Centralisation et analyse des journaux système avec rsyslog, Filebeat, Elasticsearch, Kibana

Guide Technique Complet

Champ	Valeur
Projet	BTS SIO SISR - Centralisation et analyse des logs
Etudiant	Luc
Serveur	Ubulog (SRV-RSYSLOG) - Ubuntu Server 24.04 LTS
Stack	Rsyslog + Filebeat + Elasticsearch 8.x + Kibana
Domaine	srvlog.lan
Date	Mars 2026

PARTIE 1 - DEPLOIEMENT DES SERVEURS ET CONFIGURATION DU RESEAU

1. Contexte et Objectifs

Dans toute infrastructure informatique professionnelle, la centralisation des journaux système (logs) est une pratique fondamentale. Elle permet d'assurer la supervision efficace des systèmes, de diagnostiquer rapidement les incidents, de renforcer la sécurité par la détection d'anomalies et de garantir la traçabilité des actions.

Ce document détaille la mise en place d'une solution de centralisation de logs basée sur rsyslog, intégrée dans un environnement Active Directory existant, avec visualisation via le stack ELK (Elasticsearch, Filebeat, Kibana).

1.1 Architecture de l'infrastructure

Machine	Rôle	IP	Type
SRV-AD	Contrôleur de Domaine Active Directory + DNS	192.168.10.10	Statique
Ubulog (SRV-RSYSLOG)	Serveur de logs rsyslog + ELK Stack	192.168.10.1	Statique
Clsyslog	Client Linux	192.168.10.x	DHCP
SRV-RSYSLOG.srvlog.lan	Serveur Windows + NXLog	192.168.10.10	Statique

2. Mise à Jour du Système

Avant toute installation, il est indispensable de mettre à jour le système pour s'assurer d'avoir les derniers correctifs de sécurité et les versions les plus récentes des paquets.

2.1 Pourquoi mettre à jour ?

- Corriger les failles de sécurité connues
- Obtenir les dernières versions des paquets
- Eviter les conflits de dépendances lors des installations
- Garantir la stabilité du système

2.2 Commandes de mise à jour

```
# Mettre à jour la liste des paquets disponibles
sudo apt update

# Mettre à jour tous les paquets installes
sudo apt upgrade -y

# Optionnel : mettre à jour la distribution
sudo apt dist-upgrade -y

# Supprimer les paquets inutiles
sudo apt autoremove -y
```

```
Traitement des actions différées (« triggers ») pour initramfs-tools (0.140ubunt
u13.5) ...
update-initramfs: Generating /boot/initrd.img-6.8.0-106-generic
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@ubulog:/home/luc#
```

2.3 Explication des commandes

Commande	Description
Apt update	Télécharge la liste des paquets depuis les dépôts. Ne met RIEN à jour.
Apt upgrade -y	Installe les nouvelles versions de tous les paquets installes.
Apt dist-upgrade -y	Comme upgrade mais peut aussi installer/supprimer des paquets.
Apt autoremove -y	Supprime les paquets qui ne sont plus nécessaires.

> **Important : toujours faire apt update AVANT apt upgrade. Sans cela, apt ne connait pas les nouvelles versions disponibles.**

3. Configuration Réseau Statique

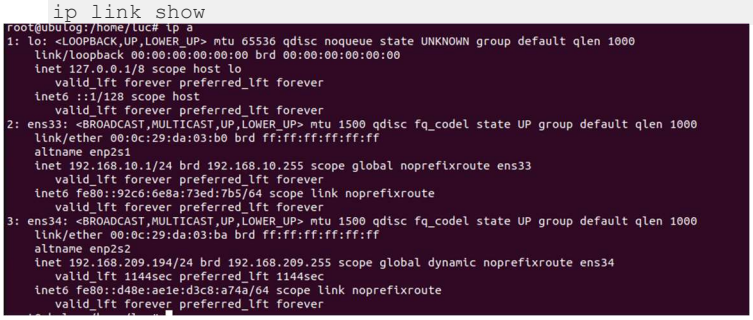
Pour assurer une communication stable et une résolution de nom cohérente via le DNS de l'Active Directory, une adresse IP statique est assignée au serveur de logs. Sans IP statique, l'adresse IP pourrait changer a chaque redémarrage, rendant la centralisation des logs instable.

3.1 Pourquoi une IP statique ?

- Les clients rsyslog envoient leurs logs a une IP fixe - elle ne doit pas changer
- Kibana et Elasticsearch doivent toujours être accessibles a la même adresse
- Le DNS du domaine Active Directory pointe vers cette IP

3.2 Identifier l'interface réseau

```
# Lister les interfaces réseau disponibles
ip a

# Ou avec cette commande
ip link show
```



Les interfaces réseau sont généralement nommées ens33, ens37, eth0, etc. Noter le nom de l'interface à configurer.

3.3 Configuration Netplan - /etc/netplan/01-netcfg.yaml

Ubuntu utilise Netplan pour la configuration réseau. Le fichier YAML définit les parametres de chaque interface.

```
network:
  version: 2
  renderer: networkd
  ethernets:
    ens33:
      dhcp4: false
      addresses:
        - 192.168.10.1/24
      nameservers:
        addresses:
          - 192.168.10.10
        search:
          - srvlog.lan
    ens37:
      dhcp4: true
```

3.4 Explication de la configuration

Parametre	Valeur	Description
dhcp4: false	-	Désactive le DHCP - on gère l'IP manuellement
addresses	192.168.10.1/24	Adresse IP statique avec masque /24 (255.255.255.0)
nameservers	192.168.10.10	Serveur DNS = le contrôleur de domaine AD
search	srvlog.lan	Domaine de recherche DNS - permet d'écrire 'ubulog' au lieu de 'ubulog.srvlog.lan'
ens37: dhcp4: true	-	Carte NAT en DHCP pour l'accès internet

3.5 Appliquer la configuration

```
# Valider la syntaxe YAML avant d'appliquer
sudo netplan generate

# Appliquer la configuration réseau
sudo netplan apply

# Vérifier que l'IP est bien configurée
ip a show ens33

# Vérifier la connectivité avec le serveur AD
ping 192.168.10.10
```

```
root@ubulog:/home/luc# ip a show ens33
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:da:03:b0 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.10.1/24 brd 192.168.10.255 scope global ens33
       valid_lft forever preferred_lft forever
    inet6 fe80::92c6:6e8a:73ed:7b5/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
root@ubulog:/home/luc# ping 192.168.10.10
PING 192.168.10.10 (192.168.10.10) 56(84) bytes of data.
64 bytes from 192.168.10.10: icmp_seq=25 ttl=128 time=3.33 ms
64 bytes from 192.168.10.10: icmp_seq=26 ttl=128 time=1.13 ms
64 bytes from 192.168.10.10: icmp_seq=27 ttl=128 time=2.54 ms
64 bytes from 192.168.10.10: icmp_seq=28 ttl=128 time=1.92 ms
^X64 bytes from 192.168.10.10: icmp_seq=29 ttl=128 time=1.66 ms
^C
--- 192.168.10.10 ping statistics ---
```

Attention : une erreur dans le fichier YAML peut couper l'acces reseau. Toujours valider avec 'netplan generate' avant d'appliquer.

