ALT Linux Master 2.0

Руководство системного администратора

А. Блохин, А. Боковой, И. Вергейчик, О. Власенко, К. Волков, С. Волков, М. Забалуев, Ю. Зотов, С. Иевлев, Д. Левин, И. Муратов, А. Новодворский, А. Орлов, А. Сергеев

ALT Linux Master 2.0: Руководство системного администратора

А. Блохин, А. Боковой, И. Вергейчик, О. Власенко, К. Волков, С. Волков, М. Забалуев, Ю. Зотов, С. Иевлев, Д. Левин, И. Муратов, А. Новодворский, А. Орлов, А. Сергеев
Большую работу по редактированию книги осуществили: А. Астафьев
А. Прокудин
М. Шигорин

Настоящая книга составлена из документов, распространяющихся под различными лицензиями. Права на материал раздела “Антивирусы” принадлежат Лаборатории Касперского и Диалог-Наука (И.Данилов). Остальная часть книги распространяется на условиях GNU Free Documentation License, версия 1.1.

Каждый имеет право воспроизводить, распространять и/или вносить изменения в настоящий Документ (кроме раздела “Антивирусы”) в соответствии с условиями этой лицензий

Данный Документ не содержит Неизменяемых разделов; Данный Документ не содержит текста, помещаемого на первой или последней страницах обложки.

Содержание

Список примеров

Часть I. Оборудование

Глава 1. Общая информация

Linux поддерживает практически все современное оборудование для архитектуры x86, за исключением специально ориентированного на ОС Windows (например, так называемые winmodem и winprinter), а также продукцию тех производителей, которые по тем или иным причинам не желают давать спецификации на устройства для написания драйверов.

Информация, предоставленная в этом руководстве, не претендует на полноту описания, поэтому, если вы не найдете здесь ответа на интересующий вас вопрос, прежде чем писать в список рассылки ALT Linux, рекомендуется посмотреть следующую документацию:

1. документация к ядру (пакет `kernel-doc`[1]);
2. Linux FAQ и HOWTO можно найти в как в Internet, так и в дистрибутиве;
3. поиск в Internet по спискам конференций;
4. исходные коды :-) – это для тех, кто способен в них разобраться.

С точки зрения системного администратора, задачей которого является настройка оборудования и проверка его работоспособности для Linux, устройства в первую очередь определяются своим типом, производителем, затем способом подключения.

Для настройки устройств в дистрибутиве ALT Linux Master существуют следующие утилиты для настройки (объединенные в DrakConf):

Утилиты для настройки оборудования

- PCI-, AGP- и USB-устройств – **kudzu**. При этом рекомендуется, чтобы сервис **kudzu** загружался автоматически при загрузке системы – в этом случае будут сконфигурированы все устройства, добавленные или удаленные с момента последней перезагрузки системы;
- звуковых карт (преимущественно ISA) – утилита **sndconfig;**
- графической карты и оболочки XFree86 – **Xfdrake;**
- мыши – **mousedrake;**
- клавиатуры – **keyboarddrake;**
- принтеров – **printerdrake;**
- сети – **draknet**.

[1] Под пакетом `kernel-doc` здесь и далее подразуемевается либо `kernel22-doc`, либо `kernel24-doc` в зависимости от того, какое ядро у вас установлено.

На сегодняшний день наиболее распространенными способами расширения конфигурации компьютера являются *шины* PCI, AGP, ISA[2], а для подключения внешнего оборудования – USB, PCMCIA, SCSI и *порты* COM (последовательные) и LPT (параллельный).

Проще всего под Linux проверяется работоспособность оборудования, использующего шину PCI: достаточно набрать команду **/sbin/lspci**, чтобы увидеть информацию обо всех подключенных PCI-устройствах. Команда **lspcidrake** в дополнение к выводу команды **/sbin/lspci** выводит информацию о наличии драйверов (модулей ядра) для них.

Это возможно потому, что каждое PCI- или AGP-устройство содержит пару уникальных *идентификационных номеров* (называемых PCI ID), в которой первым числом определяется производитель устройства, а вторым – само устройство. В дистрибутиве присутствует пакет `ldetect-lst`, который содержит информацию о наличии (или отсутствии) драйверов для каждого известного на момент создания таблицы (`/usr/share/ldetect-lst/pcitable`) PCI-устройства; если обнаружено изменение конфигурации и устройству сопоставлен драйвер, настройка производится автоматически утилитой **kudzu** (а изначально – программой установки системы).

Основные проблемы возникают в случае, когда для вашего устройства нет драйвера или неизвестны идентификационные номера устройства и его нет в таблице. В этом случае рекомендуется произвести ручную настройку устройства или написать в список рассылки по дистрибутиву. При возникновении проблем с PCI-устройством настоятельно рекомендуется выслать следующую информацию о нем:

1. название, производитель, надписи на самых больших чипах и т.д.;
2. вывод команд **lspcidrake** и **/sbin/spci -vv**;
3. содержимое файла `/proc/bus/pci/devices`;
4. описание проблемы.

Шины USB и PCMCIA

Для поддержки “горячего” подключения устройств, разработанных для шин USB и PCMCIA, в дистрибутиве ALT Linux Master существует специальная программа **hotplug**, задача которой заключается в автоматической загрузке драйверов. Эта программа входит в одноименный пакет, который устанавливается по умолчанию.

При возникновении проблем с устройствами USB необходимо найти информацию о вашем устройстве в файле `/proc/bus/usb/devices`. Информация в этом файле содержит много технической информации, для ее “отсеивания” можно воспользоваться утилитами типа **usbview** – их вывод будет более понятен начинающему пользователю. Если ни один драйвер не “подхватил” его – скорее всего, это устройство не поддерживается. Для получения помощи можно обратиться в список рассылки ALT Linux, при этом настоятельно рекомендуется выслать содержимое файла `/proc/bus/usb/devices`.

Получить информацию о поддержке USB можно на сайте http://www.linux-usb.org/.

Шина ISA

[2]Шина ISA, равно как и COM/LPT-порты, ныне относится к разряду “наследственных”.

Для шины ISA есть следующие варианты: если устройство соответствует стандарту ISA Plug'n'Play, настройку аппаратных ресурсов можно проводить через программу **isapnp**. В ином случае потребуется сконфигурировать плату либо перемычками на ней (например, звуковую), либо утилитой, которую обычно прилагают на дискете с драйверами (большинство сетевых карт). В любом случае все эти параметры придется указать вручную драйверу устройства для его работы. К счастью, устройства ISA уже менее распространены.

Устройства, присоединяемые через параллельный, последовательный или игровой порты

Что касается оборудования для последовательных и параллельных портов, а также джойстиков, то практически в каждом случае необходимо вручную настраивать драйвер соответствующего устройства. Исключение здесь составляют только внешние модемы с последовательным интерфейсом, которые не требуют драйверов.

Настройка таких устройств (за исключением принтеров) практически всегда производится вручную – например, для настройки модема необходимо указать COM-порт, к которому он подключен. Для настройки джойстика необходимо найти драйвер под него и вручную настроить его посредством редактирования конфигурационных файлов.

Рассмотрим теперь варианты настройки различных типов устройств.

