

# Sieve для Dovecot 2

## Sieve

Зачем это нужно? – появляется возможность обработки почтовых сообщений на сервере по заданным правилам (настраивали правила сортировки в почтовой программе?, а потом приходили, к примеру домой и тоже самое настраивали?)

```
cd /usr/ports/mail/dovecot2-pigeonhole/  
make install
```

поставили, то что поставили состоит из 2х частей: плагин для dovecot который обрабатывает почту и сервер, позволяющий пользователям самим править свои правила sieve.

Приступим.

Включим плагин в конфигурации dovecot, обработки запускаются при локальной доставке.

```
cd /usr/local/etc/dovecot/conf.d/
```

скопируем новый конфиг файл

```
cp /usr/local/share/doc/dovecot/example-config/conf.d/90-  
sieve.conf /usr/local/etc/dovecot/conf.d/
```

включим плагин **sieve** в дополнение к, у меня уже работающему, плагину **quota 15-lda.conf**

```
mail_plugins = $mail_plugins quota sieve
```

Если почта у Вас приходит в dovecot по LMTP (см ниже настройку DSPAM) то в **20-lmtp.conf** тоже **90-sieve.conf**.

```
sieve = ~/.dovecot.sieve  
sieve_dir = ~/sieve  
}
```

**sieve = ~/.dovecot.sieve** – путь к пользовательскому основному активному Sieve сценарию. Когда используется ManageSieve, это

символическая ссылка на активный сценарий в каталоге `sieve_dir`. Избегайте активных Sieve сценариев в `sieve_dir` каталоге. В данном примере сценарий будет располагаться в домашнем каталоге пользователя. Согласно руководству, в пути можно использовать переменные `%u`, `%d`, `..` и даже переопределить эту настройку через пользовательскую базу данных (добавляем поле, правим запрос). Позже это опробую.

```
sieve_dir = ~/
```

личные сценарии пользователя с расширением `.sieve`. При использовании `ManageSieve`, это так же каталог, в который загружаются сценарии.

```
/usr/local/etc/rc.d/dovecot restart
```

Смотрим нет ли ругани...

Внедрили, теперь можно написать скрипты и посмотреть как работает, но писать лень, и для простых пользователей необходимо более простое решение, чем написание скриптов `.sieve` в сеансе SSH или заливание их к себе в домашний каталог по FTP.

Настроим **ManageSieve Server**, скопируем, для начала, чудесно появившийся новый конфиг файл

```
cp /usr/local/share/doc/dovecot/example-config/conf.d/20-managesieve.conf /usr/local/etc/dovecot/conf.d/20-managesieve.conf
protocols = $protocols sieve
```

```
service managesieve-login {
  inet_listener sieve {
    port = 4190
  }
}
```

```
/usr/local/etc/rc.d/dovecot restart
netstat -na | grep LISTEN
tcp4 0 0 *.4190 *.* LISTEN
```

видим что у нас поднялся еще один сервис, теоретически он должен позволять рулить .sieve скриптами, но как им воспользоваться? Смотрим что есть у thunderbird-а в дополнениях. Нам повезло, доступно дополнение sieve, которое не заработало (Исправлено: скорее всего оно рабочее, но будьте внимательны, есть подозрение, что не все дополнения работают если у Вас несколько методов аутентификации, либо берется простейший (PLAIN) из возможных. На момент тестирования у меня была аутентификация "зашифрованный пароль"), попробуем другое sieve out of office (2.0.0). Sieve out of office заработало из коробки, но по функциональности полностью соответствует своему названию, включить перенаправление писем, куда нить, не более. Для тестирования все же воспользуемся предложенной функциональностью. У пользователя test@testdomain.ru создадим правило отправляющее копию пришедшего сообщения на test1@testdomain.ru. В thunderbird идем в инструменты ... Проверяем. Сразу может не заработать, нужно чуть поплясать с бубном, включить/выключить правило и т. п. Результаты получены, sieve работает.

