jmy37
GitHub : https://github.com/jmy37/infrathome

GUIDE D'INSTALLATION	
Serveurs de journaux Syslog	
Infr@home_GUI-INST_05-Syslog	
Version 1.0	**Date d'application : 30/11/2020**
Projet/SI : Infr@home	

1. HISTORIQUE DES MODIFICATIONS

Version	Date	Auteur	Objet de la modification
1.0	30/11/2020	jmy37	Création du document

Tableau 1 : Historique des modifications

2. RÉFÉRENCES

2.1. Table des matières

2.2. Liste des tableaux

2.3. Liste des figures

Aucune entrée de table d'illustration n'a été trouvée.

3. PRÉAMBULE

Ce document décrit l'installation, l'exploitation et la résolution de pannes associées au composant « Syslog » de la solution « Infr@home ».

4. PRÉSENTATION

4.1. Le projet Rsyslog

Rsyslog est un serveur de centralisation de journaux d'évènements en source ouverte. Ce serveur gère aussi bien les journaux d'évènements de systèmes d'exploitation Windows et Linux que ceux de nombreuses applications tierces.

4.2. Configuration requise

La configuration requise est définie dans le Tableau 2.

Composant	Serveur virtuel	Serveur physique
Processeur	1 processeur 2 cœurs	1 processeur 4 cœurs 1,5GHz
Mémoire vive	4 Go	4 Go
Disque dur	35 Go (système) 200 Go (données)	64Go SSD (système) 256 Go SSD (données)

Tableau 2 : Configuration requise pour un serveur Syslog

Le nombre de clients ainsi que la volumétrie à traiter nécessitent d'adapter la configuration présentée ci-dessous. À chaque modification, une supervision doit être réalisée afin de s'assurer de ne pas surcharger le serveur.

Les ports présentés dans le Tableau 3 doivent être ouverts.

Protocole	Source		Destination		Explication
	Port	Adresse	Port	Adresse	
tcp	*	Clients	514	Syslog	Remontée de logs
udp	*	Clients	514	Syslog	Remontée de logs
tcp	*	Clients d'admin	22	Syslog	Administration via SSH

Tableau 3 : Matrice des flux

5. INSTALLATION

Le serveur est supposé installé avec CentOS 8 en configuration minimale.

5.1. Configuration du disque supplémentaire

Le disque supplémentaire est celui contenant les fichiers logs des systèmes clients.

```
pvcreate /dev/sdb
vgcreate VG_Opt_rsyslog /dev/sdb
lvcreate -n /dev/VG_Opt_rsyslog/LV_opt_rsyslog -l 100%FREE
mkfs -t xfs /dev/mapper/VG_Opt_rsyslog-LV_opt_rsyslog
mkdir /opt/rsyslog

echo -e "/dev/mapper/VG_Opt_rsyslog-LV_opt_rsyslog /opt/rsyslog xfs
defaults 0 0" >> /etc/fstab

mount -a
```

5.2. Configuration du pare-feu local et de SELinux

Les flux requis doivent être ouverts sur le pare-feu local.

```
firewall-cmd --add-service=ssh --permanent
firewall-cmd --add-port={514/tcp,514/udp} --permanent
firewall-cmd --reload
```

Les flux doivent également être permis via SELinux.

```
dnf -y install policycoreutils-python-utils
chcon -R system_u:object_r:var_log_t:s0 /opt/rsyslog/
semanage port -a -t syslogd_port_t -p udp 514
semanage port -a -t syslogd_port_t -p tcp 514
```

5.3. Configuration de Rsyslog

Rsyslog est préinstallé sur tous les serveurs Linux, y compris en installation minimale. Il n'y a qu'à le configurer.

- Commenter la remontée de logs en commentaire afin de ne traiter que les requêtes en UDP
- Activer la remontée de logs en TCP
- Créer un module de remontée
- Redémarrer le démon rsyslog
- Vérifier à l'aide de netstat que le serveur soit bien à l'écoute

```
echo -e '#module(load="imudp")
#input(type="imudp" port="514")
module(load="imtcp")
input(type="imtcp" port="514")' >> /etc/rsyslog.d/00-server.conf

systemctl restart rsyslog
netstat -tulpn | grep rsyslog
```

6. EXPLOITATION

6.1. Configuration de clients

6.1.1. Configuration d'un client Linux

6.1.1.1. Configuration serveur

Le serveur doit savoir quelle action réaliser avec les logs reçus.

- Créer une règle permettant d'interpréter les logs d'audit
- Créer une règle permettant d'interpréter les logs d'authentification
- Créer une règle permettant d'interpréter les logs du noyau
- Redémarrer rsyslog pour prendre en compte les changements

```
echo -e '# Audit logs
$template COS_Audit, "/opt/rsyslog/%HOSTNAME%/auditd.log"
local6.* ?COS_Audit

# Authentication logs
$template COS_Authentication, "/opt/rsyslog/%HOSTNAME%/authentication.log"
auth,authpriv.* ?COS_Authentication

# Kernel events
$template COS_Kernel, "/opt/rsyslog/%HOSTNAME%/kernel.log"
kern.warn ?COS_Kernel

# Apache events
$template COS_apache, "/opt/rsyslog/%HOSTNAME%/%PROGRAMNAME%.log"
:programname, isequal, "httpd" ?COS_apache

# Firewalld events
$template COS_firewalld, "/opt/rsyslog/%HOSTNAME%/%PROGRAMNAME%.log"
:programname, isequal, "firewalld" ?COS_firewalld

# MariaDB events
$template COS_mariadb, "/opt/rsyslog/%HOSTNAME%/%PROGRAMNAME%.log"
:programname, isequal, "mariadb" ?COS_mariadb


# Delete other events
*.* ~' >> /etc/rsyslog.d/10-centos8.conf

systemctl restart rsyslog
```

6.1.1.2. Configuration client

Les clients doivent savoir où renvoyer les logs.

```
echo -e "\$ActionQueueFileName queue
\$ActionQueueMaxDiskSpace 1g
\$ActionQueueSaveOnShutdown on
\$ActionQueueType LinkedList
\$ActionResumeRetryCount -1
*.* @@vlslogacs01:514
*.* stop" > /etc/rsyslog.d/00-client.conf
systemctl restart rsyslog
```