Материнские платы и процессоры

ALT Linux Master поддерживает все современные 32-битные процессоры архитектуры *x86*, начиная с Intel Pentium и совместимых; если процессор исправен и хорошо охлаждается – с ним не должно возникнуть никаких проблем. Процессоры, работающие в нештатном режиме, использовать не рекомендуется[3].

Для проверки работоспособности процессора при критических нагрузках рекомендуется запустить в одном сеансе вариант программы **burn** (из пакета `cpuburn`) – например, **burnP6** для Intel Pentium i686 или AMD Athlon, а в другом – компиляцию какого-нибудь большого пакета, гарантированно собирающегося. Обычно при наличии проблем с охлаждением система сразу не зависает, но компиляция останавливается из-за ошибок.

Последние также могут возникать из-за некачественной (или нестабильно работающей) памяти – для ее проверки предназначен пакет **memtest86**, который добавляет в меню загрузки системы еще один вариант.

Специальную настройку материнских плат производить обычно не требуется – за исключением редких случаев, все работает с настройками по умолчанию (см. стр. !!!!).

При настройке BIOS стоит обратить внимание на следующие параметры:

1. Параметр Use PNP OS (как вариант – PNP OS installed) – включение этого параметра – ON (или ENABLE) приводит к тому, что BIOS перестает настраивать устройства PnP, доверяя это операционной системе. Для Linux выключение этого параметра – NO (или DISABLE) может помочь с инициализацией некоторых устройств.

[3]Однако при известной аккуратности это возможно.

2. На материнских платах с чипсетами семейства VIA (KT133, 133A, 266, 333) рекомендуется выключить параметры Passive Release и Burst Read/Write[4], которые в некоторых случаях также могут служить причиной зависаний и неполадок.

3. Если на материнской плате присутствует видеокарта AGP, рекомендуется выставить параметр AGP Aperture Size не меньше 64 мегабайт, в том случае, если объем оперативной памяти компьютера не менее 128M., В том случае, если обьем оперативной памяти менее 128M, то не более половины установленной оперативной памяти (т.е. при наличии 64 мегабайт установите значение этого параметра равным 32).

Достаточно часто возникают проблемы из-за ошибок в BIOS. Поэтому, если вы столкнулись с какой-либо странной проблемой (например, не работает заведомо поддерживаемая видеоплата) – рекомендуется посмотреть на сайте производителя материнской платы новые версии BIOS и, если в списке изменений присутствует ваша проблема – обновить BIOS. Например, при тестировании материнской платы Asus A7N266-E (на базе чипсета nForce 420D) было обнаружено, что встроенный контроллер USB не работает одновременно с загруженным модулем apm. Проблема решилась обновлением BIOS'а с версии 1001A до 1001D.

Клавиатура

С точки зрения поддержки клавиатур в Linux они отличаются по способу подключения (USB и обычные PS/2 или DIN), а также по количеству клавиш (101, 102, 104 ...).

Обычные клавиатуры настраиваются автоматически, причем дополнительные (т.н. Windows-клавиши) автоматически задействуются как в консоли, так и в X Window. Единственное, что необходимо сделать – указать раскладку клавиатуры при установке системы либо позже при помощи **keyboarddrake**.

Клавиатуры USB также определяются автоматически; единственное, что требуется для их правильной работы – это настроенный интерфейс USB и установленный пакет `hotplug`. Настройка раскладки делается точно также, как и для обычных клавиатур.

Мышь

Мыши различаются прежде всего по способу подключения: USB, PS/2, COM и BusMouse (сейчас в основном распространены две первые модификации), а также количеством кнопок и наличием колеса прокрутки.

Так как в консоли и в X Window предусмотрена поддержка третьей кнопки (с ее помощью реализуется функция вставки), рекомендуется использовать трехкнопочные мыши; при наличии двухкнопочной мыши третья кнопка может эмулироваться одновременным нажатием обеих имеющихся.

Настройка мыши производится в процессе установки, а после нее – при помощи утилиты **mousedrake**. В настройках этой программы надо выбрать следующее: тип мыши по подключению, протокол ее работы (для мышей PS/2 и COM), а также включение эмуляции третьей кнопки.

Рассмотрим поподробнее протоколы мышей:

Протоколы работы мышей

- USB – здесь есть всего два варианта настройки: обычная мышь или мышь с колесом. Соответственно, достаточно взглянуть на свою мышь, чтобы сделать выбор.

[4]Или обновить BIOS.

- PS/2 – вариантов уже больше:
  - обычная двух- или трехкнопочная мышь – выберите Generic;
  - Logitech MouseMan+ или GlidePoint (встречаются редко) – выберите соответствующую;
  - мышь с колесом – надо выбрать один из следующих вариантов (по производителю):
    - производство Genius – посмотрите на ее название (обычно написано на дне мыши) и выберите Genius Netmouse или Genius Netscroll – хотя бывают случаи, когда на мыши написано NetScroll, а работает она по протоколу NetMouse, поэтому в случае неработоспособности мыши стоит попробовать оба протокола. Мышь Netscroll+ также иногда работоспособна при выборе протокола Logitech MouseMan+;
    - Microsoft, Logitech или Mitsumi, а также другая мышь с колесом – стоит попробовать вариант Generic PS/2 Wheel mouse;
    - если мышь все же не заработает – остается выбрать вариант Generic (колесо, естественно, при этом работать не будет);
- COM – очень много вариантов, но большинство из них предназначены для специфических и малораспространенных мышей вроде Kensington. Для обычных мышей есть следующие варианты выбора:
  - двухкнопочная – выбирайте 2 button mouse;
  - трехкнопочная – это либо 3 button mouse, либо MouseSystems;
  - мышь с колесом – выбирайте по производителю (как и в варианте PS/2, для безымянных мышей скорее всего подойдет протокол Microsoft IntelliMouse).

Устройства хранения данных

Жесткие диски

Современные жесткие диски производятся со следующими интерфейсами: IDE, SCSI и USB (в основном это Flash-карты, подключенные к системе через Flash-Reader).

Жесткие диски IDE определяются системой автоматически в процессе загрузки; доступ к ним (и другим устройствам на этой шине) производится посредством специальных *файлов блочных устройств* (`/dev/hdXN`[5]).

Имя устройства формируется следующим образом:

- hda – primary master;

[5] В описании файла блочного устройства `X` означает латинскую букву, а `N` – число.

- hdb – primary slave;
- hdc – secondary master и т.д.

При этом обращение к файлу устройства подразумевает доступ ко всему диску целиком. Обращение к разделам на диске производится через устройства `/dev/hdXN`, где `/dev/hda1` – первый *основной* раздел (primary partition) на первом диске, `/dev/hda2` – второй основной раздел. Так как основных разделов может быть не более четырех, то расширенные разделы начинаются с номера 5: `/dev/hda5` – первый *логический раздел* (logical partition) в *расширенном разделе* (extended partition) на первом диске.

Протокол обмена данными с жесткими дисками IDE для всех современных чипсетов выбирается автоматически при загрузке ядра. Для более тонкой ручной настройки IDE-устройств в дистрибутиве присутствует команда **hdparm**, с помощью которой можно управлять протоколом доступа (т.е. UDMA100, UDMA33, PIO16 и т.д.), а также некоторыми другими параметрами. Подробнее смотрите **man hdparm**.