*Рекомендую сразу забыть об этих дополнениях, и настаивать правила через web-интерфейс roundcube, благо там поддержка dovecot sieve заявлена как fine. Строкой выше я имел ввиду что желательно настраивать все правила через одну "систему" или roundcube или расширения.*

Попробуем прикрутить глобальный сценарий, работающий для всех пользователей.

```
90-plugin.conf
sieve_before = /usr/local/etc/dovecot/sieve/spam.sieve

mkdir /usr/local/etc/dovecot/sieve

/usr/local/etc/dovecot/sieve/spam.sieve require ["fileinto"];
# rule:[move_spam]
if anyof (header :contains "Subject" "****SPAM****")
{
  fileinto "Spam";
}
```

```
#chown -R vmail:vmail /usr/local/etc/dovecot/sieve
chown -R mailnull:mail /usr/local/etc/dovecot/sieve
chmod 660 /usr/local/etc/dovecot/sieve
chmod 770 /usr/local/etc/dovecot/sieve/spam.sieve
```

Это мы создали каталог, с правом записи для vmail:vmail. Ситуация такова, что сценарии .sieve сначала интерпретируются – получается файл .svbin, а потом с ним работаем. Этим занимается dovecot, каждый раз при обработке письма по этому скрипту. Можно ему помочь – интерпретировать скрипт заранее, и положить рядом, тем самым интерпретация каждый раз происходить не будет (man sievec). Менять конфиг, и указывать перелопаченый скрипт не нужно.

```
sievec /usr/local/etc/dovecot/sieve/spam.sieve
chmod 770 /usr/local/etc/dovecot/sieve/spam.svbin
```

помним, что у нас, у каждого пользователя имеется почтовый каталог Spam. Мы настроили его принудительное создание в **15-mailboxes.conf**

```
/usr/local/etc/rc.d/dovecot restart
```

Проверяем как работает. Нужно отправить письмо пользователю с темой **\*\*\*SPAM\*\*\***.

У меня уже есть работающая система с web-интерфейсом roundcube, в котором очень просто создавать правила. Текст этого sieve сценария создан roundcube расширением sieve. Я стащил его, предварительно создав себе правило сортировки спама через web интерфейс.

Аналогично можно указать путь к сценарию выполняющемуся после пользовательских сценариев, параметр **sieve\_after =**

Если в этих параметрах будут пути к каталогам, то будут выполняться все сценарии из них.

Все красиво работает, если бы не одно но. Письмо с темой **\*\*\*SPAM\*\*\*** перемещается в каталог Spam и не дойдет до пользовательских фильтров(это справедливо не для всех

возможных действий, а только для тех, которые «заканчивают обработку», к примеру переместить, удалить) Предположим пользователь создает правило пересылать всю почту на свой google аккаунт. А тут приходит ошибочно помеченное как **\*\*\*SPAM\*\*\*** сообщение....

Как вариант вместо **sieve\_before**, можно использовать **sieve\_after**, либо **sieve\_default**

Что еще касаясь sieve можно прописать в 90-plugin.conf (тупо перевод мануала)

```
sieve_default = (v0.3+)
```

Расположение файла сценария Sieve, который запускается на выполнение ТОЛЬКО если частный сценарий пользователя не существует.

```
sieve_global_dir =
```

Глобальные скрипты. Можно создать некоторое количество готовых сценариев, которые будут доступны для использования пользователям.

```
sieve_extensions =
```

Какие расширения Sieve языка доступны для пользователей. По умолчанию все поддерживаемые расширения доступны, за исключением устаревших, расширений, которые требуют явного конфигурирования или те, которые находятся в стадии разработки. Некоторые системные администраторы могут отключить определенные расширения Sieve или включить те, которые не доступны по умолчанию. Все поддерживаемые расширения, перечисленные [здесь http://wiki2.dovecot.org/Pigeonhole/Sieve#features](http://wiki2.dovecot.org/Pigeonhole/Sieve#features). Как правило, все включенные расширения должны быть здесь перечислены, но начиная с Sieve версии 0.1.7, можно использовать '+' и '-', чтобы определить различия по отношению к умолчанию. Например sieve\_extensions = +imapflags[b] разрешит использовать устаревшее расширение imapflags в

