



Monit - надежный мониторинг системы

Опубликовано muff в Ср, 2011-10-05 16:55



Понадобилось решение для мониторинга и автоматического перезапуска некоторых сервисов. Поискав решение в Интернете, решил ознакомиться с возможностями **monit**.

Немного общей информации о **monit**.

Monit умеет мониторить:

- процессы (наличие, количество потребляемых ресурсов)
- файлы, директории и файловые системы на изменения (дата создания/изменения, изменения размера и контрольной суммы)
- сетевые хосты (пинг и коннект на определенный порт по определенному протоколу)

При возникновении проблемы **monit** отправляет e-mail уведомление (шаблоны можно модифицировать) и может перезапустить сервис. В **monit** встроен простенький веб-сервер, который позволяет посмотреть состояние объектов мониторинга, включить/выключить определенный объект. **Monit** умеет перезапускать сервисы если они падают или не выполняется какое-то условие.

Monit построен с идеей того что система мониторинга должна быть максимально надежной и простой. И это действительно выполняется - на **monit** можно положиться.

Конечно из-за своей простоты **monit** не обладает тем количеством возможностей, которыми обладают Enterprise-системы мониторинга. Однако существует дополнение к **monit** под названием **M/Monit**, которое позволяет управлять несколькими серверами с **monit** из одного места. К сожалению, **M/Monit** распространяется под коммерческой лицензией, то есть за деньги.

Выполним установку из портов:

```
# cd /usr/ports/sysutils/monit && make install clean && rehash
```

Создадим конфигурационный файл **monitrc** копированием предложенного примера:

```
# cd /usr/local/etc && cp monitrc.sample monitrc
```

