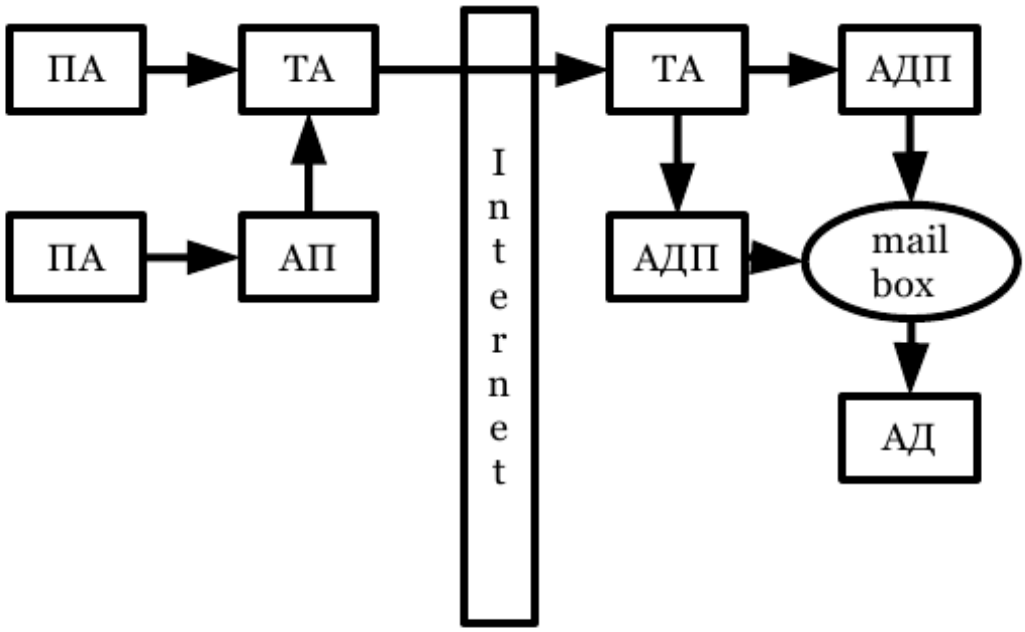
Расширенное администрирование FreeBSD.
Блок 4.
v 1.05

Оглавление

Электронная почта

Одной из важных систем, работающих на серверах FreeBSD, является система электронной почты. Администратор должен понимать как она устроена и уметь ее настраивать.
В этом блоке будут рассмотрены вопросы устройства системы электронной почты и настройка транспортного агента sendmail.



Компоненты электронной почты

Система электронной почты состоит из нескольких компонентов. Они могут быть выполнены в виде одной или нескольких программ.
Можно выделить следующие компоненты:

- Пользовательский агент (ПА).
- Транспортный агент (ТА).
- Агент подачи почты (АП).
- Агент доставки почты (АДП).
- Агент доступа (АД).

Пользовательский агент

При помощи пользовательских агентов пользователи составляют и отправляют письма. Агент должен сформировать тело письма согласно стандарту и передать его на отправку транспортному агенту или агенту подачи.
В роли пользовательских агентов выступают такие программы как: The Bat, Outlook и Outlook Express. Если говорить про FreeBSD: Evolution, KMail, pine и др.
При передачи письма транспортному агенту и агенту подачи используется протокол SMTP.

Транспортный агент

Транспортный агент выполняет две основные задачи:

- Прием почты от пользовательского агента и пересылка ее на другой

транспортный агент
- Прием почты от других транспортных агентов

При приеме почты от пользователя он должен проверить правильность адреса назначения, возможность доставки почты и доставить почту по назначению.
На другой стороне транспортный агент проверяет: предназначено ли это письмо для данной машины (домена), есть ли почтовый ящик пользователя на машине. После проверок он принимает письмо и передает его Агенту доставки почты.
В мире UNIX существует большое количество программ, реализующих функции транспортного агента. Среди наиболее популярных бесплатных реализаций транспортных агентов можно выделить sendmail, postfix, exim и qmail. Каждая из программ имеет свои достоинства и недостатки, наиболее распространенным является sendmail — один из самых первых транспортных агентов.

Агент подачи почты

Агенты подачи почты — это одна из разновидностей режима работы транспортного агента. Агенты подачи применяются на почтовых узлах с напряженным трафиком. Его задача облегчить работу основного транспортного агента.
Агент подачи:

- Проверяет, являются ли имена узлов полностью определенными
- Модифицирует заголовки сообщений, полученных от неправильно работающих пользовательских агентов
- Проверяет все ошибки перед передачей письма транспортному агенту.

Агент подачи слушает запросы на 587 порту, поэтому все пользовательские агенты необходимо настроить таким образом, чтобы они отправляли почту на этот порт.
Все особенности работы агента подачи описаны в RFC 2476.

Агент доставки почты

Транспортный агент после получения почты сам не доставляет ее в почтовый ящик пользователя. Он передает ее агенту доставки почты, задача которого доставить почту в почтовый ящик пользователя.
В качестве агента доставки может выступать простейшая программа, которая просто складывает почту. Существуют и более сложные программы, которые при доставке почты могут осуществлять ее фильтрацию, например, procmail.
Назначение программы доставки почты детально представлено в RFC2476.

Агенты доступа

Агенты доступа позволяют пользователю получить доступ к своему почтовому ящику. Они выполнены в виде программ, организующих доступ к почтовому ящику по протоколу pop3 или imap.
Если пользователь работает локально на машине, на которой хранятся его почтовые ящики, он может получить доступ без агента доступа, обращаясь к ним напрямую. По такому принципу работают mail и pine.

Протокол SMTP

SMTP — Simple Mail Transfer Protocol.
ESMTP — Extended SMTP.

В качестве основного протокола взаимодействия в системе электронной почты используются протоколы SMTP (Simple Mail Transfer Protocol) и ESMTP (Extended SMTP). Они описаны в RFC2821, 1869,1870,1891 и 1985.
Протокол SMTP задумывался как простой протокол взаимодействия, при помощи которого пользователь мог напрямую общаться с почтовым транспортным агентом. Конечно же, сейчас пользователи сами не работают с транспортными агентами. Для облечения работы используются пользовательские агенты. Но они (пользовательские агенты) для взаимодействия с транспортными агентами используют протокол SMTP или ESMTP.
В качестве примера можно показать, как пользователь при помощи программы telnet может подключиться к транспортному агенту и отправить письмо.
Пользователь, при помощи транспортного агента может только отправлять письма. Прием почты происходит при помощи агентов доступа.
Для подключения к почтовому транспортному агенту использовалась программа telnet, с явным указанием порта 25.

```
c1# telnet localhost 25
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 c1.any.com ESMTP Sendmail 8.14.3/8.14.3; Sun, 27 Sep 2009 16:07:03 +0400 (MSD)
```

При отправке почты обязательно следует указать от кого эта почта отправляется. Для этого используется команда mail from протокола SMTP. Транспортный агент по умолчанию требует в адресе отправителя обязательное указание домена в e-mail отправителя.

```
mail from: admin@localhost
250 2.1.0 admin@localhost... Sender ok
```

Если с почтовым сообщением отправителя все в порядке, транспортный агент выдает подтверждение. Теперь при помощи команды rcpt to указывается адрес получателя.

```
rcpt to: root@localhost
250 2.1.5 root@localhost... Recipient ok
```

При помощи команды data начинается ввод текста письма. Последняя строка должна содержать единственный символ «.», который обозначает конец письма.

```
data
354 Enter mail, end with "." on a line by itself
test mail
.
```

После ввода символа «.» транспортный агент принимает письмо на доставку, о чем выдается соответствующее сообщение.

```
250 2.0.0 n8RC73hp002868 Message accepted for delivery
```

Для завершения сеанса связи используется команда quit.

```
quit
221 2.0.0 c1.any.com closing connection
Connection closed by foreign host.
```

Как видно из приведенного примера, транспортному агенту для доставки почты необходимо указать два параметра: адрес отправителя и адрес получателя, а также тело письма.