Важно

Пользоваться программой **hdparm** рекомендуется исключительно осторожно, т.к. установкой неправильных настроек можно добиться потери информации, а в худшем случае – и неисправности жесткого диска. Настройки **hdparm** можно сохранить в файлах конфигурации в каталоге `/etc/sysconfig/harddisk` (в файлах с именами hdX – для каждого устройства, в том числе и Cdrom/DVD) – тогда они будут применяться автоматически в процессе загрузки системы.

Жесткие диски SCSI также определяются системой автоматически в процессе загрузки ядра. Единственное отличие от IDE для пользователя – то, что устройства называются не `/dev/hdXN`, а `/dev/sdXN`.

Носители данных USB определяются системой автоматически в момент физического их подключения при установленном пакете `hotplug`. Далее все зависит от наличия/отсутствия поддержки конкретного устройства USB в системе – если таковая присутствует, доступ к данным можно получить через интерфейс SCSI (например, как `/dev/sda` при условии незанятости этого имени другими SCSI-устройствами, в противном случае выбирается первое свободное имя).

Устройства CD-ROM (CD-RW)

IDE CD-ROM автоматически определяются системой и в процессе установки для них создаются специальные ссылки в каталоге `/dev` – т.е. `/dev/cdrom` для первого привода, `/dev/cdrom2` – для второго и т.д. Также доступ к устройству можно получить через интерфейсы `/dev/hdX` для IDE CD-ROM и `/dev/scdX` – для SCSI. Как и для всех устройств со съемными носителями, при включении сервиса autofs монтирование и размонтирование их происходит автоматически при попытке прочтения данных из каталога, куда должнен быть смонтирован носитель – обычно это `/mnt/cdrom`.

С помощью параметра -E команды **hdparm** для некоторых приводов CD-ROM можно регулировать скорость вращения их шпинделя (см. тж. **man hdparm**).

Чуть сложнее обстоит дело с настройкой устройств с функцией записи (перезаписи) дисков (т.е. CD-R/RW). Поскольку эта функциональность реализуется посредством эмуляции SCSI-интерфейса, необходимо включить таковую; это осуществляется автоматически в процессе установки системы при обнаружении такого привода. Для ручного добавления необходимо вставить в файл `/etc/modules` строку `scsi_hostadapter`, а в файл `/etc/modules.conf` – `options ide-scsi units=hdX`, где `hdX` соответствует подключению CD-R/RW (например, `hdc` для мастера на втором контроллере). Можно также создать символическую ссылку вида `/dev/cdromN`, указывающую на `/dev/scd0` (если

нет других SCSI CD-ROM) для большего удобства. В итоге записывающий привод станет доступен не как устройство `/dev/hdX`, а как устройство `/dev/scdN`. Это относится к любым IDE-устройствам, но необходимо только для CD-R/RW, так как утилита **cdrecord** может работать только через SCSI-интерфейс.

Сменные устройства типа ZIP

Определяются ядром автоматически в процессе загрузки (если они IDE или SCSI), во время подключения (USB) и вручную при подключении через параллельный порт (для настройки подобный устройств см. `paride.txt` из пакета `kernel-doc`, который находится в каталоге `/usr/share/doc/kernel`).

Единственный нюанс заключается в том, что обычно FAT на ZIP-дисках располагается на четвертом разделе (`/dev/hdX4`).

Флоппи-дисководы

Определяются автоматически в процессе загрузки системы. Для произведения расширенного конфигурирования (например, для форматирования дискет на нестандартную емкость) смотрите файл `floppy.txt` из пакета `kernel-doc`, а также документацию из пакета `fdutils`.

Видеокарты

Видеокарты с точки зрения драйверов системы X Window (являющейся в виде XFree86 основой графической подсистемы в большинстве дистрибутивов Linux) отличаются в основном типом используемого чипа; если производитель карты не производил “коррекции” его работы, один и тот же драйвер может использоваться с различными продуктами, использующими один и тот же графический процессор.

Настройка производится через утилиту **XFdrake**, которая автоматически запускается в процессе установки дистрибутива и может быть запущена вручную после установки. Как и большинство утилит настройки, **XFdrake** имеет эксперт-режим (ключ **--expert**), в котором можно вручную настроить большее количество параметров.

В дистрибутив ALT Linux Master включены две версии XFree86 – 3.3.6 и 4.x.x. Версия 3.3.6 используется для поддержки устаревших видеоплат, драйверы для которых отсутствуют в четвертой версии. Однако для некоторых видеоплат есть драйвера в обеих версиях XFree86. В этом случае при настройке платы в экспертном режиме появляется возможность выбора версии; в общем случае рекомендуется использовать 4.x.x, однако при наблюдении нестабильной работы можно откатиться на ветку 3.3.6.

Как уже было написано раньше, PCI- и AGP-видеоплаты в большинстве случаев настраиваются автоматически; если этого не произошло, можно попробовать указать тип чипа вручную, выбрав его из списка. Также в подобных случаях рекомендуется прочитать документацию о устройствах PCI в этом же разделе.

Если ваша плата определилась правильно и на экране появилась тестовое изображение – то все нормально и на этом рекомендуется остановиться. Опытные пользователи могут произвести более тонкую настройку видеоплаты – например, для некоторых видеоплат можно вручную выставить параметры в конфигурационном файле XFree86 – обычно это `/etc/X11/XF86Config` (`XF86Config-4` для 4.x.x). Документацию о них можно получить в описаниях из `/usr/X11R6/lib/doc`, а также (значительно более свежую) в дереве исходных текстов проекта XFree86.

Для Matrox существует дополнительный драйвер с закрытым исходным кодом, написанный программистами Matrox, который включает в себя улучшенную поддержку различной функциональности этих плат; для его установки необходимо скачать пакет `XFree86-4.x.x-altx-mga_hal` с нашего FTP-сервера (ftp://ftp.altlinux.ru) и установить его. Дополнительно изменять файл конфигурации не требуется.

Аппаратное ускорение 3D-графики в XFree86

В дистрибутиве ALT Linux Master включена поддержка аппаратного 3D-ускорения для некоторых видеоадаптеров. В XFree86 версии 4.x.x входит код из проекта DRI (http://dri.sourceforge.net), для XFree86-3.3.6 специально скомпилирован модуль GLX из проекта Utah-GLX.

В любом случае использование аппаратного 3D ускорения рекомендуется только в XFree86-4.x.x, использование XFree86-3.3.6 с аппаратным 3D ускорением может привести к нестабильности в работе. Поскольку 3D-ускорение в Linux пока еще находится в состоянии разработки, по умолчанию его включение производится только для наиболее стабильных драйверов.

В версии XFree86-3.3.6 поддерживаются следующие 3D акселераторы:

- Intel i810/i815 (экспериментальный)
- ATI Mach64
- Matrox G200/G400
- S3 Virge/S3 Savage 3D (экспериментальный)
- nVidia Riva (экспериментальный)
- SiS 6326 (экспериментальный)

Из этого списка достаточной стабильностью и производительностью отличается только драйвер для Matrox. Остальные драйверы являются экспериментальными.

