Скрипт удаления файлов старше N дней

Опубликовано muff в Пт, 2015-03-06 19:08

Для бекапирования данных можно использовать как специализированное ПО, так и самописные скрипты.

В случае использования самописных скриптов, возникает необходимость периодически удалять архивы бекапов. Однако и этот процесс можно автоматизировать.

Попытаемся максимально приблизить условия к боевым.

Итак, имеется в наличии сервер, на котором в родительском каталоге (например, **/backup**) созданы подкаталоги, для хранения бекапов в зависимости от типа содержимого:

- конфигурации маршрутизаторов
- конфигурации коммутаторов
- конфигурации серверов
- данные с серверов
- т.д.

Логика написания скрипта состоит в следующем. В зависимости от типа содержимого, скорости устаревания данных, доступного дискового пространства и т.д, выбираем оптимальное время хранения бекапов в каталогах. После этого необходимо "пройтись" по каталогах и удалить все файлы, которые созданы более чем N дней назад.

Для поиска файлов, которые созданы более чем N дней назад, воспользуемся возможностями утилиты find [1].

Допустим, необходимо найти в каталоге все файлы, созданные более 25 дней назад:

```
# find /backup/jun -type f -mtime +25
/backup/jun/juniper.conf.gz_20150209_141842
/backup/jun/juniper.conf.gz_20150209_142441
/backup/jun/juniper.conf.gz_20150209_142733
/backup/jun/juniper.conf.gz_20150209_151936
/backup/jun/juniper.conf.gz_20150209_152555
/backup/jun/juniper.conf.gz_20150209_173457
```

После поиска устаревших файлов, необходимо их удалить. Для этого можно использовать, например, такую конструкцию:

```
find /backup/jun -type f -mtime +25 -exec rm -f {} \;
```

Либо же такую:

```
find /backup/jun -type f -mtime +25 -print0 | xargs -0 rm -f
```

Воспользуемся вторым вариантом и добавим несколько переменных. В результате получаем примерно следующий скрипт:

```
#!/bin/sh# ???? ? ??????? ??????FIND="/usr/bin/find"# Путь к утилите удаленияRM="
/bin/rm"# Путь к утилите xargsXARGS="/usr/bin/xargs"
# Список каталогов с архивами бекаповPATH1="/backup/jun"PATH2="/backup/sw"
# Время хранения файлов (в днях)TIME1="25"TIME2="60"
# Перебираем каталоги и удаляем устаревшие бекапы
$FIND $PATH1 -type f -mtime +$TIME1 -print0 | $XARGS -0 $RM -f$FIND $PATH2 -type f -
mtime +$TIME2 -print0 | $XARGS -0 $RM -f
```

Назовем его, например, **remove_old_backups.sh** и запустим на выполнение:

```
# sh /usr/local/etc/remove_old_backups.sh
```

После этого проверим наличие файлов старше 25 дней в каталоге **/backup/jun**:

```
# find /backup/jun -type f -mtime +25
```

Таких файлов не обнаружено. А вот, для примера, файлы старше 20 дней - присутствуют:

```
# find /backup/jun -type f -mtime +20
/backup/jun/juniper.conf.gz_20150210_124853
/backup/jun/juniper.conf.gz_20150212_145222
/backup/jun/juniper.conf.gz_20150212_151451
/backup/jun/juniper.conf.gz_20150212_151608
/backup/jun/juniper.conf.gz_20150213_033623
/backup/jun/juniper.conf.gz_20150213_033643
/backup/jun/juniper.conf.gz_20150213_033751
/backup/jun/juniper.conf.gz_20150213_113431
```

Скрипт работает... Модифицируем под свои нужды и пользуемся.

По завершению редактирования, в целях безопасности, выставим необходимые права и разшешения (у меня файлы архивов пишутся от имени пользователя **proftpd**):

```
# chown proftpd:proftpd /usr/local/etc/remove_old_backups.sh
# chmod 700 /usr/local/etc/remove_old_backups.sh
```

Ну и не забываем добавить ежедневный запуск скрипта в планировщик заданий:

```
# echo '# Clear old backups' >> /etc/crontab
# echo '2       6       *       *       *       proftpd     /bin/sh
/usr/local/etc/remove_old_backups.sh' >> /etc/crontab
```

Источник (получено *2026-11-24 07:12*):
http://muff.kiev.ua/content/skript-udaleniya-failov-starshe-n-dnei

Ссылки:
[1] http://muff.kiev.ua/content/poisk-v-freebsd