Заголовки, которые Вы видите в теле письма, например, From:, To:, X-Mailer: и т.д., транспортным агентом рассматриваются как тело письма и напрямую не обрабатываются. Хотя почтовый сервер можно настроить таким образом, чтобы он проверял или изменял такие заголовки.

При доставке письма транспортный агент добавляет в его заголовок строки, свидетельствующие о прохождении письма через транспортный агент.

Каждый транспортный агент, через который проходит письмо, добавляет похожую строку. По этим строкам можно определить маршрут прохождения письма.

Выше был показан достаточно простой пример взаимодействия сервера и клиента. Не учитывающий шифрование канала и авторизацию пользователя.

Вопросы

1. Перечислите основные компоненты, из которых состоит система электронной почты.

2. Какие функции выполняет транспортный агент?

3. Как пользователи могут получить доступ к своим почтовым ящикам?

Sendmail

Конфигурационные файлы.
Конфигурация при помощи препроцессора m4.
Режимы работы.
Запуск и управление.
Почтовые ящики пользователей.
Работа с очередями.
Журнальная регистрация.

Sendmail — это наиболее распространенный транспортный агент в BSD-системах. Sendmail был написан Эриком Оллманом в 1983 году и в дальнейшем дорабатывался различными разработчиками.
Нет таких действий, которые бы Sendmail не мог делать с почтовыми сообщениями, за исключением проверки содержимого письма. Но для его проверки он может передать письмо сторонней программе. Обычно так включается проверка на вирусы в почтовых сообщениях.
В этом блоке подробно будет рассмотрено, каким образом можно настроить Sendamil для работы в различных ситуациях.

Конфигурационные файлы sendmail

/etc/mail/sendmail.cf
/etc/mail/submit.cf
Дополнительные конфигурационные файлы в директории /etc/mail.

Основной конфигурационный файл программы — sendmail.cf. Обычно он находится в директории /etc/mail. В файле определяются основные параметры и особенности работы sendmail, в том числе и дополнительные конфигурационные файлы, которые тоже обычно находятся в директории /etc/mail.
Кроме файла sendmail.cf в системе может присутствовать файл submit.cf, являющийся конфигурационным файлом агента подачи почты.
Какой из перечисленных файлов будет использоваться в качестве конфигурационного (то есть, в каком режиме будет работать программа) зависит от опций, с которыми sendmail запускается:

-Ac — запуск в качестве агента подачи почты с использованием конфигурационного файла submit.cf.
-Am — запускается в качестве основного почтового сервера с использованием файла sendmail.cf. (Параметр по умолчанию).

Файл submit.cf обычно никогда не редактируется.
Если посмотреть на содержимое файла sendmail.cf, в нем можно найти строки похожие на:

```
R$* < @ $=w > $*              $: $1 < @ $2 . > $3
R$* < @ $=M > $*              $: $1 < @ $2 . > $3
R$* < @ $={VirtHost} > $*     $: $1 < @ $2 . > $3
```

Формат файла изначально создавался с учетом облегчения синтаксического анализа

файла и поэтому он мало понятен обыкновенным пользователям. Обычно администратору крайне редко приходится редактировать sendmail.cf. Для создания файла используются специальные макросы препроцессора m4, облегчающие задачу его создания и изменения.

Конфигурация sendmail при помощи препроцессора m4

```
m4 my.mc > /etc/mail/sendmail.cf
            или
make cf
```

Как было сказано в предыдущем разделе, препроцессор m4 облегчает администратору процесс создания и изменения конфигурационных файлов sendmail.cf и submit.cf. Для создания этих файлов необходимо проверить наличие установленного препроцессора m4.

```
which m4
/usr/bin/m4
```

Для получения файла sendmail.cf необходимо создать файл, в котором будут описаны используемые макросы. Затем его пропускают через препроцессор и в результате получают конфигурационный файл.

```
m4 my.mc > /etc/mail/sendmail.cf
```

Препроцессор m4 имеет очень строгий синтаксис. Любой лишний пробел, неправильное указание скобок ведет к появлению сообщения об ошибке или созданию неправильного файла.

В качестве начала комментария используется инструкция dnl. Все, что будет написано после этой инструкции до конца строки, считается комментарием. Рекомендуется в конце каждой строки писать dnl для того, чтобы игнорировать случайные пробелы, которые были помещены в конце макроса.

Если необходимо указывать параметры или строки, их берут в кавычки. Следует обратить особое внимание на то, что открывающая кавычка — это обратная одинарная кавычка «`» (на клавиатуре расположена там же где и буква Ё), а закрывающая — это одинарная прямая кавычка «'» (на клавиатуре расположена там же где и буква Э).

Так же, в файле на языке препроцессора можно использовать инструкцию divert, при помощи которой можно писать комментарии сразу на нескольких строках.

```
divert(-1)
#
# Copyright (c) 1983 Eric P. Allman.
# Copyright (c) 1988, 1993
#      The Regents of the University of California.
Divert(0)
```

divert(-1) — очищает буфер макросов от данных, оставшихся от предыдущих попыток.
divert(0) — применяется для обозначения начала макроса.

Макрос VERSIONID

Макрос VERSIONID применяется для идентификации версии создаваемого конфигурационного файла. Все, что указывается в качестве параметра, будет без изменений помещено в выходной файл, сразу после символа комментария.

```
VERSIONID(`My super mail server')dnl
```

Макрос OSTYPE

В разных UNIX принято свое месторасположение и название дополнительных конфигурационных файлов sendmail. Каждый файл можно определить отдельно при

помощи директивы define. Но для того, чтобы эти директивы не описывать каждый раз в файле mc, используют макрос OSTYPE.

```
OSTYPE(freebsd6)dnl
```

Макрос DOMAIN

Если у Вас есть много почтовых серверов с одинаковыми параметрами, можно создать файл в котором эти параметры будут описаны. Затем при помощи макроса DOMAIN подключить этот файл.

```
DOMAIN(my-domain)dnl
```

Например, файл generic.m4 из директории /usr/share/sendmail/cf/:

```
VERSIONID(`$Id: generic.m4,v 8.15 1999/04/04 00:51:09 ca Exp $')
define(`confFORWARD_PATH', `$z/.forward.$w+$h:$z/.forward+$h:$z/.forward.$w:$z/.forward')dnl
define(`confMAX_HEADERS_LENGTH', `32768')dnl
FEATURE(`redirect')dnl
FEATURE(`use_cw_file')dnl
EXPOSED_USER(`root')dnl
```

Макрос MAILER

Этот макрос применяется для определения того, каким образом sendmail может доставлять почту. Точно так же, как и в предыдущих макросах, существует специальная директория /usr/share/sendmail/cf/mailer, в которой описаны агенты доставки почты, которыми может пользоваться sendmail.

Например, если предполагается доставка почты в локальные почтовые ящики и на удаленные транспортные агенты по протоколу smtp, необходимо описать эти возможности:

```
MAILER(local)dnl
MAILER(smtp)dnl
```

На самом деле mailer smtp включает сразу несколько транспортных агентов: smtp, esmtp, dsmtp, smtp8 и relay. Если при доставке почты в локальные почтовые ящики требуется ее фильтрация, рекомендуется добавить поддержку программы procmail:

```
MAILER(procmail)dnl
```

Макрос FEATURE

Макрос применяется для описания различных особенностей почтового сервера. Подавляющее большинство параметров будут определяться при помощи именно этого макроса. Более подробно макрос FEATURE мы будем рассматривать при конфигурации sendmail в других разделах. Кроме перечисленных макросов существует еще небольшое количество других типов макросов.

Порядок описания макросов в файле препроцессора m4

Порядок описания файлов имеет значение. Особое внимание необходимо обратить на макросы MAILER. Они должны располагаться в конце файла, но перед описанием различных локальных конфигураций.