4. Configuration SSH

SSH (Secure Shell) est le protocole utilisé pour administrer le serveur à distance de manière sécurisée. Sa configuration est essentielle pour permettre les connexions des utilisateurs du domaine Active Directory.

4.1 Installation et démarrage de SSH

```
# Installer le serveur SSH
sudo apt install openssh-server -y

# Démarrer et activer SSH au boot
sudo systemctl enable ssh
sudo systemctl start ssh

# Vérifier l'état
sudo systemctl status ssh
```

```
root@ubulog:/home/luc# systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
     Active: active (running) since Sat 2026-03-28 11:33:34 CET; 21s ago
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 40190 (sshd)
      Tasks: 1 (limit: 9380)
     Memory: 1.7M
        CPU: 25ms
     CGroup: /system.slice/ssh.service
             └─40190 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

mars 28 11:33:34 ubulog systemd[1]: Starting OpenBSD Secure Shell server...
mars 28 11:33:34 ubulog sshd[40190]: Server listening on 0.0.0.0 port 22.
mars 28 11:33:34 ubulog sshd[40190]: Server listening on :: port 22.
mars 28 11:33:34 ubulog systemd[1]: Started OpenBSD Secure Shell server.
```

4.2 Configuration SSH - /etc/ssh/sshd_config

Le fichier sshd_config contrôle le comportement du serveur SSH. Voici les paramètres importants a configurer :

```
# Editer le fichier de configuration SSH
sudo nano /etc/ssh/sshd_config
```

Paramètres a vérifier/modifier :

```
# Port d'écoute SSH (22 par defaut)
Port 22

# Autoriser l'authentification par mot de passe
PasswordAuthentication yes

# Autoriser la connexion root (désactiver en production)
PermitRootLogin no

# Autoriser les utilisateurs AD
UsePAM yes

# Autoriser la création de session pour les utilisateurs AD
AcceptEnv LANG LC_*

# Activer les sous-systemes sftp
Subsystem sftp /usr/lib/openssh/sftp-server
```

```
# Port d'ecoute SSH (22 par defaut)
Port 22

# Autoriser l'authentification par mot de passe
PasswordAuthentication yes

# Autoriser la connexion root (desactiver en production)
PermitRootLogin yes

# Autoriser les utilisateurs AD
UsePAM yes

# Autoriser la creation de session pour les utilisateurs AD
AcceptEnv LANG LC_*

# Activer les sous-systemes sftp
Subsystem sftp /usr/lib/openssh/sftp-server
```

4.3 Autoriser les utilisateurs du domaine via SSH

Pour que les utilisateurs du domaine srvlog.lan puissent se connecter via SSH, verifier que PAM est bien configure :

```
# Vérifier que pam_sss est présent dans la config PAM SSH
cat /etc/pam.d/sshd | grep sss

# Resultat attendu :
# @include common-auth  (qui inclut pam_sss.so)
```

4.4 Appliquer les modifications SSH

```
# Valider la configuration SSH
sudo sshd -t

# Redémarrer SSH pour appliquer les changements
sudo systemctl restart ssh

# Vérifier que SSH écoute bien
sudo ss -tlnp | grep 22
```

```
root@ubulog:/home/luc# ss -tlnp | grep 22
LISTEN 0      128        127.0.0.1:631      0.0.0.0:*    users:(("cupsd",pid=29922,fd=7))
LISTEN 0      128          0.0.0.0:22       0.0.0.0:*    users:(("sshd",pid=40474,fd=3))
LISTEN 0      128             [::]:22          [::]:*    users:(("sshd",pid=40474,fd=4))
LISTEN 0      128           [::1]:631          [::]:*    users:(("cupsd",pid=29922,fd=6))
```

4.5 Connexion SSH depuis Windows (MobaXterm)

Pour se connecter depuis Windows via MobaXterm :

Paramètre	Valeur
Session type	SSH
Host	192.168.10.1 (IP interne) ou 192.168.209.180 (IP NAT)
Port	22
Username	lkonesyslog@srvlog.lan

> **Si connexion via NAT VMware : utiliser l'IP 192.168.209.180 directement. Le port forwarding n'est pas nécessaire si les deux cartes réseau sont actives.**

5. Intégration a l'Active Directory

L'intégration d'Ubuntu au domaine Active Directory permet aux utilisateurs du domaine de se connecter sur le serveur Linux avec leurs comptes Windows. Cela centralise la gestion des identites et simplifie l'administration.

5.1 Installation des paquets nécessaires

```
sudo apt update
sudo apt install realmd sssd sssd-tools adcli samba-common-bin packagekit krb5-user -y
```

Paquet	Rôle
realmd	Découvre et rejoint le domaine AD
sssd	Gere l'authentification AD et le cache local
sssd-tools	Outils de gestion SSSD (sssctl, sss_cache)
adcli	Cree le compte machine dans le domaine
samba-common-bin	Fournit net ads pour les opérations AD
krb5-user	Gere les tickets Kerberos pour l'authentification
packagekit	Requis par realmd pour l'installation de paquets

5.2 Vérifier la détection du domaine AD

```
# Découvrir le domaine
realm discover srvlog.lan

# Resultat attendu :
# srvlog.lan
#   type: kerberos
#   realm-name: SRVLOG.LAN
#   domain-name: srvlog.lan
#   configured: no
#   server-software: active-directory
```

```
root@ubulog:/home/luc# realm discover srvlog.lan
srvlog.lan
  type: kerberos
  realm-name: SRVLOG.LAN
  domain-name: srvlog.lan
  configured: kerberos-member
  server-software: active-directory
  client-software: sssd
  required-package: sssd-tools
  required-package: sssd
  required-package: libnss-sss
  required-package: libpam-sss
  required-package: adcli
  required-package: samba-common-bin
  login-formats: %U@srvlog.lan
  login-policy: allow-realm-logins
```

Si rien ne s'affiche : vérifier le DNS avec 'cat /etc/resolv.conf' - la ligne nameserver doit pointer vers 192.168.10.10 (SRV-AD).

5.3 Joindre Ubuntu au domaine AD

```
# Rejoindre le domaine srvlog.lan
sudo realm join srvlog.lan -U Administrateur

# Entrer le mot de passe administrateur AD quand demande

# Vérifier que la jonction a réussi
realm list
```

Resultat attendu après realm list :

```
srvlog.lan
  type: kerberos
  realm-name: SRVLOG.LAN
  domain-name: srvlog.lan
  configured: kerberos-member
  server-software: active-directory
  client-software: sssd
```

5.4 Vérifier la visibilité des comptes AD

```
# Vérifier qu'un compte AD est visible
id "lkonesyslog@srvlog.lan"

# Resultat attendu :
# uid=1090201112(lkonesyslog@srvlog.lan) gid=1090200513(utilisateurs du
domaine@srvlog.lan)
```

```
root@ubulog:/home/luc# id lkonesyslog@srvlog.lan
uid=1090201112(lkonesyslog@srvlog.lan) gid=1090200513(utilisateurs du domaine@srvlog.lan)
.lan)
```

5.5 Creer le répertoire personnel automatiquement

Pour que les utilisateurs AD obtiennent automatiquement un dossier /home lors de leur premiere connexion :

```
# Activer la création automatique du home directory
sudo pam-auth-update --enable mkhomedir

# Ou manuellement dans le fichier PAM
sudo nano /etc/pam.d/common-session
# Ajouter la ligne :
# session required pam_mkhomedir.so skel=/etc/skel/ umask=0077
```

