

Généralités

Monit est un outil libre de surveillance des services locaux installés sur une machine Unix. Il est distribué sous licence publique générale GNU.

Monit entre dans la famille des logiciels de surveillance, comme Munin et Nagios. Il peut vérifier la disponibilité d'un daemon et les ressources occupées qu'il consomme, et en fonction du résultat choisir de le laisser tranquille, de le redémarrer ou de le stopper. De plus, il fournit un système d'alerte mail et une interface Web soigné sur lequel on peut s'appuyer au quotidien.

Installation

Installer le rpm récupéré du Web ou se reporter à la documentation [Gestionnaire Paquets GNU/Linux](#) pour plus d'informations

L'installer avec la commande suivante.

```
rpm -ivh monit.....el5.rpm
```

Je n'ai pas eu de dépendances à installer avec une installation de base de RHEL4.6. Si il y en a les obtenir sur le même site et les installer avec la même commande avant d'installer le paquetage que l'on désire.

Configuration

Documentations

- Site officiels : <http://www.tildeslash.com/monit/doc/>
- Developpez.com : <http://gorgonite.developpez.com/tutoriels/linux/monit/#LII-2>

Configuration générale

Première génération (Red Hat ES3)

Tout se situe dans le fichier `/etc/monit.conf`. Configuration + paramètres des processus (block de texte avec condition pour chaque service : check...).

Voir la deuxième génération pour la configuration. C'est exactement la même excepté que les fichiers pour les services sont séparés.

Seconde génération (Red Hat ES4)

Emplacement	Fonction
/etc/monit.conf	Fichier de configuration de monit.
/etc/monit.d	Dossier qui contient un fichier par sous-daemon sur lequel il peut effectuer des actions. La configuration est individuelle à chaque daemon ou groupe de daemons proches.

Le principe de monit est qu'il surveille les processus lancés. Il vérifie la présence d'un fichier .pid et s'il venait à manquer en informe l'administrateur par mail. De plus, sur l'interface Web, on voit l'état des processus de manière simple mais précise. On a aussi possibilité d'agir via l'interface pour redémarrer des services.

Monit ajoute à cette vérification des conditions de charge CPU qui peuvent entrainer si on le souhaite un redémarrage ainsi que de nombre de redémarrages possibles maximums (message de timeout) avant de considérer le service comme définitivement HS. Dans la majorité des cas c'est à ce moment qu'on commence à travailler...

Configuration spécifique

/etc/monit.conf

Laisser les valeurs par défaut pour ces directives.

```
set daemon 120 #durée des cycles de vérification
set logfile syslog facility log_daemon #connexion syslog
```

Configuration des alertes mails.

```
set mailserver srv-mail.domaine.local port 25 with timeout 15 seconds
set alert mailsupervision@domaine.fr with reminder on 120 cycles
```

Configuration du port d'écoute pour l'interface Web. Cette interface est accessible via l'interface d'administration grâce au reverse proxy mis en place dans la configuration Apache. Pour plus de détails se référer à la documentation correspondante. Néanmoins elle est accessible en locale sur le port 2812 (<http://localhost:2812>).

```
set httpd port 2812 and
    use address localhost # only accept connection from localhost
    allow localhost # allow localhost to connect to the server
```

Dernière directive qui inclus les fichiers positionnées dans le dossier monit.d.

```
include /etc/monit.d/*
```

Pour la configuration globale de monit, il n'y a rien d'autre à configurer.

Il faut juste vérifier deux points.

- S'assurer que le serveur est autorisé à envoyer des mails (attention aux restrictions des serveurs SMTP sur exchange par exemple)

Attention dans une configuration où on met localhost au lieu d'une adresse valide d'un serveur de messagerie (MTA=Mail Transfert Agent), il est probable que les alertes finiront dans le fichier /var/mail/root (ou la boîte de l'utilisateur qui reçoit les mails de root).

Remarque. Monit envoie lui-même les mails. Si on désactive tout les MTA du serveur, les mails sont toujours envoyés correctement.

/etc/monit.d

Ce dossier contient plusieurs fichiers à créer qui contiennent les paramètres de monit sur un ou plusieurs services. Recopier les fichiers d'un autre serveur ou sauvegardé sur un serveur de fichier dans le dossier. Enlever les services qui ne seront pas présent sur le serveur car forcément il ne seront pas monitorés. Des exemples de configuration tout prêt pour les principaux services sont présent sur le site officiel à cette adresse : <http://www.tildeslash.com/monit/doc/examples.php>

Emplacement	Fonction
/etc/monit.d/apache	Gestion du serveur Web Apache.
/etc/monit.d/basesystem	Gestion des processus de bases (atd, acpid, xinetd, syslog...) et de la surveillance des volumes disques.
/etc/monit.d/clamav	Gestion de l'antivirus clamav.
/etc/monit.d/munin-node	Gestion du client Munin.
/etc/monit.d/tina	Gestion de l'agent de sauvegarde.
/etc/monit.d/webmin	Gestion de l'interface d'administration Web.
/etc/monit.d/xfss	Gestion du serveur X font server.
/etc/monit.d/Xvnc	Gestion du serveur vncserver.