В версии XFree86-4.x.x поддерживаются следующие 3D акселераторы:

- 3DFX Voodoo (от Banshee до Voodoo 5)
- ATI Rage 128 (как PCI, так и AGP-вариантов)
- ATI Radeon (кроме 8500)
- Matrox (от G200 до G550 и только AGP)
- Intel i810/i815/i830
- 3D Labs Oxygen GMX2000 (экспериментальный)
- SiS 300/630/530 (экспериментальный)

Здесь по умолчанию настраивается 3D-ускорение для всех стабильных драйверов. Экспериментальные драйверы, как и для XFree86-3.3.6, можно настроить, запустив утилиту **XFdrake** в режиме эксперта. Если проявляются проблемы при использовании 3D, лучше всего либо его отключить (настоятельно рекомендуется, если вам оно жизненно не нужно), либо обратиться к нам за поддержкой – скорее всего проблема уже будет решена в новой версии XFree86.

Для некоторых других видеокарт (например, на чипе Kyro II) закрытые драйверы выпущены производителями и доступны на соответствующих сайтах.

Видеокарты nVidia

Для видеоплат на чипах nVidia существует два драйвера под Linux. Один из них (свободный, входящий в XFree86) достаточно простой и не поддерживает множество функций (например аппаратное 3D, а также несколько других расширений). Другой является закрытым (коммерческий, исходный код недоступен) и написан программистами nVidia. Для его установки в режиме эксперта необходимо запустить **Xfdrake** и выбрать пункт Xfree86 4.x.x с аппаратным 3D ускорением. В других режимах конфигурация будет автоматически настроена с использованием этого драйвера; для возврата к стандартному драйверу XFree86 используйте режим эксперта.

Важно

не рекомендуется собирать этот драйвер самостоятельно, при выходе его новой версии лучшим решением будет обновление драйвера вместе с ядром дистрибутива из раздела updates. Кроме этого, компания ALT Linux не несет ответственности за качество этого драйвера и не осуществляет его поддержку – используйте на свой страх и риск.

Настройка монитора

По умолчанию утилита **XFdrake** настраивает монитор автоматически, что в большинстве случаев является приемлемым. В то же время опытные пользователи в экспертном режиме могут вручную изменить настройки разрешения и глубины цвета для каждой пары монитор-видеоплата. Помните, что аппаратное ускорение 3D работает только в 16- и 32-х битной глубине цвета. Рекомендуется (если это возможно) устанавливать глубину цвета 16 бит (как это делается в большинстве случаев по умолчанию).

Для получения качественного изображения на экране рекомендуются следующие настройки видеорежимов (помните, что рекомендуется работать при частоте обновления экрана не ниже 85 Гц):

- 14" монитор – 640x480 или 800x600
- 15" монитор – 800x600 или 1024x768
- 17" монитор – 1024x768 или 1152x864
- 19" монитор – 1280x1024 или 1600x1200
- 21" монитор – 1600x1200 или выше.

При прочих равных, лучше выбирать меньшее разрешение, так как в этом случае кадровая частота обновления экрана будет выше; в то же время минимальным практически пригодным для работы является режим 800x600, а более комфортным – 1024x768 и выше.

Профессионалы также могут вручную настроить специальные параметры видеорежима – например, положение на экране, частоту обновления кадров, нестандартное разрешение (у одного из авторов на 14" мониторе используется разрешение 928x696) и т.д. Это проще всего сделать с помощью утилиты **videogen**, вручную занеся выданные этой утилиты результаты в файл настроек XFree86. Подробную документацию можно получить из соответствующего пакета (каталог `/usr/share/doc/videogen-*`), а также из xfaq (http://www.linux.org.ru/books/xfaq.html).

Звуковые карты

ALT Linux Master поддерживает большинство современных звуковых карт. Проще всего настраиваются PCI-карты – это происходит автоматически с помощью программы **kudzu**.

Звуковые карты с интерфейсом ISA можно настроить с помощью утилиты **sndconfig** или вручную.

Сейчас существует два различных проекта для поддержки звука в Linux – это достаточно старый, но в то же время распространенный стандарт OSS (драйверы для карт в этом стандарте входят в ядро Linux), а также новый улучшенный стандарт ALSA (эти драйверы входят в дополнительные пакеты `alsa-*`) для всех ядер, входящих в дистрибутив. По умолчанию в режиме автоматической настройки выбирается наилучший драйвер для каждой карты, но опытные пользователи могут попробовать как OSS, так и ALSA. Единственное, что необходимо помнить – это при использовании драйверов ALSA в файл `/etc/modules.conf` необходимо добавить строку `prereq snd-ваш_драйвер snd-pcm-oss` для включения эмуляции OSS драйверами ALSA.

Кроме того, для плат на основе чипа EMU10K1 (Creative SB Live! и Audigy) существует пакет `emu10k1-tools` с утилитами, при помощи которых опытные пользователи могут загружать микрокод для поддержки некоторых дополнительных функций.

Сетевые платы

Дистрибутив ALT Linux Master поддерживает большинство современных сетевых плат с подключением через ISA, PCI, PCMCIA и USB-интерфейсы. Все адаптеры, за исключением адаптеров для шины ISA, не требуют специальной настройки и оперделяются дистрибутивом автоматически.

Исключение составляют адаптеры фирмы Intel (серии EtherExpress100), для которых существует два драйвера – `eepro100` (написанный сообществом Linux) и `e100`, написанный фирмой Intel. В случае возникновения проблем рекомендуется попробовать драйвер, отличный от уже настроенного у вас в системе.

Такая же ситуация существует и с драйверами `3c59x` и `3c90x` соответственно для плат 3COM.

Для драйвера `tulip` существует его более старая (и, возможно, более стабильная версия) под названием `tulip_old`. При настройке сетевых плат с интерфейсом ISA, скорее всего, придется указать параметры для модуля – I/O-порт и IRQ, используемое вашей сетевой платой. При успешной загрузке драйвера в сообщениях ядра (**dmesg**) появится запись об успешной настройке сетевой платы. Если в системе установлены две одинаковые сетевые карты, для их настройки достаточно загрузить один драйвер – он будет обслуживать оба устройства. В случае наличия в системе разных сетевых плат они будут именоваться по порядку загрузки драйверов, т.е. первая – `eth0`, вторая – `eth1` и т.д.

Радио- и видеотюнеры

В ALT Linux Master входят драйвера для различных плат, поддерживающих функции радио- и видеотюнеров. Одними из наиболее популярных на сегодняшний день являются видеотюнеры, основанные на чипах Brooktree (BT848, 878 и т.д.); эти платы определяются и настраиваются автоматически, но в некоторых случаях необходимо произвести ручную более тонкую настройку платы. Как это сделать – описано в документации на драйвер bttv (`/usr/share/kernel*-doc*/video4linux/bttv/*`).

С настройкой радиотюнеров дело обстоит сложнее, т.к. они обычно выполнены для шины ISA – необходимо вручную определить подходящий драйвер для вашего тюнера (доступные драйвера лежат в каталоге `/lib/modules/kernel-_версия_ядра_/drivers/media/radio/*`) и добавить в файл `/etc/modules.conf` строку вида alias char-major-81-64 _нужный_драйвер_). Например, для платы Sound Forge с чипом SF16-FMR2 настройка выглядит так:

```
alias char-major-81-64 radio-sf16fmx2
options radio-sf16fmx2 io=0x284
```

Управление радиотюнером осуществляется любой программой, соответствующей стандарту *video4linux* (например, **qdt** или **radio** из пакета xawtv-radio); управление видеотюнером производится через программы **xawtv** или **kwintv**.