5.6 Donner les droits administrateur sudo

```
# Ouvrir un fichier sudoers dédié
sudo visudo -f /etc/sudoers.d/ad_admins

# Ajouter pour un utilisateur précis :
lkonesyslog@srvlog.lan ALL=(ALL:ALL) ALL

# Ou pour tout un groupe AD :
%admins\ du\ domaine@srvlog.lan ALL=(ALL:ALL) ALL

# Vérifier la syntaxe
sudo visudo -c
# Resultat attendu : sudo: parsed OK
```

```
root@ubulog:/home/luc# visudo -c
/etc/sudoers : analyse réussie
/etc/sudoers.d/README : analyse réussie
/etc/sudoers.d/admins : analyse réussie
```

5.7 Configuration SSSD - /etc/sssd/sssd.conf

SSSD (System Security Services Daemon) gère l'authentification AD et le cache local. Ce fichier est générer automatiquement par realm join mais peut nécessiter des ajustements :

```
[sssd]
domains = srvlog.lan
config_file_version = 2
services = nss, pam

[domain/srvlog.lan]
id_provider = ad
auth_provider = ad
access_provider = ad
ad_domain = srvlog.lan
krb5_realm = SRVLOG.LAN
ldap_id_mapping = True
use_fully_qualified_names = True
cache_credentials = True
krb5_store_password_if_offline = True
krb5_renew_interval = 300
default_shell = /bin/bash
fallback_homedir = /home/%u@%d
realmd_tags = manages-system joined-with-adcli
```

```
# Appliquer les permissions correctes (obligatoire)
sudo chmod 600 /etc/sssd/sssd.conf
sudo chown root:root /etc/sssd/sssd.conf

# Vérifier la configuration
sudo sssctl config-check

# Redémarrer SSSD
sudo systemctl restart sssd
sudo systemctl status sssd --no-pager
```

```
root@ubulog:/home/luc# sudo systemctl status sssd
● sssd.service - System Security Services Daemon
     Loaded: loaded (/lib/systemd/system/sssd.service; enabled; vendor preset: enabled)
     Active: active (running) since Sat 2026-03-28 11:53:44 CET; 6s ago
   Main PID: 41796 (sssd)
      Tasks: 4 (limit: 9380)
     Memory: 46.1M
        CPU: 324ms
     CGroup: /system.slice/sssd.service
             ├─41796 /usr/sbin/sssd -i --logger=files
             ├─41797 /usr/libexec/sssd/sssd_be --domain srvlog.lan --uid 0 --gid 0 --logger=files
             ├─41798 /usr/libexec/sssd/sssd_nss --uid 0 --gid 0 --logger=files
             └─41799 /usr/libexec/sssd/sssd_pam --uid 0 --gid 0 --logger=files

mars 28 11:53:44 ubulog systemd[1]: Starting System Security Services Daemon...
mars 28 11:53:44 ubulog sssd[41796]: Starting up
mars 28 11:53:44 ubulog sssd_be[41797]: Starting up
mars 28 11:53:44 ubulog sssd_nss[41798]: Starting up
mars 28 11:53:44 ubulog sssd_pam[41799]: Starting up
mars 28 11:53:44 ubulog systemd[1]: Started System Security Services Daemon.
```

5.8 Test de connexion avec un compte AD

```
# Test connexion locale
su - "lkonesyslog@srvlog.lan"

# Test connexion SSH depuis une autre machine
ssh lkonesyslog@srvlog.lan@192.168.10.1

# Vérifier l'identité
whoami
# Resultat attendu : lkonesyslog@srvlog.lan

# Vérifier les droits sudo
sudo whoami
# Resultat attendu : root
```

```
lkonesyslog@srvlog.lan@ubulog:~$ whoami
lkonesyslog@srvlog.lan
```

5.9 Persistance après redémarrage

```
# Redémarrer le serveur
sudo reboot

# Apres redémarrage, se reconnecter et tester :
sudo whoami
# Résultat attendu : root - la configuration est persistante
```

```
root@ubulog:/home/lkonesyslog@srvlog.lan# whoami
root
```

5.10 Commandes utiles de référence

Commande	Description
realm list	Vérifie la jonction au domaine AD
id "user@domaine"	Liste les groupes de l'utilisateur AD
sudo systemctl status sssd	Etat du service d'authentification
sudo visudo -c	Vérifie la syntaxe du fichier sudoers
sudo sss_cache -E	Vide le cache SSSD (utile si compte non reconnu)
kinit user@SRVLOG.LAN	Obtenir un ticket Kerberos manuellement
klist	Lister les tickets Kerberos actifs
realm leave	Quitter le domaine AD

PARTIE 2 - CENTRALISATION DES LOGS AVEC RSYSLOG ET ELK

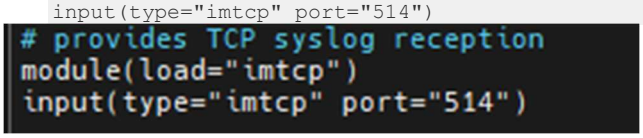
6. Configuration rsyslog

rsyslog est le service de centralisation des logs. Il reçoit les logs de toutes les machines via TCP sur le port 514, les trie par catégorie et les stocke en format JSON pour Filebeat.

6.1 Activation de la réception réseau - /etc/rsyslog.conf

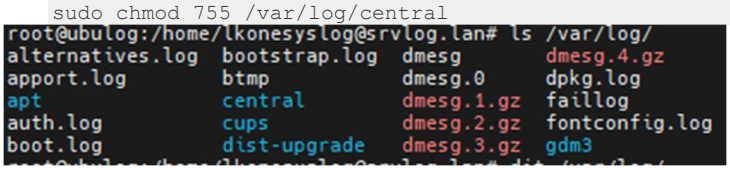
Pour recevoir les logs d'autres machines, décommenter ces lignes dans /etc/rsyslog.conf :

```
module(load="imtcp")
input(type="imtcp" port="514")
```



6.2 Creer la structure de stockage

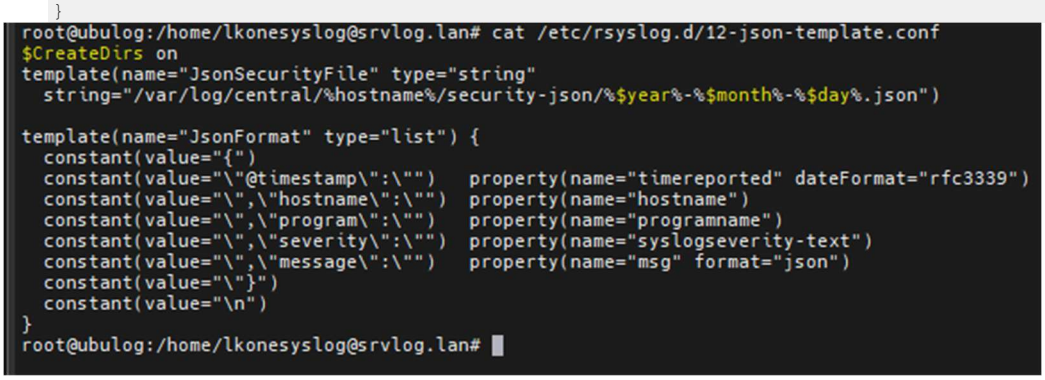
```
# Creer le dossier racine des logs centralises
sudo mkdir -p /var/log/central
sudo chown syslog:adm /var/log/central
sudo chmod 755 /var/log/central
```