Возможно кому-то пригодится перевод на русский дефолтного файла конфигурации **monitrc** (для **monit-5.3_1**):

```
##### Мо
```



```
nit control file#####
##### ?????????? ?????????? ? '#' ? ?????????????? ?? ????? ??????.## ???
???? ?????? ???? ?????????? ?????????? ? ?????????????? ? '/' .#### ????? ????????? ??????????
???? ?????? ?????????????????? ?????????????????? . ??????????????## ?????????????????? ?????? ??????????????????
? ?????????? ?????????? ?????????????? ? ?????? ??????## ?????????????????? ? ?????????????????? Monit.#####
##### Global section#####
#### ??????? Monit ? ?????????? ?????????? (????????????? ??? daemon):#set daemon 60
    # проверять сервисы с минутным интервалом#with start delay 240 # опциона
льно: задержка перед первой проверкой # задана в 4 минуты (по умолчани
ю Monit # выполняет проверку сразу после запуска### Использовать логи
рование syslog как 'daemon' facility. Если опция## FACILITY будет упущена, Monit будет испол
зовать 'user' facility по## умолчанию. Если есть необходимость логировать в определенный
лог-файл,## необходимо задать полный путь к лог-файлу.## set logfile syslog facility log_dae
mon### Задать расположение файла, где будет храниться уникальный идентификатор## е
кземпляра Monit. Идентификатор генерируется и сохраняется при первом## запуске Monit. П
о умолчанию файл размещен в $HOME/.monit.id.## set idfile /var/.monit.id### Задать распо
ложение файла состояния, куда сохраняются состояния в каждом## цикле. По умолчанию фай
л размещен в $HOME/.monit.state. Если файл состояния## расположен в постоянной файлово
й системе, то Monit восстановит состояние## мониторинга после перезапуска. Если это врем
енная файловая система, то## состояние будет утеряно после перезапуска, что может быть
удобно в## некоторых ситуациях.## set statefile /var/.monit.state### Задать список почтовы
х серверов для доставки уведомлений. Можно указывать## несколько серверов, разделяя сп
исок комами. Если первый почтовый сервер## выдает ошибку, Monit будет использовать вто
рой сервер в списке и т.д.## По умолчанию Monit использует порт 25 - возможно переименов
ать, используя## опцию PORT.## set mailserver mail.bar.baz, # primary mailserver#
    backup.bar.baz port 10025, # backup mailserver on port 10025# localhost
    # fallback relay### По умолчанию, Monit уничтожает уведомления, если почтовые сервер
а## недоступны. Если вы хотите сохранить уведомления для дальнейших попыток## достав
ки, необходимо использовать EVENTQUEUE состояние. Базовый каталог,## где будут хранить
ся недоставленные уведомления, задается опцией BASEDIR.## Можно ограничить максималь
ный размер очереди, используя опцию SLOTS (если## не указано, очередь ограничивается п
ространством, доступным на конечной## файловой системе.## set eventqueue# basedir /v
ar/monit # set the base directory where events will be stored# slots 100 # optionally limit
the queue size### Отправлять состояние и события в M/Monit (чтобы получить больше инфор
мации## о M/Monit, смотрите http://mmonit.com/
). По умолчанию Monit регистрирует## вверительные грамоты с M/Monit, так что M/Monit мо
жет обратно связываться## с Monit и нету необходимости регистрировать вверительные гра
моты вручную## в M/Monit. Есть возможность отключить регистрацию вверительных грамот,
## используя закомментированную опцию ниже. Если вы заботитесь о## безопасности, реко
мендуется использовать https для обмена с M/Monit и## отправки вверительных грамот.## s
et mmonit http://monit:monit@192.168.1.10:8080/collector
# # and register without credentials # Don't register credentials### Monit по умолчанию ис
пользует следующий формат почтовых уведомлений:#### --8<--## From: monit@$HOST
    # Отправитель## Subject: monit alert -- $EVENT $SERVICE # Тема#### $EVEN
T Service $SERVICE ### ### Date: $DATE
### Action: $ACTION ### Host: $HOST # Тело письма##
Description: $DESCRIPTION ### ### Your faithful employee,
### Monit ### --8<--#### Можно переписать формат сообщен
ия или его части, такие как тема или## отправитель, используя MAIL-FORMAT состояние. Мак
росы, такие как DATE## и т.д., применяются во время выполнения. Например, чтобы перепис
ать## отправителя, используется:## set mail-format { from: monit [at] foo [dot] bar
}### Можно задать получателей уведомлений, которые получают уведомления,## когда сер
висы, заданные в этом файле имеют ошибки. Уведомления можно## ограничивать событиям
и, используя фильтры, как указано на примере ниже.## set alert sysadm [at] foo [dot] bar
    # receive all alerts# set alert manager [at] foo [dot] bar
only on { timeout } # receive just service-# # timeout alert#### M
onit имеет встроенный web-сервер, который может использоваться для## просмотра статуса
контролируемых сервисов и управления сервисами через## web-интерфейс. Смотри Monit Wik
```



```
i, если необходима поддержка SSL для## web-сервера.#set httpd port 2812 and
use address localhost # принимать соединения только с localhost allow localhost