Ниже показан предпочтительный порядок описания макросов:

- VERSIONID
- OSTYPE
- DOMAIN
- local macro definitions
- FEATURE
- MAILER

Режимы работы sendmail

Режимы работы программы задаются при ее старте опцией -b

Sendmail поддерживает несколько режимов работы.
Режимы можно указать при запуске с помощью опции -b.

Параметр	***Значение***
-bd	Программа работает в режиме демона. Открывает на прослушивание 25-й порт.
-bD	То же самое, что и предыдущий параметр, но программа работает в foreground режиме. Применяется при отладке.
-bh	То же самое что и hoststat. Отображает статистическую информацию.
-bH	Удаляет старую статистическую информацию. То же самое, что и программа purgestat.
-bi	Инициализация псевдонимов. То же самое, что и программа newaliases.
-bm	Режим обычной доставки почты. Это значе- ние принимается по умолчанию.
-bp	Показать содержимое очереди на отправку. То же самое, что и программа mailq.
-bs	Программа читает данные из стандартного ввода, а не из порта 25.
-bt	Режим проверки адресов. Применяется толь- ко при отладке.

Запуск и управление программой

sendmail -bd -q25m
или
/etc/rc.d/sendmail restart

После изменения конфигурационного файла sendmail.cf для того, чтобы все изменения вступили в силу, программа должна перечитать его. Несмотря на то, что в разных книгах, описывающих настройку sendmail, предлагается послать сигнал HUP программе, этого оказывается недостаточно и лучший способ — это перезапустить программу.
Запуск возможен как «вручную», так и при помощи инициализационных скриптов.

```
/usr/sbin/sendmail -L sm-mta -bd -q25m
/usr/sbin/sendmail -L sm-msp-queue -Ac -q25m
```

В стартовых скриптах для запуска sendmail используется стартовый скрипт /etc/rc.d/sendmail.

```
/etc/rc.d/sendmail start
```

Параметр -bd означает, что данный экземпляр программы откроет на прослушивание 25-й порт и будет выполнять функции транспортного агента. То есть, он будет почтовым сервером.
Параметр -Ac говорит, что этот экземпляр программы откроет на прослушивание 587 порт и будет выполнять функции агента подачи.
Параметр -q определяет периодичность, с которой sendmail будет просматривать очередь на отправление и пытаться отослать накопившиеся письма. 25m — означает задержку в 25 минут.
Останов программы происходит путем посылки сигнала TERM:

```
killall sendmail
```

Поскольку сигнал TERM — это корректное завершения программы, sendmail сам завершит все процессы и сохранит необходимую информацию.

Почтовые ящики пользователей

Существует несколько разновидностей хранения почты в почтовых ящиках.

- file based система, где для каждого пользователя создается отдельный файл, в котором будет находиться вся его почта.
- directory based система, где в домашней директории пользователя создается отдельная директория, в которой каждое письмо размещается в отдельном файле.

Sendmail использует первый вариант хранения почты. Все почтовые ящики пользователей располагаются в директории /var/mail. Но его можно настроить и на второй вариант. В этом случае следует воспользоваться услугами программы procmail.

Для того, чтобы завести почтовый ящик необходимо при помощи стандартных программ добавить пользователя в систему. У пользователя обязательно должен быть установлен пароль, поскольку для доступа к своему почтовому ящику он должен себя идентифицировать.

Обычно пользователям, имеющим почтовые ящики на сервере, нет необходимости разрешать вход в систему по ssh. Поэтому у них вместо shell (седьмое поле в файле /etc/passwd) указывают либо программу /sbin/nologin, либо, если хотят объяснить причину, пишут простейший скрипт. Например такой:

```
#!/bin/sh
echo; echo; echo
echo "*******************************************"
echo "Уважаемый $USER. Вам запрещен доступ в систему"
echo "С уважением, Администратор"
echo "*******************************************"
sleep 15
exit 1
```

Когда пользователь попытается войти в систему, ему на экран будет выдано соответствующее сообщение, и в течение 15-ти секунд он может его читать. Затем программа завершает свою работу и пользователь из системы выходит.

Следующий вопрос — необходимо ли создавать домашнюю директорию пользователя? Ответ на него зависит от того какую систему хранения почты вы решили использовать и от протокола, применяемого для доступа к почтовым ящикам пользователя.

В случае использования протокола POP3, домашнюю директорию создавать не надо. Если для доступа к почтовым ящикам используется протокол IMAP — директория должна существовать. При использовании IMAP так же рекомендуется входящую почту пользователя хранить в его домашней директории.

Работа с очередью на отправку

/var/spool/mqueue
/var/spool/clientmqueue

Sendmail имеет четыре режима доставки почты:

- ***Фоновый***. Письмо доставляется немедленно. Для доставки sendmail порождает отдельный процесс.
- ***Интерактивный***. Письмо доставляется немедленно, но его отправкой занимается один и тот же процесс.
- ***С постановкой в очередь***. Письмо попадает в очередь на отправку, откуда его потом извлекает обработчик очереди.
- ***Отложенный***. Режим отложенной доставки похож на режим очереди, но в очередь так же попадают все операции поиска в таблицах, в DNS и т.д.

По умолчанию sendmail использует фоновый режим, при котором он старается немедленно доставить письмо. Но если письмо не удается доставить сразу или система сильно нагружена, оно попадает в очередь на отправку.

Письма, попавшие в очередь на отправку, находятся в директории /var/spool/mqueue. У агента подачи почты существует своя очередь на отправку — /var/spool/clientmqueue.

Для просмотра содержимого очереди на отправку можно воспользоваться командой mailq.

Для немедленного разбора очереди на отправку используют sendmail с опцией -q. В этом случае программа старается отправить все письма, находящиеся в очереди. Если письмо не удается отправить, оно остается в очереди на отправку. После просмотра всей очереди, этот экземпляр sendmail завершает свою работу.

Иногда возникает ситуация, когда очередь на отправку начинает замедлять работу sendmail. В этом случае можно пойти различными путями:

- Скопировать очередь в другую директорию. Основной экземпляр sendmail продолжит работу с нормальной скоростью. А для отправки почты из новой директории, запустить отдельный экземпляр sendmail с опциями:
- -q -O QueueDirectory=new_queue_dir.
- Если ситуация с заполнением очереди на отправку возникает часто, разделить ее на несколько групп и использовать макрос QUERY_GROUP.
- Определить опцию FALLBACK_MX, задающую пересылку почты на другой компьютер, если ее не удалось доставить с первой попытки.
- Определить опцию HOST_STATUS_DIRECTORY, указывающую директорию, в которой будет сохраняться информация о каждом компьютере, куда должна быть доставлена почта, между попытками доставить почту. В результате заметно повышается производительность сервера, на котором в очереди накапливается большое количество писем в очереди на отправку.

Журнальная регистрация

/var/log/maillog

Sendmail все сообщения об ошибках и своем состоянии передает в систему Syslog при помощи средства mail.
Все сообщения начинаются со строки sendmail. Если Вы запускаете несколько процессов sendmail, например, основной процесс демон и агент подачи почты, при помощи параметра -L можно определять, с какого слова будут начинаться сообщения программ.
Стартовый скрипт при запуске sendmail передает ему параметр -L sm-mta или -L sm-smp-queue. Таким образом все сообщения, начинающиеся со слова sm-mta, были посланы транспортным агентом, а начинающиеся с sm-smp-queue, были переданы в Syslog агентом подачи почты.
По умолчанию Syslog настроен таким образом, что все сообщения попадают в файл /var/log/maillog. В этот файл попадает вся информация о принятых и переданных письмах, об ошибках, возникнувших во время передачи, о запуске программы.
Например, при старте в файл попадают следующие строки:

```
Mar 31 09:14:22 home sm-mta[1090]: starting daemon (8.12.10): SMTP+queueing@01:00:00
Mar 31 09:14:22 home sm-msp-queue[1099]: starting daemon (8.12.10): queueing@01:00:00
```