6.3 Template JSON - /etc/rsyslog.d/12-json-template.conf

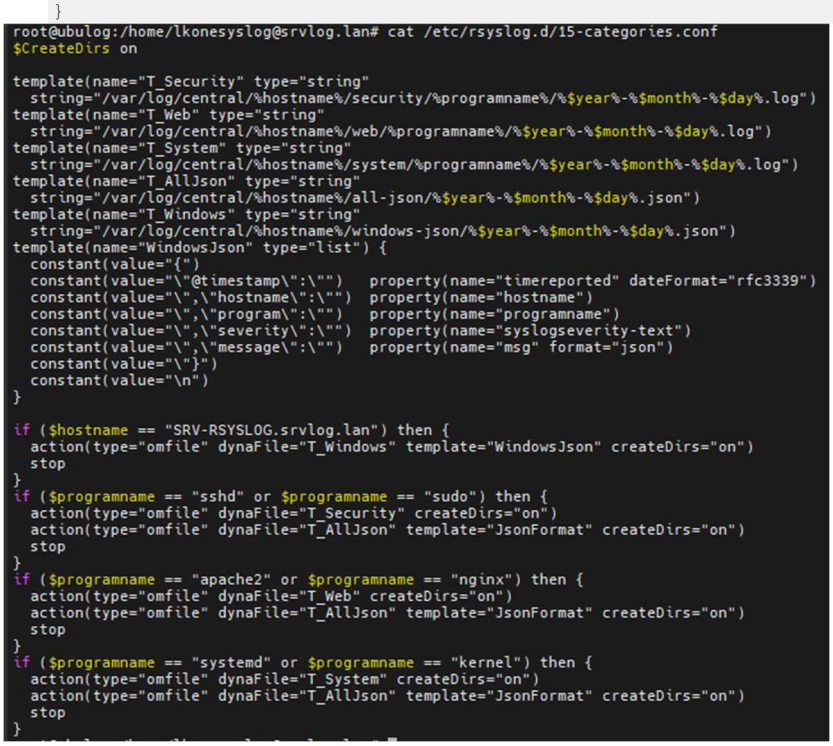
Ce fichier définit le format JSON de chaque log. Le \n final est crucial pour que Filebeat puisse lire un objet JSON par ligne.

```
$CreateDirs on
template(name="JsonSecurityFile" type="string"
  string="/var/log/central/%hostname%/security-json/%$year%-%$month%-%$day%.json")

template(name="JsonFormat" type="list") {
  constant(value="{")
  constant(value="\"@timestamp\":\"")   property(name="timereported" dateFormat="rfc3339")
  constant(value="\",\"hostname\":\"")  property(name="hostname")
  constant(value="\",\"program\":\"")   property(name="programname")
  constant(value="\",\"severity\":\"")  property(name="syslogseverity-text")
  constant(value="\",\"message\":\"")   property(name="msg" format="json")
  constant(value="\"}")
  constant(value="\n")
}
```



6.4 Catégories de logs - /etc/rsyslog.d/15-categories.conf

```
$CreateDirs on

template(name="T_Security" type="string"
  string="/var/log/central/%hostname%/security/%programname%/%$year%-%$month%-%$day%.log")
template(name="T_Web" type="string"
  string="/var/log/central/%hostname%/web/%programname%/%$year%-%$month%-%$day%.log")
template(name="T_System" type="string"
  string="/var/log/central/%hostname%/system/%programname%/%$year%-%$month%-%$day%.log")
template(name="T_AllJson" type="string"
  string="/var/log/central/%hostname%/all-json/%$year%-%$month%-%$day%.json")
template(name="T_Windows" type="string"
  string="/var/log/central/%hostname%/windows-json/%$year%-%$month%-%$day%.json")
template(name="WindowsJson" type="list") {
  constant(value="{")
  constant(value="\"@timestamp\":\"")   property(name="timereported" dateFormat="rfc3339")
  constant(value="\",\"hostname\":\"")  property(name="hostname")
  constant(value="\",\"program\":\"")   property(name="programname")
  constant(value="\",\"severity\":\"")  property(name="syslogseverity-text")
  constant(value="\",\"message\":\"")   property(name="msg" format="json")
  constant(value="\"}")
  constant(value="\n")
}

if ($hostname == "SRV-RSYSLOG.srvlog.lan") then {
  action(type="omfile" dynaFile="T_Windows" template="WindowsJson" createDirs="on")
  stop
}
if ($programname == "sshd" or $programname == "sudo") then {
  action(type="omfile" dynaFile="T_Security" createDirs="on")
  action(type="omfile" dynaFile="T_AllJson" template="JsonFormat" createDirs="on")
  stop
}
if ($programname == "apache2" or $programname == "nginx") then {
  action(type="omfile" dynaFile="T_Web" createDirs="on")
  action(type="omfile" dynaFile="T_AllJson" template="JsonFormat" createDirs="on")
  stop
}
if ($programname == "systemd" or $programname == "kernel") then {
  action(type="omfile" dynaFile="T_System" createDirs="on")
  action(type="omfile" dynaFile="T_AllJson" template="JsonFormat" createDirs="on")
  stop
}
```



6.5 Centralisation generale - /etc/rsyslog.d/99-central-logs.conf

```
$CreateDirs on
template(
  name="CentralLogs"
  type="string"
  string="/var/log/central/%HOSTNAME%/%programname%/%$year%-%$month%-%$day%.log"
)
*.* ?CentralLogs
& stop
```

```
root@ubulog:/home/lkonesyslog@srvlog.lan# cat /etc/rsyslog.d/99-central-logs.conf
$CreateDirs on
template(
  name="CentralLogs"
  type="string"
  string="/var/log/central/%HOSTNAME%/%programname%/%$year%-%$month%-%$day%.log"
)
*.* ?CentralLogs
& stop
```

6.6 Configuration des clients rsyslog

Sur chaque machine cliente Linux, ajouter ce fichier pour envoyer les logs vers le serveur :

```
# Fichier /etc/rsyslog.d/90-forward-to-central.conf
*.* @@192.168.10.1:514

# Redémarrer rsyslog sur le client
sudo systemctl restart rsyslog
```

```
root@clsyslog:/home/luc# cat /etc/rsyslog.d/90-forward-to-central.conf
*.* @@192.168.10.1:514
```