Прочее обрудование

Наладонные компьютеры (на основе PalmOS или WinCE)

Для систем на основе WinCE не существует средств для синхронизации их с Linux, поэтому для них (как и для Psion) единственным способом обмена данными является перенос данных через Flash-карты или через сеть (или нуль-модемный кабель).Для систем на основе PalmOS существует достаточно много утилит для синхронизации, установки новых программ и т.д. – утилиты нижнего уровня из пакета `pilot-link`, аналог Palm Desktop – программа **jpilot** и т.д.

Проблемы могут возникнуть, если Palm соединяется с компьютером через интерфейс USB (Visor или Palm m500) – но обычно все работает.

Дополнительную информацию можно получить из Palm-HOWTO.

Инфракрасные порты

Linux поддерживает множество инфракрасных портов – в том числе высокоскоростные стандартов MIR и HIR; программное обеспечение содержится в пакете `irda-utils`. Информацию по этой теме можно получить в Infrared-HOWTO.

Стриммеры

В дистрибутиве присутствует поддержка различных стриммеров (ленточных накопителей) – в основном это SCSI- и IDE-модели. За дополнительной информацией обращайтесь в список рассылки ALT Linux или к содержимому пакета `kernel-doc`.

Сканеры

К сожалению, с поддержкой сканеров в Linux дело обстоит не лучшим образом; тем не менее, в состав дистрибутива ALT Linux Master входит система `sane`, поддерживающая устройства, подключаемые через интерфейс SCSI или параллельный порт. Также поддерживаются некоторые USB-сканеры, для функционирования которых должна быть запущена программа

hotplug. Поскольку список поддерживаемых сканеров достаточно мал, перед приобретением сканера настоятельно рекомендуется ознакомиться с документацией из пакета `sane` или на сайте http://www.mostang.com/sane/.

Цифровые камеры, mp3-плейеры и прочие дополнительные устройства

В отличие от сканеров, цифровые камеры поддерживаются неплохо; обмен изображениями осуществляется при помощи программ **gphoto** и **gphoto2**. В документации к ним находится список поддерживаемых моделей (более 100).

Также поддерживаются некоторые mp3-плейеры на основе Flash-карт и жестких дисков (с mp3-CD-плейерами, понятное дело, проблем не возникает).

Ссылки

Для получения информации обращайтесь в список рассылки ALT Linux или поищите информацию в Internet:

1. устройства с интерфейсом USB [http://www.linux-usb.org];
2. видеоплаты на чипах nVidia Riva TNT и более поздних [http://www.nvidia.com];
3. звуковые платы Aureal [http://aureal.sourceforge.net];
4. Win-модемы на некоторых чипах (Lucent, 3COM, PCTel) – см. сайты производителей и http://www.linmodems.org;

Часть II. Настройка системы

Глава 2. Файловые системы

Разновидности файловых систем в дистрибутиве ALT Linux Master

В дистрибутиве ALT Linux Master поддерживаются следующие файловые системы:

1. ext2;
2. ext3 (только для ядра 2.4.x);
3. reiserfs (3.5 – для ядра 2.2.x, 3.5 и 3.6 для ядра 2.4.x);
4. xfs (только для ядра 2.4.x);
5. jfs (только для ядра 2.4.x);
6. vfat;
7. ntfs (только чтение);
8. isofs;
9. udf;
10. другие (менее распространенные).

Файловые системы 2–5 являются журналируемыми.

Дистрибутив ALT Linux Master может быть установлен на любую из первых трех систем; при выборе рекомендуется иметь в виду следующие соображения:

ext2 является самой “заслуженной” и обкатанной из этих ФС; она весьма стабильна, но не является журналируемой;

ext3 логическое продолжение ext2 в сторону журналируемости; хорошая совместимость с ext2 (легкое взаимопревращение);

reiserfs журналируемая система, особо оптимизированная под каталоги, содержащие большое количество файлов, а также под небольшие файлы. Для использования в данный момент рекомендуется версия 3.6 для ядер 2.4.x;

Следующие две системы являются экспериментальными. Вохможно они будут в списке доступных для установки при условии достаточной стабильности на момент выпуска дистрибутива.

xfs журналируемая ФС, оптимизированная для хранения больших объемов информации и хорошей масштабируемости; рекомендуется совместно с samba при необходимости иметь ACL (Access Control Lists);

jfs в данный момент для хранения важных данных не рекомендуется вследствие активной доработки.

Файловые системы isofs и udf используются в носителях CD/DVD-ROM; vfat и ntfs используются семейством ОС Microsoft.

Работа с файловыми системами

Утилиты для работы с файловыми системами находятся в соответствующих пакетах: для ext2 и ext3 это `e2fsprogs`, для reiserfs – `reiserfs-utils`, xfs – `xfsprogs`, jfs – `jfsprogs`.

Общее назначение утилит

mkfs – создание новой файловой системы (make filesystem);

fsck – проверка файловой системы на ошибки (filesystem check).

Также существуют и другие, специфичные для разных файловых систем утилиты.

Для различения файловых систем используется указание типа файловой системы после параметра -t или в качестве компонента имени утилиты, например:

```
mkfs -t ext2 /dev/hda1
fsck.ext2 /dev/sda2
```

Конвертирование файловых систем

Для преобразования файловой системы из ext2 в ext3 необходимо дать команду

```
tune2fs -j /dev/hdX
```

Замените `hdX` на sdX в случае SCSI-диска. Для обратного преобразования необходимо смонтировать этот раздел как ext2.

Для преобразования файловой системы reiserfs-3.5.x в файловую систему reiserfs-3.6.x необходимо смонтировать эту файловую систему с опцией conv, например:

```
mount -o conv /dev/hdx /mnt/disk
```

После этого файловая система будет преобразована в формат версии 3.6.x. Обратное преобразование невозможно; следовательно, работать с сконвертированным разделом из-под ядер ветки 2.2 тоже не получится[6].

Сохранение копии диска и последующее ее использование

Для того, чтобы сохранить копию диска (например, CD-ROM), необходимо сделать следующее:

1. убедиться в наличии в текущем каталоге достаточного свободного места;
2. дать команду

   ```
   dd if=/dev/cdrom of=cdrom.iso bs=1M
   ```

3. после этого можно просмотреть содержимое файла `cdrom.iso`, смонтировав его, например, так:

   ```
   mount -o loop cdrom.iso /mnt/cdrom
   ```

[6]На самом деле использование reiserfs в ядрах 2.2 в любом случае не может быть рекомендовано.

В качестве исходного устройства для копирования также может выступать любое дисковое устройство, например дискета или жесткий диск. Кроме того, получившийся образ CD-ROM можно записать на матрицу CD-R/RW с использованием команды **cdrecord**, т.к. файл `cdrom.iso` является полным образом диска.

Для получения дополнительной информации обратитесь к man-страницам на упомянутые команды.

Использование шифрования файловых системам

В ALT Linux Master реализована система шифрования с использованием устройств `/dev/loop*` и поддержкой следующих алгоритмов: `cipher-aes*`, `cipher-blowfish*`, `cipher-des-ede3*`, `cipher-des*`, `cipher-dfc*`, `cipher-rc5*`, `cipher-serpent*`, `cipher-twofish*`.