При отправке письма в журнальный файл попадает несколько строк:

```
Sep 28 15:48:00 c20 sm-mta[1392]: starting daemon (8.14.3): SMTP+queueing@00:30:00
Sep 28 15:49:24 c20 sendmail[1400]: n8SBnN28001400: from=admin, size=32, class=0, nrcpts=1,
msgid=<200909281149.n8SBnN28001400@c20.u20.any.com>, relay=root@localhost
Sep 28 15:49:24 c20 sm-mta[1401]: n8SBnOq8001401: from=<admin@c20.u20.any.com>, size=359,
class=0, nrcpts=1, msgid=<200909281149.n8SBnN28001400@c20.u20.any.com>, proto=ESMTP,
daemon=IPv4, relay=localhost [127.0.0.1]
Sep 28 15:49:24 c20 sendmail[1400]: n8SBnN28001400: to=root, ctladdr=admin (1001/0),
delay=00:00:01, xdelay=00:00:00, mailer=relay, pri=30032, relay=[127.0.0.1] [127.0.0.1],
dsn=2.0.0, stat=Sent (n8SBnOq8001401 Message accepted for delivery)
Sep 28 15:49:24 c20 sm-mta[1402]: n8SBnOq8001401: to=<root@c20.u20.any.com>,
ctladdr=<admin@c20.u20.any.com> (1001/0), delay=00:00:00, xdelay=00:00:00, mailer=local,
pri=30620, relay=local, dsn=2.0.0, stat=Sent
Sep 28 16:12:42 c20 sm-mta[1477]: n8SCCgEn001477: <info@u20.any.com>... User unknown
Sep 28 16:12:42 c20 sendmail[1476]: n8SCCgxP001476: to=info@u20.any.com, ctladdr=admin
(1001/0), delay=00:00:00, xdelay=00:00:00, mailer=relay, pri=30052, relay=[127.0.0.1]
[127.0.0.1], dsn=5.1.1, stat=User unknown
```

По этим строкам можно определить стадии отправки письма и то, какие ошибки возникали в процессе работы сервера.

Вопросы

1. Как называется и где находится основной конфигурационный файл программы sendmail?

2. При помощи каких макросов определяется способ доставки почты программой sendmail?

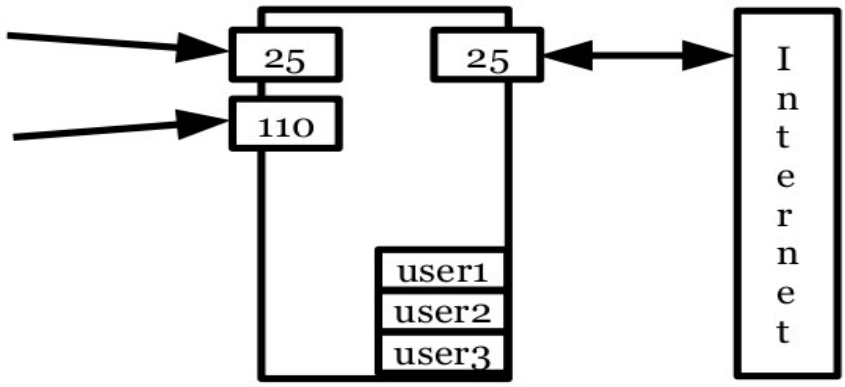
3. Какой параметр заставляет sendmail работать в режиме демона с открытием на прослушивание 25 порта?

4. Где располагаются почтовые ящики пользователей?

5. При помощи какой команды можно посмотреть очередь на отправку?

6. С какой опцией необходимо запустить sendmail для того, чтобы он попытался немедленно отправить письма, находящиеся в очереди на отправку?

Настройка простого почтового сервера



В качестве первого примера рассмотрим, как можно настроить простой почтовый сервер, обслуживающий небольшое количество почтовых ящиков.
Конфигурация sendmail по умолчанию имеет некоторые ограничения в работе:

- Принимает почту, адресованную только на машину, на которой он установлен.
- Позволяет отправлять почту только пользователям, работающим непосредственно на сервере.
- Слушает запросы на 25-м порту, но только на lo интерфейсе.

Предположим, что машина, на которой установлен sendmail, имеет имя smtp.u20.any.com. На машине заведена учетная запись пользователя — info. В DNS сервере, отвечающем за домен u20.any.com, существует запись MX:

```
u20.any.com. IN MX 5 smtp.u20.any.com.
```

Запись показывает всем почтовым серверам, что почта, предназначенная для домена u20.any.com (например: info@u20.any.com), должна пересылаться на почтовый сервер, расположенный на машине smtp.u20.any.com.
По умолчанию sendmail принимает почту, предназначенную для машины, на которой он установлен, то есть. когда e-mail выглядит как info@u20.any.com. Почта info@a.com не будет принята сервером.
Второе ограничение: запрещенная по умолчанию пересылка писем (RELAY) через сервер. Ограничение предназначено для защиты Вашего сервера. Если бы sendmail устанавливался с разрешенной пересылкой (как в старых версиях программы), то любой пользователь Интернет мог бы использовать его для отправки почты. Спамеры ищут в сети сервера с открытым RELAY и стараются пользоваться ими для пересылки спама. Если посмотреть на логи почтовых серверов, вы увидите регулярные попытки отправки писем через ваш сервер и соответствующие сообщения об ошибках.
Но проблема даже не в спамерах, а DNSBL серверах, которые собирают информацию о спамерах. Если Ваш сервер попадет в такой список (а он туда обязательно попадет, если Вы не запретите пересылку почты из Internet), то другие почтовые сервера, пользующиеся услугами DNSBL-серверов, перестанут принимать почту, посланную Вашим сервером.

Создание файла конфигурации

```
FEATURE(use_cw_file)
FEATURE(access_db)
```

При создании файла конфигурации мы будем использовать стандартные макросы VERSIONID, OSTYPE, FEATURE и MAILER.
Все особенности поведения программы sendmail, описываются при помощи макроса FEATURE. Нам необходимо заставить sendmail принимать почту для домена u20.any.com и разрешить пересылку почты внутренним клиентам нашей сети.

Средство use_cw_file

sendmail будет искать имена доменов, для которых он принимает почту, в файле /etc/mail/local-host-names.

```
FEATURE(use_cw_file)dnl
```

В файле /etc/mail/local-host-names необходимо записывать имена доменов, по одному на строку. Например:

```
u20.any.com
```

Последняя строка обязательно должна быть пустой! Если этого не сделать, она не будет прочитана и sendmail не будет принимать почту для домена, описанного в последней строке.

Ретрансляция

Для того, чтобы клиенты, располагающиеся во внутренней сети, могли отсылать почту через наш сервер, им необходимо это разрешить. Существует много способов разрешения ретрансляции почты. Наиболее популярный — использование средства access_db. Это средство позволяет sendmail пользоваться специальной базой доступа, которая выполняется в виде hash таблицы.

```
FEATURE(`access_db',`hash -T<TMPF> /etc/mail/access.db')dnl
```

В определении средства access_db указывается файл access.db, в котором будет храниться база доступа. access.db — это бинарный файл, содержащий в себе hash таблицу. Для того, чтобы создать этот файл или внести в него изменения, сначала создают текстовый файл, который при помощи программы makemap преобразуют в бинарный.
Обычно текстовый файл источник access находится там же, где и access.db.
Поскольку хеш-таблица состоит из уникальных ключей поиска и значений, в файле access сначала пишется ключ, а после пробела или табуляции, значение которое соотвествует ключу.
В качестве ключа можно использовать:

- IP адреса, в том числе и неполные. Например: 172.16.1.254 или 10.5.10.21
- Имена домена. Например: u20.any.com
- E-mail. Например: info@u20.any.com
- Неполные части e-mail. Например: info@
- Теги: Connect, From, To, Spam. Например: To:sales@

В поле значение можно использовать различные ключевые слова, но сейчас мы ограничимся только RELAY, при помощи которого разрешается пересылка почты.

