOST:[nginx][access][remote_ip]}
- %{DATA:[nginx][access][user_name]}
\[%{HTTPDATE:[nginx][access][time]}\]
\"%{WORD:[nginx][access][method]} %{DATA:[nginx][access][url]}
HTTP/%{NUMBER:[nginx][access][http_version]}\" %{NUMBER:[nginx][a
ccess][response_code]} %{NUMBER:[nginx][access][body_sent][bytes]}
\"%{DATA:[nginx][access][referrer]}\"
\"%{DATA:[nginx][access][agent]}\""] }
        remove_field => "message"
      }
      mutate {
        add_field => { "read_timestamp" => "%{@timestamp}" }
      }
      date {
        match => [ "[nginx][access][time]", "dd/MMM/YYYY:H:m:s Z" ]
```

```
      remove_field => "[nginx][access][time]"
    }
    useragent {
      source => "[nginx][access][agent]"
      target => "[nginx][access][user_agent]"
      remove_field => "[nginx][access][agent]"
    }
    geoip {
      source => "[nginx][access][remote_ip]"
      target => "[nginx][access][geoip]"
    }
  }
  else if [fileset][name] == "error" {
    grok {
      match => { "message" => ["%{DATA:[nginx][error][time]}
\[%{DATA:[nginx][error][level]}\] %{NUMBER:[nginx][error][pid]}#%{NUMB
ER:[nginx][error][tid]}:
(\*%{NUMBER:[nginx][error][connection_id]} )?%{GREEDYDATA:[nginx][
error][message]}"] }
      remove_field => "message"
    }
    mutate {
      rename => { "@timestamp" => "read_timestamp" }
    }
    date {
      match => [ "[nginx][error][time]", "YYYY/MM/dd H:m:s" ]
      remove_field => "[nginx][error][time]"
    }
  }
 }
}
output {
 stdout {
  codec => rubydebug
 }
 elasticsearch {
  hosts => localhost
  manage_template => false
  index =>
"%{[@metadata][beat]}-%{[@metadata][version]}-%{+YYYY.MM.dd}"
 }
 file {
  path => "/var/log/logstash/sr07.log"
```

```
    codec => rubydebug
  }
}
```

Le pattern en rouge peut être éventuellement testé sur le site *grok debugger* (trouvé avec une recherche Google : grok debugger).

Lancer Logstash en mode debug en vérifiant la syntaxe du fichier de configuration.

```
# /usr/share/logstash/bin/logstash --debug --path.settings /etc/logstash -f /etc/logstash/conf.d/01-sr07.conf  -t
```

Cela peut prendre quelques secondes :

```
=> Config Validation Result : OK
```

Modifier les droits sur les 2 répertoires :

```
# chown  -R  logstash:logstash  /var/lib/logstash  /var/log/logstash
```

Relancer en mode debug sans verification de conf:

```
# systemctl enable logstash

# systemctl start logstash
```

Pour vérifier le démarrage de Logstash :

```
tail -f /var/log/logstash/logstash-plain.log
```

5. Installation de Filebeat sur la VM Nginx:

En root, sur la VM Nginx :

```
# wget http://www.utc.fr/~quetwilf/sr07/elk_repo.sh
# chmod 750 elk_repo.sh
#./elk_repo.sh
# apt-get update
# apt-get install filebeat
# filebeat modules enable nginx
```

Editer le fichier **/etc/filebeat/filebeat.yml** :

```
filebeats.inputs:
-type: log
  enable: true
  paths:
    - /var/log/nginx/*.log
#   - /var/log/*.log
....
...
# output.elasticsearch:
# hosts: ["localhost:9200"]
output.logstash:
  hosts: ["192.168.56.104:5044"]
...
processors:
  - add_host_metadata: ~
#  - add_cloud_metadata: ~
```

avec 192.168.56.104 = @IP du serveur Logstash

Afin qu'Elasticsearch indexe ses documents en function d'une source de type Filebeat, il faut transférer le template Filebeat à Elasticsearch directement en bypassant Logstash.
Pour ce, exécuter la commande sur le serveur Nginx/filebeat :

```
# filebeat setup --template -E output.logstash.enabled=false -E
'output.elasticsearch.hosts=["192.168.56.104:9200"]'
```

avec 192.168.56.104 = @IP du serveur Elasticsearch

6. Test d'envoi des logs:

Sur la machine Logstash :