Процедура создания зашифрованной файловой системы обычно выглядит так:

1. Необходимо создать файл необходимого размера – например, для 8 мегабайт:

   ```
   dd if=/dev/zero of=test_file count=8 bs=1M
   ```

2. Необходимо настроить алгоритм шифрования:

   ```
   modprobe cryptoloop[7]
   losetup -e blowfish /dev/loop0 test_file
   ```
   Программа спросит размер ключа:

   ```
   Available keysizes (bits): 128 160 192 256
   Keysize:
   ```
   Д алее будет запрошен пароль.После введения пароля алгоритм шифрования blowfish будет подключен
   как-то не так :-/ ---- mike, 03.11.2002, 22:18 ----
   к устройству `/dev/loop0`. Данные в зашифрованном виде будут сохраняться в файле `test_file`.

3. Необходимо создать файловую систему

   ```
   mke2fs /dev/loop0
   ```

4. Смонтировать зашифрованное устройство

   ```
   mount /dev/loop0 /mnt/disk
   ```

После этого можно работать с `/mnt/disk` как с обычным устройством, которое по окончании работы необходимо размонтировать. Для последующего использования данных необходимо повторить шаги 2 и 4. Таким образом можно организовать работу с зашифрованными файловыми системами.

Важно

ля обеспечения сохранности ваших данных рекомендуется каждый раз после изменения данных в зашифрованном файле делать его копию. Особенно это важно при обновлении ядра, т.к. файловые системы, зашифрованные на ядрах версии 2.2.x, могут не прочитаться на ядрах версии 2.4.x и наоборот.

[7] Это необходимо сделать только для ядер 2.4.x.

Глава 3. Управление пакетами

В нашем дистрибутиве программы (состоящие, как правило, из нескольких файлов) распространяются объединенными в пакеты формата RPM (RedHat Packet Manager).

С помощью программы **rpm** можно легко устанавливать, модифицировать, удалять и создавать пакеты программного обеспечения, а также получать о них разнообразную информацию. Весь дистрибутив ALT Linux Master (кроме программы начальной установки) состоит из таких пакетов.

Каждый пакет определяется именем программы, номером ее версии и номером версии релиза этой программы нашего дистрибутива, а также архитектурой пакета. Например, `bash-2.0.5-alt2.i586.rpm`: в этом пакете имя – `bash`, номер версии – `2.0.5`, номер релиза – `alt2`, архитектура – `i586`. Чем больше номер версии (или при одинаковых номерах версии – чем больше номер релиза), тем, соответственно, новее пакет.

Часто бывает удобнее, однако, применять программу **rpmdrake**, разработанную MandrakeSoft, **kpackage** из KDE, **gnorpm** из GNOME или систему **apt**, подробно описанную на стр. !!!.

Проще всего управлять пакетами через графическую оболочку **rpmdrake**, которую можно запустить через панель управления DrakConf (находящуюся на рабочем столе). Можно выбрать два режима работы – установка или удаление – при помощи кнопок в правом верхнем углу. Выделив пакет, можно получить информацию о нем, входящих в его комплект файлах, а также некоторую другую. Нажав кнопку "Удалить выбранное" или "Установить выбранное", можно удалить или установить выбранные пакеты. Часто бывает так, что требуемый пакет для нормального функционирования требует другие; в этом случае программа предложит вам установить или удалить еще несколько пакетов. При удалении пакетов необходимо соблюдать осторожность, чтобы не удалить важные части системы, например пакеты `kernel` или `glibc`. Для использования функции обновления пакетов необходимо указать программе через меню Файл->Настройки дополнительный источник пакетов, в качестве которого может выступать как ресурс Internet, так и локальный каталог или диск CD-ROM.

Установку пакетов весьма удобно выполнять и через консольную программу **urpmi** – с тем отличием, что все действия будут выполняться менее наглядно. Для установки пакетов, поставляемых ALT Linux Team, можно даже запускать программу **urpmi** не от имени суперпользователя, а от обычного пользователя; единственное, что необходимо сделать для этого – добавить его в группу **urpmi**.

Управлять пакетами можно из командной строки при помощи программы **rpm**, которая имеет следующий синтаксис:

```
rpm -options rpm_package_name
```

Далее приводятся возможные параметры.
вставить насчет rpm4, db3, ^C, rm -f /var/lib/rpm/__ ---- mike, 02.22.2002, 18:58 ----*

- Установка пакета. Вы можете установить программу, используя опцию **-i** (опции **-v** и **-h** выставлены здесь для того, чтобы включить визуальное отображение процесса установки). Например, для того, чтобы установить klyx, наберите:

  ```
  rpm -ivh klyx-0.10.9-ipl6mdk.i586.rpm
  ```
  (настоящее имя зависит от версии программы на доступном носителе).

  Заметим, что `ipl6mdk` означает, что пакет был модифицирован ALT Linux Team (ранее – IPLabs Linux Team) для русской редакции, это его шестая сборка, он входит в дистрибутив Mandrake. `i586` указывает на то, что он скомпилирован для процессоров не ниже Pentium(tm). Наличие

в имени пакета аббревиатуры `alt2` означает, что пакет был собран ALT Linux Team и это его вторая сборка.

- Обновление пакета. Для того чтобы обновить программу (с целью установки более свежей версии), нужно использовать опцию **-U**, вместо **-i**, это позволит сохранить все текущие конфигурационные файлы. Если пакета ранее не было в системе, то он будет установлен.

- Удаление пакета. Если вы желаете удалить пакет из системы, внимательно введите:

  ```
  # rpm -e имя_пакета_без_номера_версии_и_релиза
  ```

  то есть, например, для пакета `klyx`:
  D1`# rpm -e klyx`

  Если в процессе удаления пакета произойдет нарушение зависимостей, программа **rpm** сообщит об этом.

- Информация о пакете. Вы можете запросить у **rpm** ряд полезной информации о пакете, не устанавливая его – например, бывает удобно просмотреть список всех файлов пакета или краткое описание его возможностей. Для этого используйте опцию **-q** (query, запрос).

  - **-qi** используется для получения некоторой информации о ранее установленном пакете;
  - **-qip** используется для еще не установленных пакетов. В этом случае вы должны указать полный путь и имя пакета (например, `/mnt/cdrom/Mandrake/RPMS/klyx-0.10.9-ipl6mdk.i586.rpm`);
  - **-ql** используется для того, чтобы просмотреть список файлов пакета. Добавьте **p**, если пакет еще не был установлен;
  - **-qa** выдает список всех установленных пакетов (не нужно указывать имя пакета).

Будьте осторожны с опцией **--force** – ее можно употреблять только в тех случаях, когда вы хорошо знаете, что делаете. Если надо установить два или более пакетов, зависящих друг от друга, то установите их одновременно:

```
# rpm -ihv foo-1.1-3mdk.rpm libfoo-1.5-2mdk.rpm
```

Для получения дополнительной информации наберите **man rpm**.