6.7 Validation et redémarrage

```
# Valider la configuration
sudo rsyslogd -N1

# Redémarrer rsyslog
sudo systemctl restart rsyslog

# Creer le dossier central avec les bonnes permissions
# (obligatoire - $CreateDirs on ne crée pas le dossier racine)
sudo mkdir -p /var/log/central
sudo chown syslog:adm /var/log/central
sudo chmod 755 /var/log/central

# Redémarrer rsyslog après création du dossier
sudo systemctl restart rsyslog

# Tester avec un log manuel
logger -p auth.info -t sshd "TEST CONNEXION"
sleep 2

# Vérifier que les dossiers ont été crées
sudo ls /var/log/central/ubulog/

# Vérifier le contenu JSON
sudo tail -5 /var/log/central/ubulog/all-json/$(date +%Y-%m-%d).json
```

```
root@ubulog:/home/lkonesyslog@srvlog.lan# tree /var/log/central/
/var/log/central/
├── SRV-RSYSLOG.srvlog.lan
│   └── windows-json
│       └── 2026-03-28.json
└── ubulog
    ├── all-json
    │   └── 2026-03-28.json
    ├── rsyslogd
    │   └── 2026-03-28.log
    ├── security
    │   ├── sshd
    │   │   └── 2026-03-28.log
    │   └── sudo
    │       └── 2026-03-28.log
    └── system
        └── systemd
            └── 2026-03-28.log
```

7. Installation Elasticsearch

7.1 Installation

```
# Ajouter le dépôt Elastic
wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | \
  sudo gpg --dearmor -o /usr/share/keyrings/elasticsearch-keyring.gpg
echo "deb [signed-by=/usr/share/keyrings/elasticsearch-keyring.gpg] \
  https://artifacts.elastic.co/packages/8.x/apt stable main" | \
  sudo tee /etc/apt/sources.list.d/elastic-8.x.list
sudo apt update && sudo apt install elasticsearch -y
```

```
1 055 ko réceptionnés en 1s (711 ko/s)
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Tous les paquets sont à jour.
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
elasticsearch est déjà la version la plus récente (8.19.13).
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
```

Le mot de passe elastic est générer automatiquement a l'installation. Le noter impérativement dans un endroit sécurise.

7.2 Démarrage et activation

```
sudo systemctl daemon-reload
sudo systemctl enable elasticsearch
sudo systemctl start elasticsearch

# Verification
curl -k -u elastic:"MOT_DE_PASSE" "https://localhost:9200"
```

```
root@ubulog:/home/lkonesyslog@srvlog.lan# curl -k -u elastic:"=qdRry31ufpF_beGXYE6" "https://local
host:9200"
{
  "name" : "ubulog",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "IxeS-VViQSy75DDyuYTMHw",
  "version" : {
    "number" : "8.19.13",
    "build_flavor" : "default",
    "build_type" : "deb",
    "build_hash" : "f6279a2167a248af2bb287409231fb0cf52aa0db",
    "build_date" : "2026-03-16T10:08:32.007626438Z",
    "build_snapshot" : false,
    "lucene_version" : "9.12.2",
    "minimum_wire_compatibility_version" : "7.17.0",
    "minimum_index_compatibility_version" : "7.0.0"
  },
  "tagline" : "You Know, for Search"
}
```

7.3 Configuration snapshots - /etc/elasticsearch/elasticsearch.yml

```
# Ajouter a la fin du fichier
path.repo: ["/var/backups/elasticsearch"]
```

```
path.repo: ["/var/backups/elasticsearch"]
```

8. Installation Kibana

8.1 Installation et configuration

```
sudo apt install kibana -y
```

```
Creating kibana group... OK
Creating kibana user... OK
Kibana is currently running with legacy OpenSSL providers enabled! For details and instructions on
 how to disable see https://www.elastic.co/guide/en/kibana/8.19/production.html#openssl-legacy-pro
vider
Created Kibana keystore in /etc/kibana/kibana.keystore
```

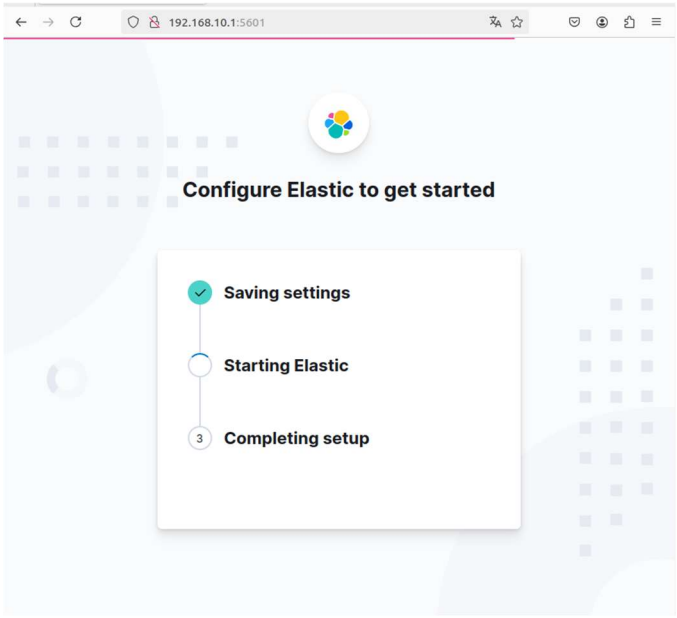
Configuration dans /etc/kibana/kibana.yml :

```
server.host: 0.0.0.0
xpack.encryptedSavedObjects.encryptionKey: "cle_32_caracteres_minimum_obligatoire"
xpack.reporting.encryptionKey: "cle_32_caracteres_minimum_obligatoire"
xpack.security.encryptionKey: "cle_32_caracteres_minimum_obligatoire"


# Générer 3 clés aléatoires de 32 caractères
openssl rand -hex 32
openssl rand -hex 32
openssl rand -hex 32
```

```
server.host: 0.0.0.0
xpack.encryptedSavedObjects.encryptionKey: "f5c51841c6154e264611f0e6c9ff98c68a7fbb085ad2d0bc1ef0672cda3e858e"
xpack.reporting.encryptionKey: "7da454cb25abff5a7ac890f6963f743439d969dc60af845f96f8ff3f24a978ea"
xpack.security.encryptionKey: "d151c00f4b3710f1acfd082068b7abddd958ade0e3509949d9317e764520f9d2"
```

8.2 Connexion a Elasticsearch

```
# Générer un token d'enrôlement
sudo /usr/share/elasticsearch/bin/elasticsearch-create-enrollment-token -s kibana

# Coller le token dans l'interface Kibana lors du premier démarrage
# Puis obtenir le code de verification :
sudo /usr/share/kibana/bin/kibana-verification-code
```



8.3 Permissions certificats

```
sudo chmod 755 /etc/elasticsearch/
sudo chmod 755 /etc/elasticsearch/certs/
sudo chmod 644 /etc/elasticsearch/certs/http_ca.crt
sudo usermod -aG elasticsearch kibana
sudo systemctl restart kibana
```

Accès Kibana : http://192.168.10.1:5601 - Uniquement depuis le réseau interne srvlog.lan

9. Installation et Configuration Filebeat

9.1 Installation

```
sudo apt install filebeat -y
```

9.2 Configuration - /etc/filebeat/filebeat.yml

```
filebeat.inputs:
- type: log
  enabled: true
  paths:
    - /var/log/central/**/all-json/*.json
  json.keys_under_root: true
  json.add_error_key: true
  json.message_key: message

- type: log
  enabled: true
  paths:
    - /var/log/central/**/windows-json/*.json
  json.keys_under_root: true
  json.add_error_key: true
  json.message_key: Message

output.elasticsearch:
  hosts: ["https://localhost:9200"]
  username: "elastic"
  password: "MOT_DE_PASSE"
  ssl.verification_mode: none
  index: "rsyslog-logs"

setup.ilm.enabled: false
setup.template.name: "rsyslog-logs"
setup.template.pattern: "rsyslog-logs"
```