Для разрешения пересылки почты в файл access необходимо поместить следующие строки:

```
Connect:127.0.0.1      RELAY
Connect:172.16.1      RELAY
```

Последняя строка разрешает пересылку почты со всех машин, расположенных в сети 172.16.1.0/24.
Для того, чтобы файл access был преобразован в файл access.db, необходимо выполнить следующую программу:

```
makemap hash access < access
```

Программа makemap создаст файл, содержащий hash таблицу. Файл будет называться access.db (расширение можно не указывать явно). Данные будут браться из файла access. Обратите внимание на то, что в примере не указаны полные пути к файлам, поэтому программу makemap необходимо запускать либо в директории /etc/mail, либо указывать полные пути к файлам access.db и access.

Лабораторная работа А

Настройка простого почтового сервера

Задачи	***Описание***
1. Создание mc файла	1. Перейдите в директорию /etc/mail. 2. В этой директории создайте файл cX.uX.any.com.mc следующего содержания: `OSTYPE(freebsd6)` `FEATURE(use_cw_file)` `FEATURE(access_db, `hash -o -T<TMPF> /etc/mail/access')` `MAILER(local)` `MAILER(smtp)` 3. Сохраните файл.

3. Подготовка дополнительных конфигурационных файлов.	1. В конец файла /etc/mail/local-host-names добавьте имя Вашего домена.
	`uX.any.com`
	Не забудьте последнюю строку оставить пустой.
	2. В файл /etc/mail/access добавьте строки, разрешающие пересылку почты с вашей подсети и localhost.
	`127.0.0.1 RELAY`
	`172.16.1  RELAY`
	3. Создайте файл access.db. Перейдите в директорию /etc/mail и выполните программу makemap.
	`make maps`

2. Создание файла sendmail.cf.	1. Для создания файла sendmail.cf используйте `make install`
4. Запуск программы.	1. После подготовки всех конфигурационных файлов перезапустите почтовую систему. `/etc/rc.d/sendmail restart` 2. Убедитесь, что в списке процессов присутствует два процесса sendmail. 3. Посмотрите содержимое файла /var/log/maillog.

Если во время выполнения лабораторной работы возникли ошибки, опишите ниже эти ошибки и способы их решения.

Вопросы

1. При помощи какого макроса включается поддержка файла local-host-names?

2. При помощи какой программы создается база доступа?
Напишите командную строку, которую необходимо выполнить.

3. Какая строка должна быть добавлена в базу доступа, для разрешения пересылки почты с компьютера с IP 10.10.108.1?

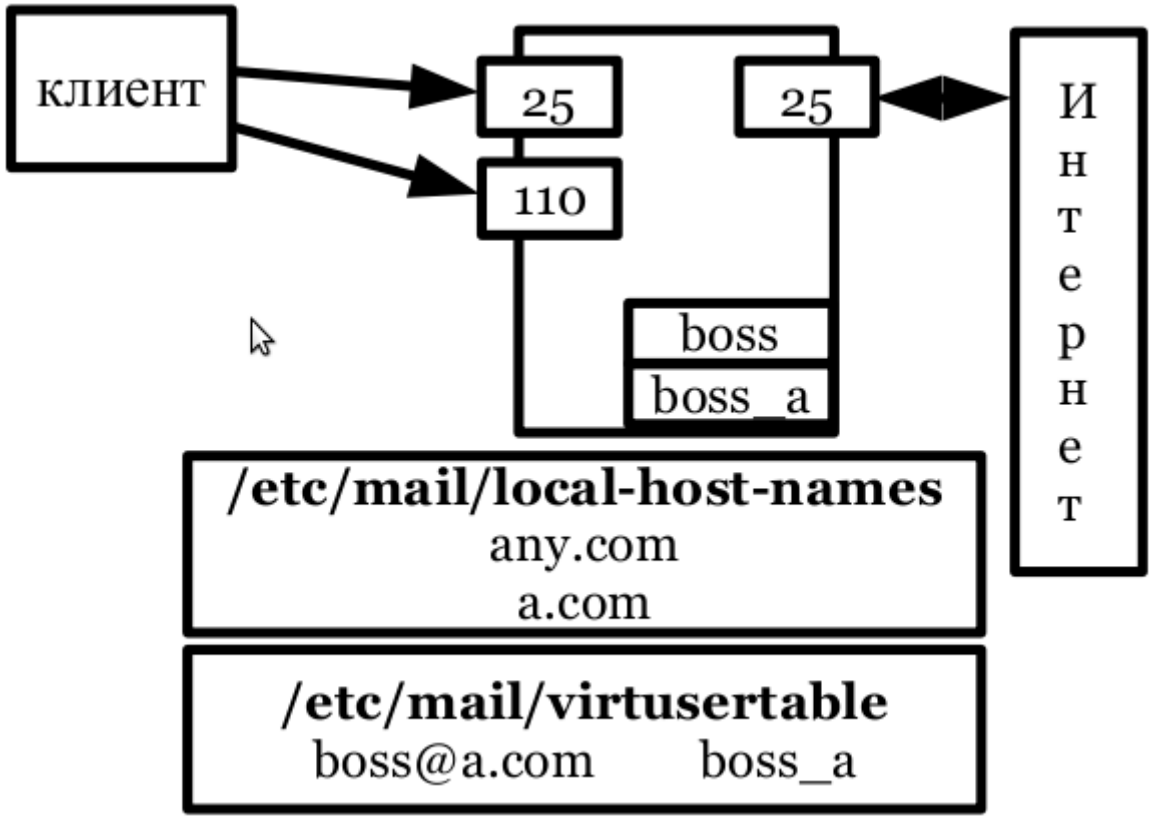
Виртуальный хостинг почтовых серверов

Предоставление почтового ящика ISP.
Виртуальный хостинг почтового сервера.

В этой и следующих главах мы рассмотрим более сложные варианты конфигурации почтовых серверов:

- Как ISP может создавать почтовые ящики пользователям различных организаций.
- Организация виртуального хостинга почтового сервера предприятия, имеющего постоянное соединение с Internet.

Предоставление почтового ящика ISP



Начнем с самого простого случая. Вы работаете в ISP и к Вам обратился клиент, которому необходимо организовать несколько почтовых ящиков для одной организации. У клиента зарегистрирован домен a.com. Какие действия Вам необходимо предпринять в этом случае?

Обычно достаточно завести учетные записи пользователей на Вашем почтовом сервере и разрешить пересылку почты с машин клиентов. Но в нашем случае речь идет об организации, у которой есть свое доменное имя и необходимо, чтобы их e-mail выглядели как info@a.com, а не как info@u20.any.com.

- Необходимо, чтобы запись MX в файле описания зоны a.com, ссылалась на Ваш почтовый сервер smtp.u20.any.com.
- В файл my.mc, из которого потом будет создан файл sendmail.cf Вашего сервера, добавьте средство use_cw_file.
- В файл /etc/mail/local-host-names необходимо добавить домен a.com для того,

чтобы Ваш сервер принимал почту для этого домена.

- В файле c20.u20.any.com.mc необходимо описать средство access_db.
- В файл /etc/mail/access следует добавить строку, разрешающую пересылку почты с компьютеров клиентов через Ваш почтовый сервер.
- Из файла access создать hash таблицу в файле access.db.
- Добавить учетные записи пользователей на почтовом сервере.

Как видите, все действия, которые перечислены выше, ничем не отличаются от действий, которые мы выполняли при настройке простого почтового сервера. За небольшим исключением — в файле local-host-names был добавлен домен a.com. Это значит, что наш сервер будет принимать почту не только для своего домена u20.any.com, но и для домена a.com.