дополнение ко всем включенным по умолчанию.

```
sieve_global_extensions = (v0.3+)
```

Какие расширения [b]Sieve языка доступны ТОЛЬКО в глобальном сценарии. Это может быть использовано для ограничения использования некоторых Sieve расширений, например, когда эти расширения могут вызвать проблемы безопасности. Этот параметр имеет более высокий приоритет, чем `sieve_extensions` (см. выше), что означает, что эти расширения не доступны для личных сценариев пользователей. Синтаксис этого параметра похож на `sieve_extensions`. По умолчанию пусто.

```
sieve_plugins =
```

Pigeonhole Sieve интерпретатор может иметь свои собственные плагины.

```
sieve_user_log =
```

Путь к файлу, в котором ведется пользовательский журнал.

```
recipient_delimiter = +
```

Разделитель, использующийся, между: пользователь и расширением.

```
sieve_max_script_size = 1M
```

Максимальный размер Sieve сценария. Компилятор откажется компилировать любой сценарий больше, чем этот предел. Если установлено в 0, ограничений на размер сценария не соблюдается.

```
sieve_max_actions = 32
```

Максимальное количество действий, которые могут быть выполнены за одно выполнение сценария. Если установлено в 0, никаких ограничений.

```
sieve_max_redirects = 4
```

Максимальное количество действий перенаправления, которые

могут быть выполнены за одно выполнение сценария. Значение 0 различается в зависимости от версии. Для версии v0.3.0 и выше это означает, что перенаправление запрещено. Для более старых версий, это означает, что число перенаправлений не ограничено, так что будьте осторожны.

## Extension-specific Configuration

Некоторые расширения языка Sieve могут/нуждаются в дополнительном конфигурировании. Рассмотрим на примере `vacation-seconds`.

Расширение Sieve `vacation`(отпуск) определяет механизм для создания автоматических ответов на входящие сообщения электронной почты. В оригинальном расширении `vacation`, интервал, как часто отправлять ответы на определенный контакт, указывается в днях с минимум в один день. Если необходимо отправлять ответы чаще чем один день, можно использовать `vacation-seconds`. Это позволяет задать минимальный интервал в секундах, с минимумом равным нулю (всегда посылать ответ), в зависимости от административной настройки.

Расширение `vacation` доступно по умолчанию. В отличие от него, расширение `vacation-seconds` не доступно по умолчанию и должно быть включено явно, добавлением в настройке **`sieve_extensions`**.

Приступим к настройке.

```
90-sieve.conf
sieve_extensions = +vacation-seconds
sieve_vacation_min_period = 1h
sieve_vacation_default_period = 1d
```

Проверить, нужно подождать `roundcube`.

Следующие параметры могут быть настроены для расширение `vacation` (значения по умолчанию указаны):

```
sieve_vacation_min_period = 1d
```

Это определяет минимальный период, Минимум 0 означает, что пользователи могут указать интерпретатору Sieve отправлять

сообщения в ответ на каждое входящее сообщение удовлетворяющее остальным критериям. Нулевое значение, не рекомендуется.

```
sieve_vacation_max_period = 0
```

Максимальный срок. Заданное значение должно быть больше, чем `sieve_vacation_min_period`. Значение 0 имеет особое значение: это означает, что не существует верхнего предела.

```
sieve_vacation_default_period = 7d
```

Это указывает период по умолчанию. Заданное значение должно лежать между

```
sieve_vacation_min_period и sieve_vacation_max_period.
```

При неверных значениях параметров сделает Sieve интерпретатор напишет предупреждение в лог и вернется к значениям по умолчанию.