Ajouter le mot de passe crée par elasticsearch

9.3 Démarrage

```
sudo systemctl enable filebeat
sudo systemctl start filebeat

# Vérifier que les logs arrivent dans Elasticsearch
curl -k -u elastic:"MOT_DE_PASSE" "https://localhost:9200/rsyslog-logs/_count"
```

9.4 Mise à jour du mot de passe dans la configuration

```
Si le mot de passe elastic a été réinitialisé, mettre à jour la configuration Filebeat :
bash
# Editer la configuration Filebeat
sudo nano /etc/filebeat/filebeat.yml

# Remplacer la ligne password par le nouveau mot de passe :
# password : "NOUVEAU_MOT_DE_PASSE"

# Vérifier que le mot de passe est bien mis a jour
sudo grep "password" /etc/filebeat/filebeat.yml

# Redémarrer Filebeat pour appliquer les changements
sudo systemctl restart filebeat
sleep 15

# Vérifier que l'index est bien crée dans Elasticsearch
curl -k -u elastic:"MOT_DE_PASSE" "https://localhost:9200/_cat/indices?v" | grep rsyslog
```

Important : A chaque réinitialisation du mot de passe elastic, il faut mettre à jour le mot de passe dans /etc/filebeat/filebeat.yml ET dans /etc/kibana/kibana.yml sinon Filebeat et Kibana ne peuvent plus se connecter à Elasticsearch.

10. Sécurité et Pare-feu (UFW)

```
# SSH
sudo ufw allow 22/tcp

# Logs rsyslog
sudo ufw allow 514/tcp
sudo ufw allow 514/udp

# Elasticsearch
sudo ufw allow 9200/tcp

# Kibana - reseau interne uniquement
sudo ufw allow from 192.168.10.0/24 to any port 5601
sudo ufw allow from 192.168.209.0/24 to any port 5601

# Activer
sudo ufw enable
sudo ufw status verbose
```

```
Vers                       Action      De
----                       ------      --
22/tcp                     ALLOW IN    Anywhere
514/tcp                    ALLOW IN    Anywhere
514/udp                    ALLOW IN    Anywhere
9200/tcp                   ALLOW IN    Anywhere
5601                       ALLOW IN    192.168.10.0/24
5601                       ALLOW IN    192.168.209.0/24
22/tcp (v6)                ALLOW IN    Anywhere (v6)
514/tcp (v6)               ALLOW IN    Anywhere (v6)
514/udp (v6)               ALLOW IN    Anywhere (v6)
9200/tcp (v6)              ALLOW IN    Anywhere (v6)
```

11. Dashboard Kibana

Une fois les logs centralisés dans Elasticsearch via Filebeat, il est nécessaire de mettre en place une interface de visualisation permettant d'exploiter ces données de manière efficace. C'est le rôle de Kibana et de ses dashboards.

Kibana permet de transformer des milliers de lignes de logs bruts en graphiques interactifs, tableaux de bord et alertes automatiques. Plutôt que de parcourir manuellement les fichiers de logs sur chaque serveur, l'administrateur dispose d'une vue centralisée, en temps réel, de l'ensemble de l'infrastructure.

Dans cette section, nous allons créer deux dashboards complémentaires :

Dashboard Centralisation Logs — offre une vue globale de toutes les machines et de leur activité (ubulog, clsyslog, SRV-RSYSLOG).

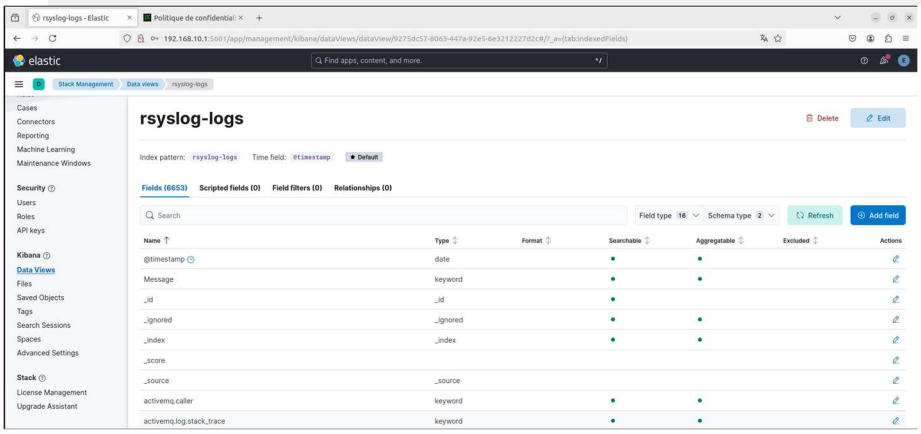
Dashboard Authentification SSH — se concentre sur les événements de sécurité liés aux connexions SSH, avec détection des tentatives de brute force en temps réel.

Nous configurerons également une alerte automatique qui se déclenche dès qu'un nombre anormal de tentatives de connexion SSH échouées est détecté, permettant une réaction rapide face à une potentielle attaque.

11.1 Création du Data View

Aller dans Stack Management > Data Views > Create data view :

```
Champ : Valeur
Name : rsyslog-logs
Index pattern : rsyslog-logs*
Timestamp field : @timestamp
```



Cliquer Save data view to Kibana.

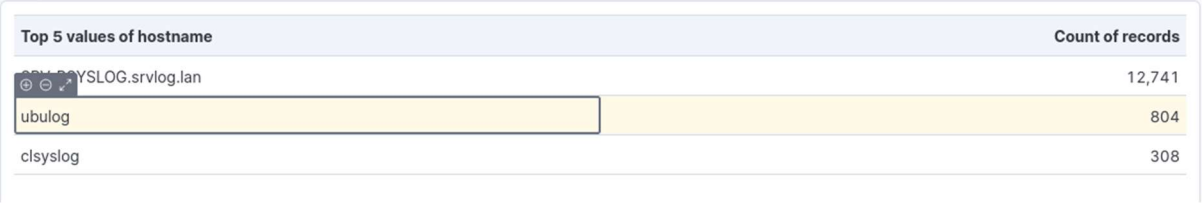
11.2 Dashboard Centralisation Logs

Aller dans **Analytics > Dashboard > Create dashboard**, nommer le dashboard : Dashboard Centralisation Logs et cliquer **Save**.