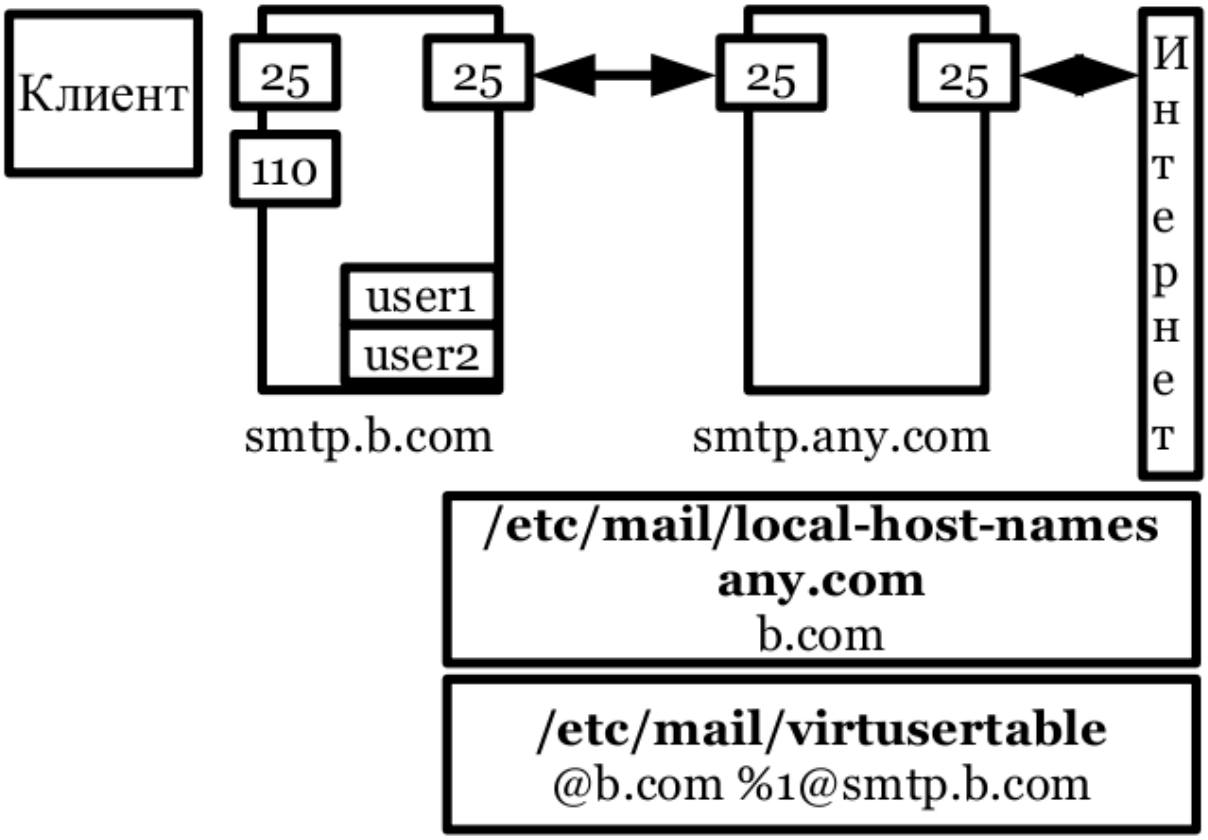
Предположим, что клиент попросил, чтобы у их компании был e-mail info@a.com. А у нас на сервере уже есть пользователь info нашей компании u20.any.com, с e-mail: info@u20.any.com. После записи в файл local-host-names домена a.com, все письма, посланные на info@a.com и info@u20.any.com будут попадать в почтовый ящик одного и того же пользователя — info. Напомню, что часть e-mail, стоящей после символа @ указывает на какую машину необходимо доставить почту. Что необходимо сделать, чтобы почта, посланная на e-mail info@a.com не попадала в почтовый ящик пользователя info, а попадала в почтовый яцик пользователя info_a (которого мы заведем для компании a.com)?

Самым лучшим способом решения этой проблемы является использования средства virtusertable, позволяющее использовать таблицу виртуальных пользователей, при помощи которой легко организовывать виртуальный хостинг почтовых серверов.

- В файл c20.u20.any.com.mc добавляем средство virtusertable.
  ```
  FEATURE(`virtusertable',`hash -o /etc/mail/virtusertable.db')dnl
  ```
- Создаем текстовый файл /etc/mail/virtusertable, в который добавляем строку, которая заставит sendmail пересылать всю почту, приходящую на e-mail info@a.com в локальный почтовый ящик пользователя info_a.
  ```
  info@a.com   info_a
  ```
- Из файла virtusertable при помощи программы `make map` создаем hash таблицу virtusertable.db.
- Из файла c20.u20.any.com.mc создаем файл sendmail.cf.
- Перегружаем почтовый сервер.

На машине клиента необходимо указать имя почтового ящика, откуда он будет забирать почту — info_a. В поле Mail From почтовый адрес будет указан в домене a.com — info@a.com.

Организация виртуального хостинга почтового сервера предприятия, имеющего постоянное подключение к Internet



Предположим, что к нам обратился клиент (b.com), который по каким-либо причинам не хочет, чтобы его почтовый сервер имел прямое подключение к Internet. Наша задача заключается в том, чтобы принимать и пересылать всю почту, предназначенную для домена b.com на их почтовый сервер (smtp.b.com). А почтовый сервер клиента настроить таким образом, чтобы он всю исходящую почту отсылал через наш сервер.
Действия, которые мы будем выполнять на сервере smtp.any.com:

- Запись MX в файле описания зоны b.com должна ссылаться на Ваш почтовый сервер smtp.any.com.
- В файл my.mc, из которого потом будет создан файл sendmail.cf Вашего сервера, добавьте средство use_cw_file.
- В файл /etc/mail/local-host-names необходимо добавить домен b.com, для того, чтобы Ваш сервер принимал почту для этого домена.
- В файле my.mc необходимо описать средство access_db.
- В файл /etc/mail/access следует добавить строку, разрешающую пересылку почты с сервера smtp.b.com через Ваш почтовый сервер.
- Из файла access создать hash таблицу в файле access.db.
- В файле my.mc добавить средство virtusertable.
- В файл /etc/mail/virtusertable добавить строку, которая заставит sendmail пересылать всю почту, пришедшую на домен b.com на машину smtp.b.com. Символ %1 означает, что почта будет адресоваться тому же пользователю.
  @b.com %1@smtp.b.com
- Из файла virtusertable создать hash таблицу virtusertable.db.
- Перезапустите почтовый сервер.

Чтобы наш сервер пересылал всю почту, предназначенную для домена b.com на сервер smtp.b.com, необходимо использовать virtusertable.
Почтовый сервер клиента настраиваем точно так же, как и в предыдущей лабораторной работе. То есть, все почтовые ящики пользователей находятся на этом сервере, запросы пользователей он обрабатывает самостоятельно.
Sendmail при отправке почты пытается самостоятельно доставить почту на сервер назначения. По условию задачи нам необходимо сделать так, чтобы он всю исходящую почту пересылал через наш сервер. Для этого в файле mc сервера smtp.b.com необходимо определить так называемый SMART_HOST:

```
define(`SMART_HOST',`smtp:smtp.any.com')dnl
```

Лабораторная работа Б

Использование средства virtusertable.

Задачи	*Описание*

1. Изменение mc файла	1. Перейдите в директорию /etc/mail.
	
	2. Откройте на редактирование файл c20.u20.any.com.mc.
	3. Перед макросом MAILER добавьте следующую строку
	`FEATURE(`virtusertable',`hash -o /etc/mail/virtusertable.db')dnl`
	
	4. Сохраните файл.
	
	5. Перейдите в директорию /etc/mail.
	
	6. В файл virtusertable добавьте следующие строки:
	`virt1@u20.any.com admin@u20.any.com`
	`virt2@u20.any.com root@u20.any.com`
	
	Вместо домена u20 подставьте Ваш домен. Например:
	`virt1@u2.any.com`
	
	7. Сохраните файл.
	
	8. Создание hash таблиц
	`make && make install`
	
	9. Перегрузите почтовый сервер
	`/etc/rc.d/sendmail restart`
	
	10. Проверка работоспособности - отправьте почту любому из пользователей virt в своем домене.

Вопросы

1. Для чего используется средство virtusertable?

2. Какую строку следует добавить в таблицу виртуальных пользователей, чтобы вся почта, предназначенная для домена b.com, попадала в локальный почтовый ящик пользователя b_user?