```
# tail -f /var/log/logstash/sr07.log
```

Sur la machine Hôte, dans un navigateur :
http://192.168.56.103

avec 192.168.56.103 = @IP de la machine Nginx

Le fichier **/var/log/logstash** doit faire apparaître les lignes de Logs Nginx.

Pour vérifier si Elasticsearch indexe bien les données envoyées par Logstash, lancer la commande suivante sur la machine Logstash :

```
# curl "localhost:9200/_cat/indices?v"
health status index                    uuid                   pri rep
docs.count docs.deleted store.size pri.store.size
yellow open   filebeat-6.5.1-2018.12.01 baqppwnqR6-E6SA_lzezDg
          3   1         24            0      148kb          148kb
```

7. Installation de Kibana:

Sur la machine Logstash :

```
# apt-get install kibana
```

Editer le fichier **/etc/kibana/kibana.yml** et ajouter la ligne suivante :

```
server.host: "0.0.0.0"
```

```
# systemctl enable kibana
# systemctl start kibana
```

Depuis le système hôte, accéder à http://192.168.56.104:5601
avec 192.168.56.104 = @IP du serveur Kibana

7.1. Création de l'index pattern pour Filebeat

Suivre les liens suivants sur l'interface web Kibana :

Dashboard => Create index Pattern
Index Pattern = filebeat* => Next
Time Filter field name = @timestamp => Create index pattern

7.2. Visualisation des logs :

Logs => Streaming live

Sur le serveur Nginx, ajouter manuellement des lignes de logs au fichier access.log :

```
# tail -1 /var/log/nginx/access.log | sed
's/192.168.56.2/195.83.155.55/g' > /tmp/sr07.log
# tail -1 /var/log/nginx/access.log | sed
's/192.168.56.2/195.83.155.56/g' >> /tmp/sr07.log
# tail -1 /var/log/nginx/access.log | sed
's/192.168.56.2/195.83.155.57/g' >> /tmp/sr07.log
# tail -1 /var/log/nginx/access.log | sed
's/192.168.56.2/195.83.155.58/g' >> /tmp/sr07.log
# cat  /tmp/sr07.log >>  /var/log/nginx/access.log
```

7.3. Création d'une visualisation Visiteurs:

Visualize => Create a visualization :
Area => select index = filebeat*
Data => metrics =>
Y-Axis : Aggregation = unique count
Filter = nginx.access.remote_ip
Custom_label = Visiteurs
X-Axis : Aggregation = date histogram
Filter = @timestamp
Custom_label = Date

Save => Save visualisation as ... = Visiteurs

7.4. Création d'une visualisation Visites:

Créer une visualization :

Visualize => Create a visualization :
Area => select index = filebeat*
Data => metrics =>
Y-Axis : Aggregation = count
Custom_label = Visites
X-Axis : Aggregation = date histogram
Filter = @timestamp
Custom_label = Date

Save => Save visualization as ... = Visites

7.5. Création d'un Dashboard :

Dashboard => Add visualization : Visiteurs + Visites
Save => SR07

7.6. Injection de logs :

Sur la machine Nginx, télécharger le fichier nginx_sr07.log.gz.

```
# wget http://www.utc.fr/~quetwilf/sr07/nginx_sr07.log.gz
# gunzip nginx_sr07.log.gz
# cat nginx_sr07.log >> /var/log/nginx/access.log
```

7.7. Création de la map de provenance des requêtes :

Créer une visualization :

Type = Coordinate Map

Metrics :
value = count

Buckets :
type = Geo Coordinate
aggregation = Geohash
field = nginx.access.geoip.location
v change precision...
v place markers...
v only requests...

7.7. Création du Top 10 des OS Agent :

Type = Pie
Metrics : Slice Size
Aggregation = unique count
Filed = nginx.access.remote_ip
Buckets : Splice slices
Aggregation = Terms
Field = nginx.access.user_agent.os_name
Order_by = Remote_IP
Order = Descending
Size = 10

7.8. Création du Top 10 des URLs accédées :

A vous de jouer .....

8. Interrogation de serveur Nginx en SNMP :

Prérequis : package snmpd

Faire en sorte que l'on puisse interroger en snmp le nombre d'octets entrants et sortants de l' interface enp0s3 du serveur Nginx.

Interroger l'utilisation de la CPU du serveur Nginx.

Indications :
*- fichier PC : **/etc/snmp/snmpd.conf***