Visualisation 1 — Tableau des machines :

Cliquer Create visualization

```
Type : Table
Rows → hostname → Top values of hostname
Metric → Count of records
Période : Last 7 days
```



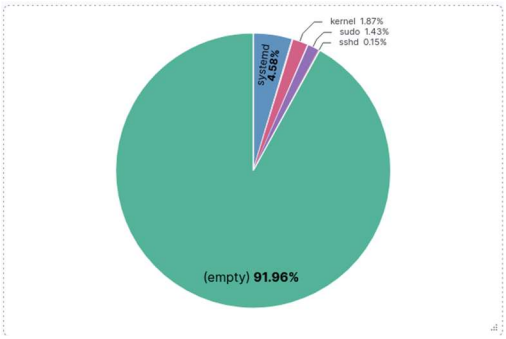
Cliquer Save and return

Visualisation 2 — Camembert par programme :

Cliquer Create visualization

```
Type : Pie
Slice by → program → Top values of program
Metric → Count of records
Période : Last 7 days
```

Cliquer Save and return

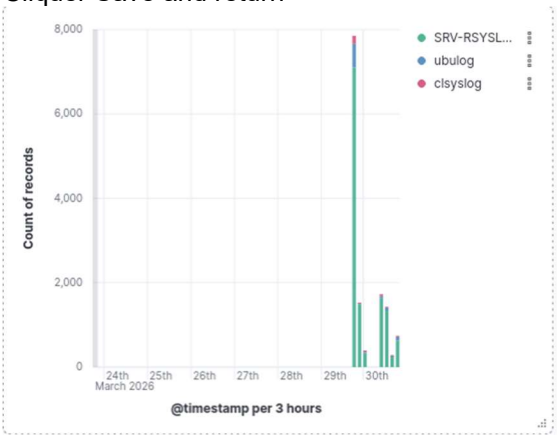


Visualisation 3 — Bar chart temporel :

Cliquer Create visualization

```
Type : Bar vertical
Horizontal axis → @timestamp
Vertical axis → Count of records
Breakdown → hostname
Période : Last 7 days
```

Cliquer Save and return



Cliquer **Save** pour sauvegarder le dashboard.

11.3 Dashboard Authentification SSH

Aller dans **Analytics > Dashboard > Create dashboard**, nommer le dashboard Dashboard Authentification et cliquer **Save**.

Visualisation 1 — Compteur SSH total :

Cliquer Create visualization

```
Type : Metric
KQL : program : "sshd"
Metric → Count of records
```

Cliquer Save and return

Visualisation 2 — Tableau sévérité SSH :

Cliquer Create visualization

```
Type : Table
KQL : program : "sshd"
Rows → severity → Top values of severity
Metric → Count of records
```

Cliquer Save and return

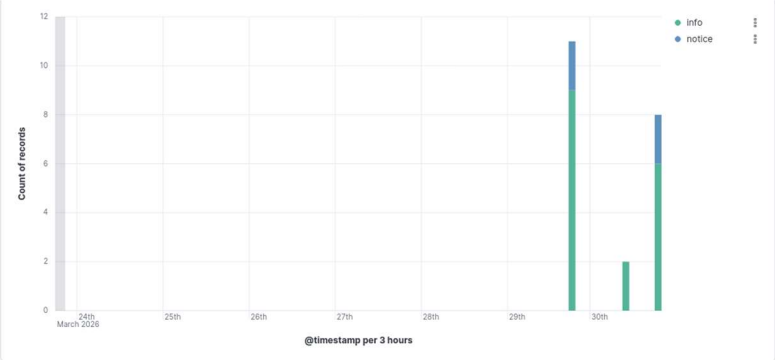
Top 5 values of severity	Count of records
info	17
notice	4

Visualisation 3 — Bar chart SSH dans le temps :

Cliquer Create visualization

```
Type : Bar vertical
KQL : program : "sshd"
Horizontal axis → @timestamp
Vertical axis → Count of records
Breakdown → severity
```

Cliquer Save and return



Visualisation 4 — Compteur Failed password :

Cliquer Create visualization

```
Type : Metric
KQL : message : "Failed password"
Metric → Count of records
```

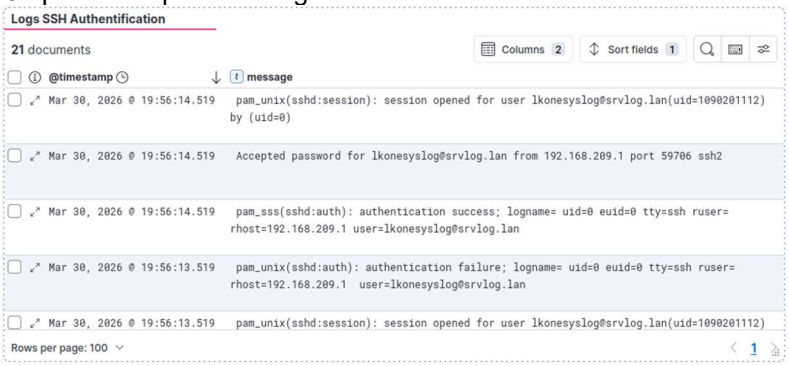
Cliquer Save and return

Vue Discover — Logs SSH en temps réel :

Aller dans Analytics > Discover

```
KQL : program : "sshd"
Ajouter la colonne message depuis le panneau gauche
Cliquer Save → nommer Logs SSH Authentication
Retourner sur le dashboard → Add from library → sélectionner Logs SSH Authentication
```

Cliquer Save pour sauvegarder le dashboard.

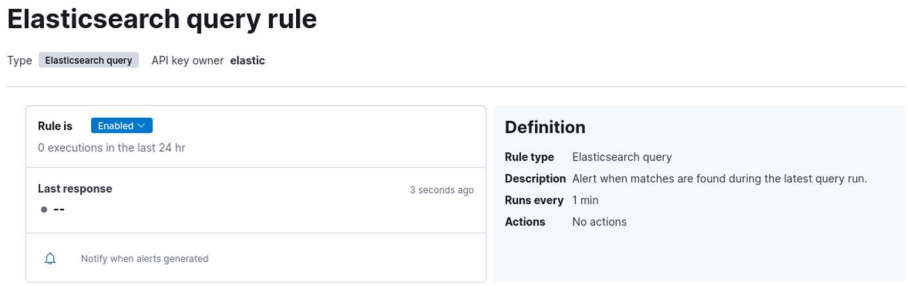


11.4 Création de l'alerte Brute Force SSH

Aller dans Stack Management > Rules > Create rule :

```
Paramètre : Valeur
Nom : Alerte Brute Force SSH
Type : Elasticsearch query
Query type : KQL
Query : program : "sshd" and message : "Failed password"
Condition : WHEN count() IS ABOVE 5 FOR THE LAST 1 minute
Fréquence Every 1 second
```

Cliquer Save.



11.5 Test de l'alerte

```
bash
# Simuler une attaque brute force SSH
for i in {1..20}; do
  logger -p auth.warning -t sshd "Failed password for hacker from 10.0.0.1 port 22 ssh2"
done
```

Vérifier dans **Stack Management > Rules > Alerte Brute Force SSH > onglet Alerts** — l'alerte doit passer au statut **Active** (rouge).

11.6 Activer le rafraîchissement automatique Kibana

Dans n'importe quel dashboard ou Discover, cliquer sur l'icône **Refresh** en haut à droite → **Set auto-refresh** → choisir **5 seconds**.