Обеспечение и поддержание целостности системы с помощью APT

Александр Боковой

`<ab@altlinux.ru>`

Дмитрий Левин

История переиздания
Издание 0.2 23 May 2002
Рабочая версия

Введение

Современные системы на базе Linux состоят из огромного числа разделяемых библиотек, исполняемых файлов, скриптов и т.д. Удаление или изменение версии одного из составляющих систему компонентов может повлечь неработоспособность других, связанных с ним компонентов, или даже вывести из строя всю систему. В контексте системного администрирования проблемы такого рода называют нарушением целостности системы, а задачу по обеспечению наличия в системе всех необходимых программных компонент согласованных версий — задачей обеспечения целостности системы.

Для целей поддержания целостности и обеспечения возможности распространения программ в двоичном виде в первую очередь стали использоваться менеджеры пакетов (такие, как RPM в дистрибутивах *RedHat Linux* или `dpkg` в *Debian* GNU/Linux). Менеджеры пакетов давали возможность унифицировать и автоматизировать сборку двоичных пакетов и облегчали их установку, позволяя проверять наличие необходимых для работы устанавливаемой программы компонент подходящей версии непосредственно в момент установки. Однако менеджеры пакетов оказались неспособны предотвратить все возможные коллизии при установке или удалении программ, а тем более эффективно устранить нарушения целостности системы. Особенно сильно этот недостаток сказывается при обновлении систем из централизованного репозитория пакетов, в котором последние могут непрерывно обновляться, дробится на более мелкие и т.д. Этот недостаток и стимулировал создание систем управления программными пакетами и поддержания целостности системы.

Усовершенствованная система управления программными пакетами APT (Advanced Packaging Tool) первоначально была разработана для управления установкой и удалением программ в дистрибутиве *Debian* GNU/Linux. При разработке ставилась задача заменить используемую в *Debian*систему выбора программных пакетов `dselect` на новую, обладающую большими возможностями и простым пользовательским интерфейсом, а также позволяющую производить установку, обновление и повседневные "хозяйственные" работы с установленными на машине программами без необходимости изучения тонкостей используемой в дистрибутиве менеджера программных пакетов.

Эти привлекательные возможности были долгое время доступны только пользователям *Debian* GNU/Linux, поскольку в APT поддерживалась только один менеджер пакетов, а именно применяемый в *Debian* GNU/Linux менеджер пакетов `dpkg`, несовместимый с используемой в *ALTLinux* RPM. Эта несовместимость заключается прежде всего в различии используемых форматов данных (хотя сущесвуют программы-конверторы), хотя имеютсяа и другие различия, обсуждение которых выходит за рамки изложения.

APT, однако, изначально проектировался, как не зависящий от конкретного метода работы с установленными в системе пакетами, и эта особенность позволила разработчикам из бразильской компании *Conectiva* [http://www.conectiva.com.br] реализовать в нем поддержку менеджера пакетов RPM. Таким образом, пользователи основанных на RPM дистрибутивов (*ALTLinux* входит в их число) получили возможность использовать этот мощный инструмент.

APT и в настоящее время находится в стадии разработки, а текущая версия с поддержкой RPM классифицируется, как нестабильная. Это, тем не менее, не означает, что операции, выполняемые посредством APT, безусловно приведут к нестабильности системы. Более того, с помощью APT возможен строгий контроль за целостностью системы: проверка нарушенных зависимостей между установленными пакетами и исправление выявленных ошибок.

Системы управления пакетами RPM и `dpkg` используют концепции представления программного обеспечения в виде набора компонент — программных пакетов. Такие компоненты содержат в себе набор исполнимых программ и вспомогательных файлов, необходимых для корректной работы ПО. Часто компоненты, используемые различными программами, выделяют в отдельные пакеты и помечают, что для работы ПО, предоставленного пакетом A, необходимо установить пакет B. В таком случае говорят, что пакет A *зависит* от пакета B или что между пакетами A и B существует *зависимость*.

Отслеживание зависимостей между такими пакетами представляет собой серьезную задачу для любого дистрибутива — некоторые компоненты могут быть взаимозаменяемыми и при удовлетворении тех или иных требований может обнаружиться несколько пакетов, предлагающих затребованный ресурс.

Задача контроля целостности и непротиворечивости установленного в системе ПО еще сложнее. Представим, что некие программы A и B требуют наличия в системе компоненты C версии 1.0. Обновление версии пакета A, требующее обновления компоненты C до новой, использующей новый интерфейс доступа, версии (скажем, до версии 2.0), влечет за собой обязательное обновление и программы B.

Для автоматизации этого процесса и применяется APT. Такая автоматизация достигается созданием одного или нескольких внешних репозиториев, в которых хранятся пакеты программ и относительно которых производится сверка пакетов, установленных в системе. Репозитории могут содержать как официальную версию дистрибутива, обновляемую его разработчиками по мере выхода новых версий программ, так и локальные наработки, например, пакеты, разработанные внутри компании.

Таким образом, в распоряжении APT находятся две базы данных: одна, описывающая установленные в системы пакеты и вторая, с описанием внешнего репозитория. APT отслеживает целостность установленной системы и, в случае обнаружения противоречий в зависимостях пакетов, руководствуется сведениями о внешнем репозитории для разрешения конфликтов и поиска корректного пути их устранения.

Использование APT

Система APT состоит из нескольких утилит. Главной и наиболее часто используемой является утилита управления пакетами **apt-get**: она автоматически определяет зависимости между пакетами и строго следит за их соблюдением при выполнении любой из следующих операций: установка, удаление или обновление пакетов.

apt-get позволяет устанавливать в систему пакеты, требующих для своей работы других, пока еще не установленных. В этом случае он определяет, какие из отсутствующих пакетов необходимо установить, и доустанавливает их, пользуясь всеми доступными репозиториями. Для того, чтобы **apt-get** мог использовать тот или иной репозиторий, информацию о нем необходимо поместить в файл `/etc/apt/sources.list` и выполнить команду

```
# apt-get update
```

Эту команду необходимо также выполнять каждый раз, когда вы собираетесь работать с репозиторием после длительного перерыва, так как при поиске пакетов APT должен руководствоваться базой данных, отражающей актуальное состояние репозитория. Такая база данных создается заново каждый раз, когда в репозитории происходит изменение: добавление, удаление или переименование пакета. Для ускорения работы **apt-get** хранит локальную копию базы данных, которая через некоторое время может уже не соответствовать реальному состоянию репозитория.

В качестве источника пакетов можно использовать и компакт-диски дистрибутива, поскольку на каждом диске присутствует вся необходимая для APT информация о содержащихся на нем пакетах. Для этого необходимо использовать утилиту **apt-cdrom** с единственным параметром `add`:

```
# apt-cdrom add

Используется точка монтирования CD-ROM /mnt/cdrom/
Размонтирование CD-ROM
```

```
Пожалуйста, вставьте диск в устройство и нажмите <Enter>
Монтирование CD-ROM
Используется точка монтирования CD-ROM /mnt/cdrom
Определение... [8d56fef8c93e5255540c843e4b9f49fa-2]
Сканирование диска в поисках индексных файлов...
Найдено 1 бинарных пакетов и 1 исходных пакетов.
Пожалуйста, укажите имя этого диска, например, 'Мой Дистрибутив Диск
1':
Master Disk 1
Этот диск называется
'Master Disk 1'
Reading Indexes... Завершено
Reading Indexes... Завершено
Запись нового списка источников
Список источников для этого диска:
rpm cdrom:[Master Disk 1]/ Mandrake Master
rpm-src cdrom:[Master Disk 1]/ Mandrake Master
```

Повторите этот процесс для всех CD в вашем наборе.