On peut au choix regrouper des services dans un fichier ou en faire un pour chaque. La combinaison des deux est intéressant comme c'est le cas ici.

Vérifier que monit est au démarrage.

```
chkconfig --list | grep monit
```

Si non, mettre monit au démarrage du système.

```
chkconfig monit on
```

A chaque modification d'un fichier, il faut redémarrer monit pour que cela soit pris en compte.

```
service monit restart
```

Le seul service que monit ne peut pas surveiller est lui-même.

Cas particuliers

Dans le fichier `/etc/monit.d/basesystem`, il faut adapter la vérification des volumes disques à la configuration du serveur.

```
#####SURVEILLANCE FS NON CLUSTER#####  
  
check device Mes_Logs_Space with path /dev/mapper/Mon_Linux-Mes_Logs  
  if space usage > 80% then alert  
  mode passive  
  every 3 cycles  
  group local  
  
check device Mes_Programmes_Space with path /dev/mapper/Mon_Linux-  
Mes_Programmes  
  if space usage > 80% then alert  
  mode passive  
  every 3 cycles  
  group local  
  
check device Mon_Temporaire_Space with path /dev/mapper/Mon_Linux-  
Mon_Temporaire  
  if space usage > 80% then alert  
  mode passive  
  every 3 cycles  
  group local  
  
check device Boot_Space with path /dev/sda1  
  if space usage > 80% then alert  
  mode passive  
  every 3 cycles  
  group local  
  
check device Ma_Racine_Space with path /dev/mapper/Mon_Linux-Ma_Racine  
  if space usage > 80% then alert  
  mode passive  
  every 3 cycles  
  group local
```

Surveillance de l'acpid

Pour surveiller le daemon `acpid`, il faut modifier le script de lancement dans `/etc/init.d` pour lui dire de créer un fichier qui contient son identifiant de processus. Ce fichier sert à sa supervision dans `monit`.

Il faut modifier les deux lignes suivantes en gras du script `/etc/init.d/acpid`.

```
start() {
```

```

# Check if it is already running
if [ ! -f /var/lock/subsys/acpid ]; then
    echo -n "Starting acpi daemon: "
    daemon /usr/sbin/acpid
    RETVAL=$?
    [ $RETVAL -eq 0 ] && touch /var/lock/subsys/acpid
    echo
    /usr/bin/pgrep -x -P 1 acpid > /var/run/acpid.pid
fi
return $RETVAL
}

stop() {
    echo -n "Stopping acpi daemon: "
    killproc /usr/sbin/acpid
    RETVAL=$?
    [ $RETVAL -eq 0 ] && rm -f /var/lock/subsys/acpid
    echo
    rm -f /var/run/acpid.pid
    return $RETVAL
}

```

Redémarrer monit avec un service monit restart. Le daemon acpid doit être visible en running sur l'interface.

Surveillance d'xinetd

Normalement par défaut, la configuration d'xinetd va mal être vu dans monit. Pour que cela fonctionne, activer echo sur localhost :

Dans /etc/xinetd.d/echo, rajouter les lignes suivantes en gras et surtout passer la valeur de disable à no pour l'activer justement.

```

service echo
{
    type                = INTERNAL
    id                  = echo-stream
    socket_type         = stream
    protocol            = tcp
    user                = root
    wait                = no
    interface           = 127.0.0.1
    only_from           = 127.0.0.1
    disable              = no
}

```

Redémarrer xinetd.

Service xinetd restart

Redémarrer monit.

Service monit restart

Avec un `netstat -natp`, on doit voir `xinetd` sur le port 7 qui correspond à `echo` (on peut vérifier son port dans `/etc/services`).

Vérification de l'ordre de démarrage des processus

Commencer par vérifier que tous les services monitorés sont lancés au démarrage pour les runlevel 3 et 5 avec la commande `chkconfig`. Ceci évite que `monit` les lance au démarrage car il ne les voit pas activés. Si on voit des mails à ce sujet c'est certainement parce qu'on a oublié de faire la manipulation sur le service en question.

Dans mon cas, `webmin` était paramétré pour démarrer mais était lancé après `monit`. Du coup au démarrage `monit` démarrait directement `webmin` et envoyait un mail concernant l'arrêt de `webmin` qu'il démarrait aussitôt.

Objectif : Faire en sorte que monit soit le dernier script lancé par rapport aux processus qu'il supervise.

Si on met un service au démarrage du système, un ordre lui est attribué en fonction des entêtes en commentaires (et oui les commentaires sont lus !) qui sont présents dans les scripts d'initialisations situés dans `/etc/init.d/*`.

La commande `chkconfig` lit ces entêtes et ajoute un lien dans `/etc/rcX.d/*`.

Un `S` pour start (lancé au démarrage dans l'ordre de son numéro qui suit la lettre). Un `K` pour kill (pas lancé au démarrage).