12. Archivage et Sauvegarde

12.1 Archivage logrotate - /etc/logrotate.d/central-logs

```
/var/log/central/**/all-json/*.json
/var/log/central/**/windows-json/*.json
{
    daily
    rotate 180
    compress
    delaycompress
    missingok
    notifempty
    dateext
    dateformat -%Y-%m-%d
    olddir /var/log/archives
    createolddir 755 syslog adm
    postrotate
        systemctl restart rsyslog > /dev/null 2>&1 || true
    endscript
}
```

12.2 Snapshots Elasticsearch

```
# Creer le depot
sudo mkdir -p /var/backups/elasticsearch
sudo chown elasticsearch:elasticsearch /var/backups/elasticsearch
curl -k -u elastic:"MOT_DE_PASSE" -X PUT "https://localhost:9200/_snapshot/backup_logs" \
  -H "Content-Type: application/json" \
  -d '{"type":"fs","settings":{"location":"/var/backups/elasticsearch","compress":true}}'

# redémarrer Elasticsearch
 sudo systemctl restart elasticsearch
sleep 20
curl -k -u elastic:"=qdRry31ufpF_beGXYE6" "https://localhost:9200/_cluster/health?pretty"
| grep status

# Script automatique - /etc/cron.daily/elasticsearch-snapshot

#!/bin/bash
DATE=$(date +%Y-%m-%d)
curl -s -k -u elastic:"MOT_DE_PASSE" -X PUT \
  "https://localhost:9200/_snapshot/backup_logs/snapshot_${DATE}" \
  -H "Content-Type: application/json" \
  -d '{"indices":"rsyslog-logs","ignore_unavailable":true,"include_global_state":false}'
```

13. Chemins Importants

13.1 Fichiers de configuration

Chemin	Description
/etc/netplan/01-netcfg.yaml	Configuration reseau statique
/etc/ssh/sshd_config	Configuration du serveur SSH
/etc/sssd/sssd.conf	Configuration SSSD (authentification AD)
/etc/sudoers.d/ad_admins	Droits sudo pour les utilisateurs AD
/etc/rsyslog.conf	Configuration principale rsyslog
/etc/rsyslog.d/12-json-template.conf	Template JSON pour Filebeat/Kibana
/etc/rsyslog.d/15-categories.conf	Regles de tri des logs par categorie
/etc/rsyslog.d/99-central-logs.conf	Centralisation de tous les logs
/etc/filebeat/filebeat.yml	Configuration Filebeat
/etc/elasticsearch/elasticsearch.yml	Configuration Elasticsearch

Chemin	Description
/etc/kibana/kibana.yml	Configuration Kibana
/etc/logrotate.d/central-logs	Configuration archivage logrotate
/etc/cron.daily/elasticsearch-snapshot	Script sauvegarde quotidienne

13.2 Dossiers de logs

Chemin	Description
/var/log/central/	Racine de tous les logs centralises
/var/log/central/ubulog/all-json/	Logs JSON Linux ubulog (pour Filebeat)
/var/log/central/clsyslog/all-json/	Logs JSON Linux clsyslog (pour Filebeat)
/var/log/central/SRV-RSYSLOG.srvlog.lan/windows-json/	Logs JSON Windows (pour Filebeat)
/var/log/central/ubulog/security/sshd/	Logs SSH bruts ubulog
/var/log/central/ubulog/system/	Logs systeme (kernel, systemd)
/var/log/archives/	Archives compressees .gz (6 mois)
/var/backups/elasticsearch/	Snapshots Elasticsearch
/var/lib/filebeat/registry/	Registre Filebeat

14. Problemes Rencontres et Solutions

14.1 - omelasticsearch incompatible avec Elasticsearch 8.x

Contexte

Le module rsyslog omelasticsearch (version 8.2112) ne supportait pas HTTPS avec Elasticsearch 8.x. Les logs n'etaient jamais envoyes sans aucune erreur visible.

Symptomes

- Index rsyslog absent dans Elasticsearch
- Aucun fichier d'erreur cree par action.errorfile
- omelasticsearch suspendu silencieusement au demarrage

Solution

```
sudo apt remove --purge rsyslog-elasticsearch -y
sudo apt install filebeat -y
```

Lecon : Utiliser Filebeat au lieu d'omelasticsearch pour Elasticsearch 8.x.

14.2 - Guillemets typographiques dans les configs rsyslog

Contexte

Des guillemets typographiques remplacaient les guillemets droits lors de l'edition avec nano. rsyslog ne peut pas les parser.

Symptomes

- Erreur : invalid character - is there an invalid escape sequence somewhere?
- rsyslog demarre mais n'ecrit plus aucun log

Solution

```
sudo tee /etc/rsyslog.d/15-categories.conf << 'EOF'
[contenu du fichier]
EOF
```

Lecon : Toujours utiliser tee avec heredoc pour ecrire des configs rsyslog.

14.3 - JSON mal forme - plusieurs objets sur une seule ligne

Contexte

Le template JsonFormat generait des fichiers JSON avec plusieurs objets colles sur une meme ligne. Filebeat s'attend a un objet JSON par ligne.

Symptomes

- Filebeat affichait events filtered:X au lieu de events acked:X
- Aucun document n'apparaissait dans Elasticsearch

Solution

```
constant(value="\"}\n")  # Ajouter \n a la fin du template JsonFormat
```

Lecon : Le \n est obligatoire dans le template rsyslog pour que Filebeat lise correctement.

14.4 - Kibana sans permissions sur le certificat SSL

Symptomes

- FATAL Error: EACCES: permission denied, open /etc/elasticsearch/certs/http_ca.crt

- Kibana ne demarre pas

Solution

```
sudo chmod 755 /etc/elasticsearch/certs/
sudo chmod 644 /etc/elasticsearch/certs/http_ca.crt
sudo usermod -aG elasticsearch kibana
sudo systemctl restart kibana
```

Lecon : Apres reinstallation d'Elasticsearch, toujours verifier les permissions sur les certificats.

14.5 - Cluster Elasticsearch en etat RED

Contexte

Apres un test de restauration, un index reste en etat RED car non replique sur un seul noeud.

Solution

```
curl -k -u elastic:"MOT_DE_PASSE" -X DELETE \
  "https://localhost:9200/_data_stream/restored_rsyslog-logs"
```

Lecon : Supprimer les index de test via DELETE _data_stream apres verification.

14.6 - Alertes Kibana - cle de chiffrement manquante

Symptomes

- Message : Additional setup required
- Message : You must configure an encryption key to use Alerting

Solution

```
# Ajouter dans /etc/kibana/kibana.yml :
xpack.encryptedSavedObjects.encryptionKey: "cle_32_caracteres_minimum"
xpack.reporting.encryptionKey: "cle_32_caracteres_minimum"
xpack.security.encryptionKey: "cle_32_caracteres_minimum"
sudo systemctl restart kibana
```

Lecon : Toujours configurer les 3 cles de chiffrement Kibana des l'installation.

14.7 - Alerte Brute Force SSH - règle ne se déclenche pas

Symptômes

```
L'alerte s'exécute sans erreur (status Succeeded) mais Active alerts reste à 0
Aucune notification ne se déclenche malgré des tentatives SSH échouées visibles dans les
logs
```

Solution

```
Erreur 1 — KQL incorrecte : "Failed password" cherché dans le champ program au lieu de
message
# Incorrect
program : "sshd" and program : "Failed password"

# Correct
program : "sshd" and message : "Failed password"
Erreur 2 — Fenêtre temporelle trop courte (5 secondes → impossible à atteindre)
# Correct
FOR THE LAST 5 minutes
```

Leçon : Toujours vérifier que chaque condition KQL cible le bon champ (program pour le processus, message pour le contenu), et dimensionner la fenêtre temporelle à la réalité d'une attaque (5 minutes minimum pour du brute force SSH).

Document genere dans le cadre du BTS SIO SISR
Infrastructure Linux & Active Directory - Centralisation des Logs - Mars 2026