# разрешить соединения только с localhost и allow admin:monit
# разрешить пользователя 'admin' с паролем 'monit' allow @monit
# разрешить пользователям группы 'monit' соединяться (rw) allow @users readonly
# разрешить пользователям группы 'users' соединяться (ro)

##### Se
rvice#####
#### ?????????? ?????????? ?????????? ?????????, ?????? ??? load average, ??????????????##
cpu ? memory. ?????? ?????????? ?????????? ?????????, ?????????????? ? ??????????, ??????????## ??????????
?????, ?????? ?????????????? ?????????????????? ??????????## check system myhost.mydomain.tld#
if loadavg (1min) > 4 then alert# if loadavg (5min) > 2 then alert# if memory
usage > 75% then alert# if swap usage > 25% then alert# if cpu usage (user) >
70% then alert# if cpu usage (system) > 30% then alert# if cpu usage (wait) >
20% then alert### ?????????? ?????????????????? ??????, ?????????????? ??????, ?????, uid ? gid.
?????## ?????, ?????? ?????????????????? ?????????????? ? "global section", ??????????????## ??????
????????? ?????? ?????? ?????????????? ?????????????????? ??????????????????, ??????????## ?????????? ??????
????????? ??????????????. ?????????? ?????? ?????? ??????????????????,## ?????????? ?????? GROUP. ??????
? ?????? ?????? ?????? ??????, ?????????? ??????????????## ?????????? 'group name'## check f
ile apache_bin with path /usr/local/apache/bin/httpd# if failed checksum and#
expect the sum 8f7f419955cefa0b33a2ba316cba3659 then unmonitor# if failed perm
ission 755 then unmonitor# if failed uid root then unmonitor# if failed gid ro
ot then unmonitor# alert security [at] foo [dot] bar
on {# checksum, permission, uid, gid, unmonitor# } with the mail-format { subject: Alar
m! }# group server### Проверка того, что процесс запущен, в этом случае - Apache, и что о
н## отвечает на HTTP и HTTPS запросы. Проверка на использование ресурсов,## таких как и
спользование сру и memory, а также количества дочерних процессов.## Если процесс не зап
ущен, Monit по умолчанию перезапустит его. В случае,## если сервис перезапускается очень
часто и проблема повторяется, есть## возможность отключить мониторинг, используя состо
яние TIMEOUT. Этот## сервис зависит от другого сервиса (apache_bin), который определяетс
я выше.## check process apache with pidfile /usr/local/apache/logs/httpd.pid# start program =
"/etc/init.d/httpd start" with timeout 60 seconds# stop program = "/etc/init.d/httpd stop"# if cpu
> 60% for 2 cycles then alert# if cpu > 80% for 5 cycles then restart# if totalmem > 200.0 MB f
or 5 cycles then restart# if children > 250 then restart# if loadavg(5min) greater than 10 for 8 c
ycles then stop# if failed host www.tildeslash.com
port 80 protocol http# and request "/somefile.html"# then restart# if failed port 443 type
tcpssl protocol http# with timeout 15 seconds# then restart# if 3 restarts within 5 cycles t
hen timeout# depends on apache_bin# group server### Проверка прав доступа файловой с
истемы, uid, gid, свободного пространства и## использования inode. Другие сервисы, такие к
ак базы данных, могут зависеть## от этих ресурсов и автоматический "мягкий" останов мож
ет быть каскадным,## перед заполнением файловой системы и потери данных.## check file
system datafs with path /dev/sdb1# start program = "/bin/mount /data"# stop program = "/bin
/umount /data"# if failed permission 660 then unmonitor# if failed uid root then unmonitor# if
failed gid disk then unmonitor# if space usage > 80% for 5 times within 15 cycles then alert# if
space usage > 99% then stop# if inode usage > 30000 then alert# if inode usage > 99% then
stop# group server#### Проверка временной метки файла. Например, мы проверяем, что е
сли файл## старше 15 минут, предполагаем что он ошибочен, если не обновлен. Также,## е
сли размер файла превышает указанный лимит, выполняется скрипт.## check file database w
ith path /data/mydatabase.db# if failed permission 700 then alert# if failed uid data then alert#
if failed gid data then alert# if timestamp > 15 minutes then alert# if size > 100 MB then exe
c "/my/cleanup/script" as uid dba and gid dba### Проверка прав на каталог, uid and gid. Событи
е выполняется, если каталог не## принадлежит пользователю с uid 0 и gid 0. Кроме того, пр
ава доступа должны## совпадать с восьмеричному описанию 755 (см. chmod(1)).## check di
rectory bin with path /bin# if failed permission 755 then unmonitor# if failed uid 0 then unmonit
or# if failed gid 0 then unmonitor### Проверка доступности удаленного хоста, используя дл
я теста ping и проверяя## содержимое ответа web-сервера. Отправляется до 3 ping и подкл
ючение к порту## выполняется на уровне приложений сетевой проверки.## check host mys
erver with address 192.168.1.1# if failed icmp type echo count 3 with timeout 3 seconds then aler
```



```
t# if failed port 3306 protocol mysql with timeout 15 seconds then alert# if failed url
http://user:password [at] www [dot] foo [dot] bar:8080/?querystring
# and content == 'action="_j_security_check"'# then alert#####
#####
## Includes#####
##### Есть возможность подключать дополнительные
части конфигураций с других файлов## или каталогов.## include /etc/monit.d/*#
```