Дополнительная конфигурация sendmail

Конфигурационные параметры.
Опции безопасности.
База доступа.
Чёрные списки.

В этом разделе будут рассмотрены вопросы настройки sendmail, такие как:

- Конфигурационные параметры.
- Опции безопасности.
- База доступа.
- Черные списки.

Конфигурационные параметры

define(`параметр',`значение')dnl

Конфигурационные параметры sendmail можно задавать в файле mc при помощи директивы define. Например:

```
define(`confPRIVACY_FLAGS',`novrfy,noexpn')dnl
```

У sendmail существует порядка 175 конфигурационный параметров. Мы рассмотрим только те параметры, которые наиболее часто используются.

В таблице при указании параметра отсутствует слово conf, которое обязательно необходимо добавлять к названию параметра при его определении.

Параметр	***Описание***
MIN_FREE_BLOCKS	Параметр определяет количество свободного пространства на диске, измеряемого в блоках. Если места на диске останется меньше указанного, sendmail перестает прини- мать почту. Значение по умолчанию: 100
MAX_MESSAGE_SIZE	Максимальный размер тела письма в байтах, которое sendmail может принять на отправку. Значение по умолчанию: не ограничено.
DELIVERY_MODE	Определяет один из четырех режимов работы sendmail. Значение по умолчанию: background
MAX_HOP	Определяет максимальное количество серверов, через которое может проходить письмо. Значение по умолчанию: 25
LOG_LEVEL	Определяет, насколько подробные сообщения будут отправляться в лог. Файлы. Значение по умолчанию: 9
PRIVACY_FLAGS	Определяет ограничения на применение методов протокола SMTP. Значение по умолчанию: authwarnings

Параметр	*Описание*
COPY_ERRORS_TO	Определяет e-mail, на который будут отсылаться копии сообщений об ошибках. Значение по умолчанию: не определено.
TO_имя	Опции задают различные значения timeout для указанных ситуаций.
TO_IDENT	Время ожидания ответа на запрос об аутентификации отправителя. Если в сети используются только Windows клиенты, рекомендуется этот параметр устанавливать равным 0. Значение по умолчанию: 5s
TO_QUEUEWARN	Определяет, сколько времени письмо будет находиться в очереди на отправление, прежде чем поль- зователю, отправившему его, будет послано предупреждение. Значение по умолчанию: 4h
TO_QUEUERETURN	Определяет, сколько времени письмо может находиться в очереди на отправку. Значение по умолчанию: 5d
QUEUE_LA	Показатель средней загруженности (Load average), при достижении которого все сообщения помещаются в очередь на оправку, а не отправляются немедленно. Значение по умолчанию: 8*чис-ло_процессоров.
MAX_ALIAS_RECURSION	Определяет максимальное количество вложений псевдонимов, ко- торые sendmail может разрешать. Значение по умолчанию: 10
MAX_DAEMON_CHILDREN	Максимальное количество порожденных процессов. Sendmail на каждое соединение по доставке или приему почты порождает новую копию. Значение по умолчанию: не ограничено.
MAX_HEADERS_LENGTH	Максимальная величина заголовка письма. MAX_MESSAGE_SIZE ограничивает только размер письма, размер заголовка он не ограничивает. Значение по умолчанию: 32768
SMTP_LOGIN_MSG	Определяет строку, которая будет выводиться при подключении клиента к почтовому серверу. Значение по умолчанию: $j Sendmail $v/$Z; $b
MAX_RCPTS_PER_MESSAGE	Ограничивает количество получателей одного письма. Значение по умолчанию: не ограничено
DONT_PROBE_INTERFACES	Если этот параметр установлен, sendmail НЕ пытается при старте определить имя сетевых интерфейсов и подставить их в класс w конфигурационного файла. Обычно таким образом определяется имя машины, для которой sendmail будет принимать почту. Значение по умолчанию: false
REJECT_MSG	Определяет сообщение, которое будет выдаваться при отклонении почты. Значение по умолчанию: 550 Access denied
RELAY_MSG	Определяет сообщение, которое будет

Параметр	***Описание***
	выдаваться при запрещении пересылки почты. Значение по умолчанию: 550 Relaying denied

Опции безопасности

define(`confPRIVACY_FLAGS',`...')dnl

При помощи опции PRIVACY_FLAGS можно определить различные опции безопасности, которые определяют:

- Какие сведения о системе можно узнать по протоколу SMTP из внешнего мира.
- Могут ли пользователи просматривать или обрабатывать очередь почтовых сообщений.
- Что требуется от узла на противоположном конце SMTP-соединения.

В таблице приведены некоторые опции безопасности.

Значение	***Описание***
public	Проверка безопасности отменяется. *Не лучший вариант.*
needmailhelo	При подключении к серверу необходимо представиться при помощи команды HELO.
noexpn	Запрещает команду EXPN.
novrfy	Запрещает команду VRFY.
needexpnhelo	Запрещает использование команды EXPN без предварительного использования команды HELO.
needvrfyhelo	Запрещает использование команды VRFY без предварительного использования команды HELO.
noverb	Запрещает «многословный» режим команды EXPN, при котором выдается дополнительная информация о пересылки почты из файлов forward и базы псевдонимов.
restrictmailq	Только пользователи, которые входят в группу, которой принадлежит директория /var/spool/mailq, могут просматривать очередь сообщений.
restrictqrun	Только владелец каталога mqueue может обрабатывать очередь сообщений.
noetrn	Запрещает узлам инициализировать передачу почты при помощи команды ETRN. Обычно этой командой пользуются узлы, подключаемые по коммутируемым каналам.
authwarnings	К сообщениям добавляется заголовок «Authentication Warning». (Установка по умолчанию).
noreceipts	Запрещается выдача кодов состояния доставки.
nobodyreturn	При выдачи кода состояния о доставке не возвращается тело сообщения.
goaway	Отменяет все статусные SMTP запросы: EXPN, VRFY и другие.

При определении параметры безопасности перечисляются через запятую. Например:

```
define(`confPRIVACY_FLAGS', `authwarnings,novrfy,noexpn,restrictqrun')dnl
```

База доступа

```
FEATURE(access_db)
FEATURE(blacklist_recipients)
```

Для включения поддержки базы доступа необходимо использовать средство access_db. Мы его уже определяли, но использовали базу только для разрешения пересылки почты. После определения средства access_db, можно определить дополнительное средство blacklist_recipients, которое позволит использовать дополнительные возможности базы доступа.

База доступа выполнена в виде hash таблицы, состоящей из двух частей: уникального ключа поиска и значения. В поле значение можно использовать следующие ключевые слова:

- RELAY — прием почты на пересылку.
- REJECT — отклонение почты, с выдачей типового сооб- щения об ошибке.
- DISCARD — удаление почты без объяснения причин.
- OK — прием почты в обычном режиме, даже если e-mail попадал под ограничения вводимые правилами REJECT и DISCARD.
- FRIEND — служит для борьбы со спамом. Для его работы необходимо явно описать FEATURE(`delay_checks',`friend').
- HEATER — служит для борьбы со спамом. Для его рабо- ты необходимо явно описать FEATURE(`delay_checks',`heater').
- xxx сообщение — письмо не принимается. Обратно возвращается код ошибки и сообщение. Вместо xxx должен стоять код ошибки, определенный в RFC821.

Пример:

```
@mail.ru 550 мы не принимаем почту с публичных серверов
user@mail.ru   OK
@spamer.com    DISCARD
sex@   REJECT
```

В первой строке запрещается прием любой почты, посланной с mail.ru, с возвращением сообщения «550 мы не принимаем почту с публичных серверов».

Во второй строке делается исключение из предыдущего правила. Почта от пользователя user@mail.ru будет принята, несмотря на то, что он находится в домена mail.ru.

В третьей строке запрещает прием любой почты с домена spamer.com. В ответ сообщение ошибки не отсылается.

