2- Rester informé sur l'activité du serveur

Un serveur mail nous permettra d'obtenir un retour sur ce qui se passe sur notre serveur et de rester informer sur les points importants sans devoir continuellement regarder les fichiers logs.

C'est quoi les logs ? Ils servent à quoi ?

Les applications créent des fichiers "logs" pour garder une trace de l'activité qui c'est passé à un moment donné. Ces fichiers sont loin d'être de simples sorties de texte et peuvent être très complexes à parcourir. En cas de panne d'un service, il devient vital d'utiliser toute l'aide disponible. Les logs permettent la plupart du temps d'analyser ce qui s'est exactement passé et permettent de trouver plus facilement une solution. Logwatch est un analyseur de log très puissant qui rend la vie plus facile. Un bon fichier journal doit être aussi détaillé que possible afin d'aider l'administrateur qui a la responsabilité de maintenir le système, à trouver les informations exactes nécessaires à une panne ou pour s'assurer d'un bon fonctionnement. De ce fait les journaux, logs, sont généralement peu concis et ils contiennent des tonnes de répitions qui nécessitent des analyses et un filtrage approfondis. C'est là que Logwatch entre en jeu.

- Setup du service mail
- Monitorer ses Logs avec Logwatch

Setup du service mail

FQDN

Pour cette étape vous devez posséder un nom de domaine qui pointe sur votre serveur (DNS) et connaître votre FQDN (Fully Qualified Domain Name
Pour connaître son FQDN:

```
hostname -f
```

Pour éditer son `hostname` il suffit d'éditer:

```
nano /etc/hostname
```

Puis **remplacer le hostname actuel** par celui voulu par exemple: `server-ex52` deviendra `toto` puis on sauvegarde. Le changement sera pris en compte au prochain reboot sinon vous pouvez le réactualiser avec la commande `sudo hostname toto` puis vérifier si le changement a été pris en compte avec la commande `hostname`.

Ensuite mettons en place le FQDN pour cela faite:

```
nano /etc/hosts
```

Modifier votre fichier dans notre cas on veut rester en local *(pensez à changer domaine.tld par votre nom de domaine)*:

```
127.0.0.1 toto.domaine.tld toto
127.0.0.1 localhost
```

A la fin il devrait ressembler à cela:

```
127.0.0.1 toto.domaine.tld toto
127.0.0.1 localhost

# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
```

```
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

Une fois vos changements sauvegardés vérifier votre FQDN: Il suffit de taper `hostname -f`

SERVEUR MAIL POSTFIX

Maintenons installons postfix, quelques questions nous serons posées:

```
apt-get install -f postfix mailutils
```

On répond aux questions suivantes posées par l'installation :

General type of mail configuration ? Internet Site
System mail name ? domaine.tld

Puis on relance la configuration complète du paquet :

```
dpkg-reconfigure postfix
```

General type of mail configuration ? Internet Site
System mail name ? domaine.tld
Root and postmaster mail recipient ? Laisser vide
Other destinations to accept mail for ? domaine.tld, localhost.domaine.tld, localhost
Force synchronous updates on mail queue ? No
Local networks ? Laisser par défaut 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
Mailbox size limit (bytes) ? 0
Local address extension character ? Laisser par défaut (+)
Internet protocols to use ? ipv4 *all si vous avez ipv6 de config*

Pour tout problèmes vous pouvez voir votre configuration en allant voir le fichier de configuration: `/etc/postfix/main.cf` et les logs sont disponibles sous `/var/log/mail.log`

Testons notre configuration:

```
echo 'Hello World ! Je suis un email du serveur.' | mail -s 'Hello World'
mon_adresse_email@mail.com
```

Le mail peut arriver dans les spams c'est tout à fait normal ne vous inquiétez pas. Nous allons maintenant rediriger les mails de root vers notre adresse mail. Pour cela:

```
nano /etc/aliases
```

Puis rajouter à la suite:

```
root: mon_adresse_email@mail.com
```

Puis pour valider les changements:

```
newaliases
```

Testons ensuite de nouveau:

```
echo 'Hello World ! Test root mail.' | mail -s 'Hello World' root
```

Afin de ne pas avoir Postfix qui écoute inutilement sur le port 25:
Allons éditer:

```
nano /etc/postfix/master.cf
```

et commenter la ligne suivante:

```
#smtp      inet  n       -       y       -       -       smtpd
```

On relance ensuite postfix:

```
service postfix restart
```

On va mettre ensuite en place un email lorsqu'un utilisateur accède au compte root pour cela:

```
nano /root/.bashrc
```

Puis rajouter à la fin du fichier:

```
echo 'Acces Shell Root le: ' `date` 'par' `who` | mail -s 'Connexion serveur via root' root
```

Il est également possible de faire pour tout les utilisateurs et donc pour chaque connexion ssh ! Pour cela il suffit de modifier le fichier `/etc/bash.bashrc`

On ajoute ensuite la ligne suivante à la fin du fichier:

```
echo 'Acces Shell le: ' `date` 'par' `who` | mail -s 'SSH: acces sur serveur: '`hostname` root
```

Voilà nous avons fini de configurer notre serveur mail en cas de problèmes.

Monitorer ses Logs avec Logwatch

Installation de Logwatch

Les rapports créés par Logwatch sont classés selon les services (Applications) exécutés sur votre système. Les applications peuvent être définies dans le fichier de configuration de logwatch. De plus, Logwatch permet la création de scripts d'analyse personnalisés pour des besoins spécifiques

Installation du paquet:

```
apt install logwatch
```

S'il n'existe pas déjà créez le dossier `/var/cache/logwatch` nécessaire au bon fonctionnement de logwatch :

```
mkdir /var/cache/logwatch
```

Configuration de Logwatch

On va modifier le fichier de configuration se trouvant à l'adresse `/usr/share/logwatch/default.conf/logwatch.conf`, ouvrons le avec l'éditeur de notre choix.

Pour cela on va créer une copie du fichier de configuration par défaut

```
cp /usr/share/logwatch/default.conf/logwatch.conf /etc/logwatch/conf/logwatch.conf
```

Puis on l'édite:

```
nano /etc/logwatch/conf/logwatch.conf
```

On observera une longue liste de variables que l'application utilise lors de son éxécution.
Nous modifierons ou ajusterons les valeurs suivantes:

```
# Le type d'output ici on veut recevoir nos logs par mail
Output = mail
# Pour recevoir les mails au format html, c'est plus agréable à lire
```

```
Format = html
# Adresse sur laquelle vous allez recevoir les mails
MailTo = root # <= Correspondra a l'adresse précédemment définis pour root dans le setup du service mail
# Niveau de détail des logs Low | Med | High
Detail = Med

# Pour les services vous pouvez gardez la ligne Service = All mais si vous souhaitez
# recevoir seulement des rapports spécifiques vous pouvez lister chaque service comme ci dessous
# (en retirant le #). Nous garderons All dans notre cas

# Service = sendmail
# Service = http
# Service = identd
# Service = sshd2
# Service = sudo
```

Ajoutons un rapport de log pour Nginx (Si vous utilisez Nginx of course)

Pour commencer on va créer une configuration spécifique:

```
nano /etc/logwatch/conf/logfiles/nginx.conf
```

Puis nous allons copier dedans la configuration ci dessous:

```
########################################################
# Define log file group for nginx
########################################################

# What actual file? Defaults to LogPath if not absolute path….
LogFile = nginx/*access.log
LogFile = nginx/*access.log.1
LogFile = nginx/*error.log
LogFile = nginx/*error.log.1

# If the archives are searched, here is one or more line
# (optionally containing wildcards) that tell where they are…
#If you use a “-” in naming add that as well -mgt
Archive = nginx/*access.log*
Archive = nginx/*error.log*
```

```
# Expand the repeats (actually just removes them now)
*ExpandRepeats

# Keep only the lines in the proper date range…
*ApplyhttpDate

# vi: shiftwidth=3 tabstop=3
```

On crée ensuite le second fichier essentiel pour faire correspondre notre service *(en se basant sur la conf http existante)*:

```
cp /usr/share/logwatch/default.conf/services/http.conf /etc/logwatch/conf/services/nginx.conf
```

On l'ouvre ensuite avec notre éditeur préféré:

```
nano /etc/logwatch/conf/services/nginx.conf
```

Puis on modifie le début du fichier:

```
Title = “nginx”
LogFile = nginx
```

Et pour finir on copie le fichier du script:

```
cp /usr/share/logwatch/scripts/services/http /etc/logwatch/scripts/services/nginx
```

Puis on relance logwatch vous devriez recevoir un mail récapitulatif et par la suite un mail journalier:

```
logwatch restart
```

Et voilà logwatch vous permettra de garder une trace de ce qu'il se passe sur votre serveur quotidiennement !
Pour modifier un service reprenez l'exemple ci dessus et modifiez `http` et `nginx` par le service correspondant à votre besoin.