Следующий шаг - создание собственного **monitrc**, редактируя пример. В моем случае **monitrc** получился следующий:

```
#####
## Global section
#####

set daemon 120
with start delay 240
set logfile /var/log/monit.log
set idfile /var/monit/.monit.id
set statefile /var/monit/.monit.state
set mailserver localhost
set eventqueue
    basedir /var/monit/alerts
    slots 100
set mail-format {
    from: monit [at] muff [dot] kiev [dot] ua
    subject: $SERVICE $EVENT at $DATE}
set alert admins [at] muff [dot] kiev [dot] ua
set httpd port 2812 and
    use address 0.0.0.0
    allow 192.168.168.22/32
    allow admin:monit

#####
## Includes
#####

include /usr/local/etc/monit/*.conf
```

Создадим каталоги, на которые имеются ссылки в файле конфигурации:

```
# mkdir /var/monit
# mkdir /var/monit/alerts
# mkdir /usr/local/etc/monit
```

Также рекомендую настроить ротацию логов (чтобы не исчерпать в результате все доступное дисковое пространство на **/var**). Более детально о настройке ротации логов можно почитать в [этой статье](#) [1]. Ротацию логов будем выполнять один раз в неделю:

```
# echo 'var/log/monit.log 644 3 * $W6D0 JC' >> /etc/newsyslog.conf
```

Теперь создадим несколько конфигурационных файлов для мониторинга и разместим их в каталог **/usr/local/etc/monit**. Будем выполнять проверку общих системных ресурсов, работу некоторых сервисов, свободного дискового пространства. В результате имеем такие конфигурационные файлы:



1. Проверка общих системных ресурсов - `/usr/local/etc/monit/system.conf`

```
check system muff.kiev.ua
  if loadavg (1min) > 4 then alert
  if loadavg (5min) > 2 then alert
  if memory usage > 75% then alert
  if swap usage > 25% then alert
  if cpu usage (user) > 70% then alert
  if cpu usage (system) > 30% then alert
  if cpu usage (wait) > 20% then alert
```

2. Проверка сервисов - `/usr/local/etc/monit/process.conf`

```
check process apache with pidfile /var/run/httpd.pid
  start program = "/usr/local/etc/rc.d/apache22 start"
  stop program = "/usr/local/etc/rc.d/apache22 stop"
  if cpu > 60% for 2 cycles then alert
  if cpu > 85% for 5 cycles then restart
  if totalmem > 600.0 MB for 5 cycles then restart
  if children > 100 then restart
  if failed host muff.kiev.ua port 80 protocol http
    and request "/index.php"
    then restart
  if 3 restarts within 5 cycles then timeout
check process dovecot with pidfile /var/run/dovecot/master.pid
  start program = "/usr/local/etc/rc.d/dovecot start"
  stop program = "/usr/local/etc/rc.d/dovecot stop"
  if cpu > 60% for 2 cycles then alert
  if cpu > 80% for 5 cycles then restart
  if totalmem > 100.0 MB for 5 cycles then restart
  if children > 60 then restart
  if 3 restarts within 5 cycles then timeout
  if failed port 110 type TCP protocol POP then restart
check process exim with pidfile /var/run/exim.pid
  start program = "/usr/local/etc/rc.d/exim start"
  stop program = "/usr/local/etc/rc.d/exim stop"
  if cpu > 60% for 2 cycles then alert
  if cpu > 80% for 5 cycles then restart
  if totalmem > 200.0 MB for 5 cycles then restart
  if children > 60 then restart
  if 3 restarts within 5 cycles then timeout
  if failed port 25 type TCP protocol SMTP then restart
check process mysql with pidfile /var/db/mysql/web0.muff.kiev.ua.pid
  start program = "/usr/local/etc/rc.d/mysql-server start"
  stop program = "/usr/local/etc/rc.d/mysql-server stop"
  if failed unixsocket /tmp/mysql.sock then restart
  if failed host 127.0.0.1 port 3306 protocol mysql then restart
  if 3 restarts within 5 cycles then timeout
check process proftpd with pidfile /var/run/proftpd.pid
  start program = "/usr/local/etc/rc.d/proftpd start"
  stop program = "/usr/local/etc/rc.d/proftpd stop"
  if cpu > 60% for 2 cycles then alert
  if cpu > 85% for 5 cycles then restart
  if failed host 127.0.0.1 port 21 protocol ftp then restart
  if 3 restarts within 5 cycles then timeout
```