В последней строке запрещается прием почты от пользователя sex любого домена. В ответ возвращается стандартное сообщение об ошибке.

В поле ключа можно указывать e-mail с ключевым словом To:, From: и т.д.

Черные списки

FEATURE(dnsbl)
FEATURE(enhdnsbl)

В sendmail можно использовать так называемые черные списки MAPS (Mail Abuse Prevention System), которые публикуются на сайте mail-abuse.org. А так же любые другие списки. Для подключения списков можно использовать следующие средства:

```
FEATURE(dnsbl)
FEATURE(enhdnsbl)
```

Эти средства заставляют sendmail не принимать почту, пришедшую от пользователей или доменов, занесенных в эти списки.
Существует несколько разновидностей списков:

- Содержащих IP адреса известных спамеров.
- Содержащих IP адреса, узлов поддерживающих открытую ретрансляцию.
- Список спамеров, работающих по коммутируемым линиям.

Этот список считается устаревшим и не рекомендуется к дальнейшему применению.

Улучшенная версия enhdnsbl позволяет использовать пять параметров для более точного определения поведения программы в зависимости от результатов поиска.

Все черные списки хранятся в специальных DNS серверах, в которых используются следующие записи:

```
IP.blackholes.mail-abuse.org IN A IP
```

Средство dnsbl можно определять несколько раз для того, чтобы sendmail мог обращаться к нескольким черным спискам. Например:

```
FEATURE(`dnsbl',`blackhole.mail-abuse.org',`Rejectet: see www.mail- abuse.org')dnl
FEATURE(`dnsbl',`relays.mail-abuse.org',`Rejectet: see www.mail- abuse.org')dnl
```

Первый аргумент средства dnsbl определяет список, к кото- рому будет обращаться sendmail. Второй аргумент — строку, которая будет возвращена в качестве ошибки.

На данный момент служба Mail Abuse является платным сервисом. Но существуют и бесплатные черные списки. Их список можно посмотреть на http://www.declude.com/junkmail/support/ip4r.htm

Вопросы

1. Какой конфигурационный параметр позволит ограничить количество процессов sendmail?

2. Какой конфигурационный параметр позволит ограничить объем почтового сообщения?

3. Для чего используется параметр noexpn в опциях безопасности?

4. Какие средства можно использовать для подключения черных списков к sendmail?

Почтовые псевдонимы

Определение псевдонимов.

Загрузка списка рассылки из файла.

Направление почтового сообщения в файл.

Направление почтового сообщения в программу.

Файл ~/.forward

Средство redirect.

Sendmail позволяет определять почтовые псевдонимы, при помощи которых почту, пришедшую в адрес пользователя, можно перенаправить в другой почтовый ящик или по другому адресу.

Не следует путать средство virtusertable и почтовые псевдонимы. Первое позволяет осуществлять пересылки не только почты пользователя, но и учитывать домен пользователя. Почтовые псевдонимы работают только с именами пользователей.

Sendmail позволяет получать базу почтовых псевдонимов из различных источников:

- Локального файла с базой псевдонимов (настройка по умолчанию)
- По протоколу LDAP
- Из служб NIS, NIS+

Мы рассмотрим формирование локального файла почтовых псевдонимов /etc/mail/aliases.db. Поскольку это бинарный файл с базой данных, сначала создается текстовый файл aliases, а затем он преобразуется в файл aliases.db.

Для создания базы данных почтовых псевдонимов используют программу newaliases.

Определение псевдонимов

director: boss
director: boss,spy@u20.any.com

Предположим, что в системе есть учетная запись boss, принадлежащая директору компании. Директор попросил создать дополнительный e-mail: director.
Можно добавить еще одну учетную запись director. Настроить почтовую программу на работу с двумя почтовыми ящиками. Что достаточно неудобно. А можно использовать почтовый псевдоним для этого в файл /etc/mail/aliases нужно добавить следующую строку:

```
director:           boss
```

Она заставит sendmail всю почту, пришедшую пользователю director, поместить в почтовый ящик пользователя boss. В этом случае не надо заводить учетную запись пользователя director, т.к. sendmail сначала смотрит содержимое базы данных почтовых псевдонимов и только потом обращается к системе аутентификации FreeBSD.

В качестве пункта назначения можно указать не только локальный почтовый ящик, но и e-mail.

```
director:           boss@u20.any.com
```

А так же список адресов назначения, разделенный запятыми:

```
director:      boss,spy@u20.any.com
```

В этом случае письмо будет помещено в локальный почтовый ящик пользователя director и отправлено по e-mail: spy@u20.any.com.

Перечисляя список адресатов, можно создать простой список рассылки. Обычно почту, предназначенную пользователю root, перенаправляют реальному пользователю системы.

Загрузка списка рассылки из файла

list: :include:/путь/к/файлу

Для организации списка рассылки можно перечислять всех адресатов сразу после имени псевдонима. Но вести этот список становится неудобно, когда количество адресатов очень большое.

Вы можете перечислить все e-mail по одному на строку, в отдельном файле, а сам файл подключить при помощи директивы «:include:».

```
list:   :include:/путь/к/файлу
```

Файл со списком адресатов должен находиться на локальных дисках, а не на сетевом диске, подключенном по NFS. Если файловая система смонтирована с параметром hard и сервер NFS выйдет из строя, программа sendmail зависнет.

Направление почты в файл

user: /путь/к/файлу

Если в качестве объекта, на который ссылается псевдоним, указать абсолютный путь к файлу, почта будет добавляться в конец этого файла.

```
user:   /путь/к/файлу
```

Файл, куда будут попадать почтовые сообщения, должен принадлежать пользователю, определяемому при помощи параметра DefaultUser.

Направление почты в программу

manager: | /путь/к/программе

Очень интересной особенностью базы данных почтовых псевдонимов является возможность направления почты на стандартный ввод программы.

```
manager:        |     /путь/к/программе
```

Обычно при помощи этой особенности реализуются менеджеры списка рассылки. Например, если послать письмо в поле Subject которого стоит ключевое слово Subscribe, на такой псевдоним программа может добавить e-mail отправителя письма в список рас- сылки. Если же в Subject стоит Unsubscribe — удалить e-mail из списка рассылки.

Программа будет выполняться от имени пользователя, с правами которого работает sendmail. Использование перенаправления почты в программу — это потенциальная возможность взломать систему. Поэтому пользоваться перенаправлением следует очень аккуратно.

Файл ~/.forward

~/.forward

Все изменения в файле aliases может делать только суперпользователь. Но в системе есть возможность обыкновенным пользователям осуществлять пересылку почты. Им достаточно в файле ~/.forward написать e-mail, куда необходимо пересылать почту.

В этом файле работают все механизмы, которые можно использовать в файле aliases. Для их записи не надо писать имя почтового псевдонима.

Пример файла .forward:

```
boss@any.com
spy@any.com, /home/user/new_mail_box
```

Согласно информации, находящейся в этом фале, почта, приходящая текущему пользователю, будет пересылаться по адресам boss@any.com и spy@any.com, а так же добавляться в конец файла new_mail_box.

Средство redirect

FEATURE(redirect)

Когда сотрудник увольняется из организации, администратор должен удалить его учетную запись. При этом людям, приславшим почту на старый e-mail сотрудника, будет возвращаться сообщение об ошибке.

По просьбе сотрудника Вы можете либо перенаправлять приходящую почту на новый e-mail, либо возвращать сообщение, содержащее новый e-mail, человеку, пославшему письмо.

Для того, чтобы обратно возвращался e-mail, а не сообщение об ошибке, необходимо включить средство redirect.

```
FEATURE(redirect)dnl
```

А в псевдонимах написать следующую строку:

```
user:   user@new.com.REDIRECT
```

Вопросы

1. Каким образом можно перенаправить почту, пришедшую пользователю user, в почтовый ящик пользователя any?

2. Напишите строку, которая позволит загрузить список адресов, куда будет направлено письмо из внешнего файла?

3. Какой файл могут использовать пользователи для самостоятельно определения перенаправления почты?