Ex : `S99webmin K99lvm2-monitor`

Il faut éviter cela :

```
lrwxrwxrwx 1 root root 15 nov 29 14:27 S97rhnsd -> ../init.d/rhnsd
lrwxrwxrwx 1 root root 14 dÃ©c 5 10:52 S97tina -> ../init.d/tina
lrwxrwxrwx 1 root root 19 nov 28 16:40 S98haldaemon -> ../init.d/haldaemon
lrwxrwxrwx 1 root root 15 dÃ©c 3 09:18 S98monit -> ../init.d/monit
lrwxrwxrwx 1 root root 11 nov 28 16:41 S99local -> ../rc.local
lrwxrwxrwx 1 root root 26 nov 29 17:09 S99tina.tina ->
/etc/rc.d/init.d/tina.tina
lrwxrwxrwx 1 root root 16 nov 29 16:23 S99webmin -> ../init.d/webmin
```

Ici `Webmin` est après `monit`. Ils sont tous les deux en "S". `Webmin` est à 99 et `monit` à 96 donc c'est `monit` le premier lancé (ordre croissant au démarrage). Il voit `webmin` à l'arrêt et le démarre.

Pour changer cela, aller dans le fichier `/etc/init.d/webmin` et changer la valeur de la deuxième ligne. On passe de ca :

```
#!/bin/sh
# chkconfig: 235 99 10
```

```
# description: Start or stop the Webmin server
```

à ca :

```
#!/bin/sh
# chkconfig: 235 96 10
# description: Start or stop the Webmin server
```

Suppression des scripts de démarrage.

```
chkconfig --del webmin
```

Ajout des scripts de démarrage mais en position 96 : avant monit.

```
chkconfig --add webmin
```

```
lrwxrwxrwx 1 root root 16 dÃ©c 6 13:59 S96webmin -> ../init.d/webmin
lrwxrwxrwx 1 root root 20 nov 28 16:40 S97messagebus ->
../init.d/messagebus
lrwxrwxrwx 1 root root 15 nov 29 14:27 S97rhnsd -> ../init.d/rhnsd
lrwxrwxrwx 1 root root 14 dÃ©c 5 10:52 S97tina -> ../init.d/tina
lrwxrwxrwx 1 root root 19 nov 28 16:40 S98haldaemon -> ../init.d/haldaemon
lrwxrwxrwx 1 root root 15 dÃ©c 3 09:18 S98monit -> ../init.d/monit
```

Si on redémarre un serveur et qu'on a des mails de monit qui ne trouve pas un service, c'est qu'il n'est pas lancé avant monit mais après. Il faut donc vérifier dans les dossiers /etc/rc5.d/ et /etc/rc3.d/ que monit est bien après les autres.

Conclusion

Vérifier sur l'interface Web que les services sont visibles comme "running". Si ce n'est pas le cas, modifier les fichiers de configurations pour que cela soit bon.

Au final, voici la liste des processus qui sont monitorés et visibles sur l'interface Web de monit.

```
Xfs
Webmin
Tina
munin-node
clamd
syslogd
snmpd
xinetd
sshd
crond
gpm
atd
acpid
sendmail
```

```
irqbalance
ntpd
httpd
Xvnc
```

+ Espace des partitions et présence de certains fichiers notamment pour clamav (pattern_file) et syslog (syslog_file).

Exemples monit

Exemple d'un monit sur le service apache dans /etc/monit.d/apache.

```
##### Apache #####
check process httpd with pidfile /var/run/httpd.pid
  if cpu is greater than 90% for 5 cycles then alert
  if totalmemory is greater than 90% for 5 cycles then alert
  if failed host localhost port 80 protocol apache-status dnslimit > 25% or
loglimit > 80% or waitlimit < 20% then alert
  #if failed host localhost port 80 then alert
  #if failed host nomappli.domaine.fr port 80 protocol HTTP request
/monit/token then restart
  mode passive
  group applimetier
```

Exemple d'un monit sur le service tomcat dans /etc/monit.d/tomcat.

```
##### Tomcat #####
check host nomappli.domaine.fr with address 172.25.30.130
  every 3 cycles
  #test de la presence du mot Bonjour dans la page pour verifier qu elle s
affiche bien
  if failed url https://nomuser:motdepasseuser@nomappli.domaine.fr/ and
content == "Bonjour" with timeout 30 seconds then alert
  mode passive
  group applimetier
```

Exemple d'un monit sur le service postgres dans /etc/monit.d/postgres.

```
##### PostgreSQL #####
check process postgres with pidfile /var/run/postmaster.pid
  if cpu is greater than 90% for 5 cycles then alert
  if totalmemory is greater than 90% for 5 cycles then alert
  mode passive
  group applimetier
```


From:
<https://wiki.ouiehoutca.eu/> - **kilsufi de noter**

Permanent link:
<https://wiki.ouiehoutca.eu/monit>

Last update: **2021/01/21 21:42**