3. Проверка свободного дискового пространства - **/usr/local/etc/monit/filesystem.conf**

```

check device root with path /dev/mirror/gm0s1a
  if space usage > 85% for 5 times within 15 cycles then alert
  if inode usage > 85% then alert
check device home with path /dev/mirror/gm0s1d
  if space usage > 85% for 5 times within 15 cycles then alert
  if inode usage > 85% then alert
check device usr with path /dev/mirror/gm0s1g
  if space usage > 85% for 5 times within 15 cycles then alert
  if inode usage > 85% then alert
check device var with path /dev/mirror/gm0s1e
  if space usage > 85% for 5 times within 15 cycles then alert
  if inode usage > 85% then alert

```

Добавим поддержку **monit** в **rc.conf** и запустим сервис:

```

# echo '# Monit - monitoring system' >> /etc/rc.conf
# echo 'monit_enable="YES"' >> /etc/rc.conf
# sh /usr/local/etc/rc.d/monit start
Starting monit.
monit: generated unique Monit id 10473348799bae9558af6f918ef160fd and stored to
'/var/monit/.monit.id'
Starting monit daemon with http interface at [0.0.0.0:2812]
Monit start delay set -- pause for 240s

```

Проверим, какие возможности предоставляет web-интерфейс. Скриншоты - ниже.



[2]



[3]



[4]



[5]



[6]

Выполним проверку работы **monit** - остановим сервис proftpd:

```
# killall proftpd
```

Через небольшой промежуток времени на почту должны прийти уведомления - о перезапуске сервиса, и о том, что сервис успешно запущен.

В логе `/var/log/monit.log` также можно обнаружить запись о сбое в работе сервиса:

```
[EEST Oct 5 16:47:33] error   : 'proftpd' process is not running
[EEST Oct 5 16:47:34] info    : 'proftpd' trying to restart
[EEST Oct 5 16:47:34] info    : 'proftpd' start: /usr/local/etc/rc.d/proftpd
[EEST Oct 5 16:49:34] info    : 'proftpd' process is running with pid 78440
```

Сервис успешно перезапущен.

Однако стоит иметь ввиду, что возможностей у **monit** намного больше, нежели описано в этой статье. Рекомендую ознакомиться с [документацией monit](#) [7].

Источник (получено 2026-04-22 21:35):

<http://muff.kiev.ua/content/monit-nadezhnyi-monitoring-sistemy>

Ссылки:

[1] <http://muff.kiev.ua/content/newsyslog-rotatsiya-logov>

[2] <http://muff.kiev.ua/files/imagepicker/1/monit-00.png>

[3] <http://muff.kiev.ua/files/imagepicker/1/monit-03.png>

[4] <http://muff.kiev.ua/files/imagepicker/1/monit-01.png>

[5] <http://muff.kiev.ua/files/imagepicker/1/monit-04.png>

[6] <http://muff.kiev.ua/files/imagepicker/1/monit-05.png>

[7] <http://mmonit.com/monit/documentation/monit.html>