Exim – некоторые полезные команды

Командная строка Exim’а использует стандартный для Unix-систем способ задания опций. Каждая начинается с дефиса, после каждой может следовать несколько аргументов.

exim -bP

Вывод всех значений, установленных в конфигурационном файле.

exim -bp

Вывод почтовой очереди. Вывод списка сортируется в хронологическом порядке по прибытию сообщений. Если опция сопровождается списком идентификаторов сообщений, то показываются только эти сообщения. Пример вывода:

```
 10h 729K 1RdheP-00021S-GM <info@domain.com.ua>
 D user@rambler.ru
 user2000@gmail.com
 D user@safaritour.com.ua
 D user@fortunat.zssm.zp.ua
```

Первая строка содержит 4 колонки: *сколько сообщение находится в очереди, размер сообщения, ID сообщения, отправитель, как он указан в “конверте” (для bounce “<>”)*. Если сообщение заморожено (приостановлена попытка его доставки), в конце этой строки показывается текст “*** **frozen** ***“. Адреса по которым сообщение уже доставлено отмечены символом “**D**” (*Delivered – доставлено*). Если оригинальный адрес раскрывается в несколько адресов через файл алиасов или форвардов, оригинальный показывается с “D” только когда завершены доставки для всех дочерних адресов.

exim -bpr

Вывод почтовой очереди, но без сортировки. Полезно, когда в очереди много сообщений, а сортировка не нужна.

exim -bpc

Вывод количества сообщений в очереди.

exim -Mvl <message id>

Посмотр лог сообщения.

exim -Mvb <message id>

Вывод тела сообщения.

exim -Mvh <message id>

Вывод заголовков сообщения

exim -Mrm <message id> <message id> ...

Удалить сообщение из очереди и не посылать никаких ошибок (в логах запись о удалении будет).

exim -Mg <message id> <message id> ...

Удалить сообщение из очереди и отослать “отлуп” (cancelled by administrator).

exim -M <message id> <message id> ...

Ускорить доставку сообщения (немедленно выполняет доставку сообщения).

exim -Mar <message id> <address> <address> ...

Добавить адрес в список получателей сообщения.

exim -Mes <message id> <address>

Заменить адрес отправителя в сообщении на указанный.

exim -Mf <message id> <message id> …

Отметить перечисленные сообщения как “frozen”. Попытки доставки сообщения прекращаются, пока сообщение не будет разморожено вручную, или пока не пройдет время указанное в “auto_thaw”.

```
exim -Mt <message id> <message id> ...
```

“Разморозить” сообщения.

Для сообщений, которые подозреваются **в спаме** удобно сначала массово, сделать -Mf, потом -Mvl, -Mvh и -Mvb для пары выбранных наугад сообщений. Если все еще невозможно с консоли определить спам ли это (например, проблемы с кодировками или есть вложения), то можно с помощью -Mar добавить себя в список получателей, и ускорить данное сообщение с помощью -M. Когда все наконец станет ясно, то либо -Mrm, либо -Mt.

```
exipick -i | xargs exim -Mrm
```

Очистить очередь. То есть, по сути удаляем все сообщения из очереди.

```
exipick -zi | xargs exim -Mrm
```

Удалить из очереди все “замороженные” сообщения (помеченные как “frozen”).

Поиск в журнальных файлах с помощью exigrep

Утилита *exigrep* (не путайте с *exiqgrep*, использующейся для поиска в очереди), используется для поиска по лог файлам. Например *exigrep* может вывести все записи из лог файла с совпадающим *ID* сообщения, что довольно удобно, учитывая что каждое сообщение занимает 3 строки в лог файле.

Поиск сообщений отправленных с определенного IP адреса:

```
root@localhost# exigrep '<= .* \[12.34.56.78\] ' /path/to/exim_log
```

Поиск сообщений отправленных на определенный IP адрес:

```
root@localhost# exigrep '=> .* \[12.34.56.78\]' /path/to/exim_log
```

Данный пример ищет сообщения содержащие символы “=>”, и отправленные на адрес *“user@domain.tld”*, далее по конвейеру,

результат передается команде *grep*, которая из полученного результата выбирает строки, содержащие “<=” с информацией об отправителе, почтовом адресе, IP адресе, размере сообщения, ID сообщения и заголовок *subject*, если логгирование этой строки включено.

```
root@localhost# exigrep '=> .*user@domain.tld' /path/to/exim_log | fgrep '<='
```

Генерировать из лог файла и показать статистику *Exim*:

```
root@localhost# eximstats /path/to/exim_mainlog
```

То-же что и выше но с более подробными данными:

```
root@localhost# eximstats -ne -nr -nt /path/to/exim_mainlog
```

Аналогично но за определенный день:

```
root@localhost# fgrep YYYY-MM-DD /path/to/exim_mainlog | eximstats
```

В качестве дополнения

Удалить все сообщения в очереди, содержащие в теле, определенную строку:

```
root@localhost# grep -lr 'a certain string' /var/spool/exim/input/ | \
               sed -e 's/^.*\/\([a-zA-Z0-9-]*\)-[DH]$/\1/g' | xargs exim -Mrm
```

Командой выше, мы проверяем содержимое каталога */var/spool/exim/input/*, в поисках файлов очереди, содержащих определенную строку в теле сообщения, поскольку команда *exiqgrep* не умеет просматривать тело сообщений. Если вы решите удалить найденные файлы напрямую, ЭТО БУДЕТ НЕ ПРАВИЛЬНО, используйте предназначенные для этого команды *exim*.
Если вывод используемой команды слишком длинный, например ID сообщений при *exiqgrep -i*, которые нужно передать дальше по конвейеру команде*exim*, может быть превышено количество аргументов командной строки вашей системной оболочки. В этом

случае передавайте результат поиска через конвейер, команде *xargs*, которая будет обрабатывать результаты ограниченными порциями.
Например удалим тысячи сообщений, отправленных с адреса *joe@example.com*:

```
root@localhost# exiqgrep -i -f '' | xargs exim -Mrm
```

После того как вы внесли изменения в файл конфигурации, необходимо перезапустить exim, или послать рабочему процессу сигнал SIGHUP, что-бы он перечитал конфигурационный файл и изменения вступили в силу. Предпочтительней естественно отправить сигнал, нежели перезапускать приложение.

```
root@localhost# kill -HUP cat /var/spool/exim/exim-daemon.pid
```