После этого в `/etc/apt/sources.list` появится запись о подключенном диске:

```
rpm cdrom:[Master Disk 1]/ i586/Mandrake Master
rpm-src cdrom:[Master Disk 1]/ Mandrake Master
```

Если подключение к Internet отсутствует, то следует закомментировать те строчки в `/etc/apt/sources.list`, в которых говорится о ресурсах, доступных по Сети. Непосредственно после установки дистрибутива *ALTLinux* в `/etc/apt/sources.list` указаны несколько таких источников:

- репозиторий обновлений в системе безопасности дистрибутива;
- бинарные пакеты из репозитория *Sisyphus [http://www.altlinux.ru/index.php?module=sisyphus]* ("Сизиф");
- исходные тексты архивов, использовавшихся для сборки пакетов в репозитории *Sisyphus [http://www.altlinux.ru/index.php?module=sisyphus]*.

Проект *Sisyphus [http://www.altlinux.ru/index.php?module=sisyphus]* команды *ALT Linux Team [http://www.altlinux.ru]* содержит большое количество программ, в том числе и не вошедших в тот или иной дистрибутив. Следует иметь в виду, что он не является самостоятельным дистрибутивом, а отражает текущее состояние разработки и может содержать нестабильные версии пакетов. Периодически на базе этого проекта выпускаются отдельные оттестированные "срезы"-дистрибутивы. Репозиторий ежедневно обновляется разработчиками, поэтому необходимо синхронизировать локальную базу данных с сервером *ALTLinux* (или его зеркалами) перед началом работы с APT. Такую синхронизацию достаточно делать один раз в день командой **apt-get update**. Для репозиториев, подключенных командой **apt-cdrom add**, синхронизацию достаточно сделать один раз в момент подключения.

Установка или обновление пакета

Установка пакета с помощью APT, выполняется командой

```
# apt-get install имя-пакета
```

Иногда, в результате операций с пакетами без использования APT, целостность системы нарушается и **apt-get** отказывается выполнять операции установки, удаления или обновления. В этом случае необходимо повторить операцию, задав опцию `-f`, заставляющую **apt-get** исправить нарушенные зависимости, если это возможно. В этом случае необходимо внимательно следить за сообщениями, выдаваемыми **apt-get**, анализировать их и четко следовать рекомендациям программы.

Команда **apt-get install имя_пакета** используется и для обновления уже установленного пакета или группы пакетов. В этом случае **apt-get** дополнительно проверяет, не обновилась ли версия пакета в репозитории по сравнению с установленным в системе. Если вы не знаете точное название пакета, для его поиска можно воспользоваться утилитой **apt-cache**, описанной ниже.

Пример 3.1. Установка пакета `clanbomber` командой apt-get install clanbomber приведет к следующему диалогу с APT:

```
Обработка файловых зависимостей... Завершено
Чтение списков пакетов... Завершено
Построение дерева зависимостей... Завершено
Следующие дополнительные пакеты будут установлены:
clanlib clanlib-mikmod clanlib-sound libmikmod
Следующие НОВЫЕ пакеты будут установлены:
clanbomber clanlib clanlib-mikmod clanlib-sound libmikmod
0 пакетов будет обновлено, 5 будет добавлено новых,
0 будет удалено(заменено) и 0 не будет обновлено.
Необходимо получить 0B/2577kB архивов. После распаковки 3862kБ будет
использовано.
Продолжить? [Y/n] y
Выполняется программа RPM (/bin/rpm -Uv --replacepkgs -h)...
Подготовка... ########################################
libmikmod ############################################
clanlib ##############################################
clanlib-mikmod #######################################
clanlib-sound ########################################
clanbomber ###########################################
```

Внимание

apt-get всегда спрашивает подтверждение выполнения операции установки и обновления, за исключением случая, когда реально требуется установить в систему (или обновить) только один пакет. Если вы не уверены в том, что результате выполнения операции система останется работоспособной, запустите **apt-get** с опцией `-S`, которая покажет отчет выполнения операции обновления, но реально обновление произведено не будет.

В случае обнаружения противоречий между установленными в системе пакетами, следует запустить команду **apt-get -f install**, и APT постарается разрешить найденные конфликты, предложив удалить или заменить конфликтующие пакеты. Любые действия в этом режиме обязательно требуют подтверждения со стороны пользователя.

Удаление установленного пакета

Для удаления пакета используется команда **apt-get remove имя_пакета**. Для того, чтобы не нарушать целостность системы, будут удалены и все пакеты, зависящие от удаляемого: если отсутствует необходимая для работы приложения библиотека, то само приложение становится бесполезным). В случае удаления пакета, который относится к базовым компонентам системы,

apt-get потребует дополнительного подтверждения производимой операции с целью предотвратить возможную случайную ошибку.

Запрос на подтверждение операции удаления базовой компоненты системы выглядит следующим образом:

```
# apt-get remove filesystem
Обработка файловых зависимостей... Завершено
Чтение списков пакетов... Завершено
Построение дерева зависимостей... Завершено
Следующие пакеты будут УДАЛЕНЫ:
basesystem filesystem ppp sudo
Внимание: следующие базовые пакеты будут удалены:
В обычных условиях этого не должно было произойти, надеемся, Вы точно
представляете, что требуете!
basesystem filesystem (по причине basesystem)
0 пакетов будет обновлено, 0 будет добавлено новых, 4 будет
удалено(заменено) и 0 не будет обновлено.
Необходимо получить 0B архивов. После распаковки 588kБ будет
освобождено.
Вы собираетесь предпринять что-то потенциально вредное
Для продолжения, наберите по-английски 'Yes, I understand this may be
bad'
(Да, я понимаю, что это может быть плохо).
```

Каждую ситуацию, в которой APT генерирует такое сообщение, необходимо рассматривать отдельно. Однако, вероятность того, что после выполнения этой команды система окажется неработоспособной, очень велика.

Обновление всех установленных пакетов

Для обновления всех установленных пакетов используется команда **apt-get upgrade**. Она позволяет обновить те и только те установленные пакеты, для которых в репозиториях, перечисленных в `/etc/apt/sources.list`, имеются новые версии; при этом из системы не будут удалены никакие другие пакеты. Этот способ полезен при работе со стабильными пакетами приложений, относительно которых известно, что они при смене версии изменяются несущественно.

Иногда, однако, происходит изменение в именовании пакетов или изменение их зависимостей. Такие ситуации не обрабатываются командой **apt-get upgrade**, в результате чего происходит нарушение целостности системы: появляются неудовлетворенные зависимости. Например, переименование пакета `MySQL-shared`, содержащего динамически загружаемые библиотеки для работы с СУБД MySQL, в `libMySQL`, отражая общую тенденцию к наименованию библиотек в дистрибутиве, не приводит к тому, что установка обновленной версии `libMySQL` требует удаления старой версии `MySQL-shared`. Для разрешения этой проблемы существует режим обновления в масштабе дистрибутива — **apt-get dist-upgrade**.

В случае обновления всего дистрибутива APT проведет сравнение системы с репозиторием и удалит устаревшие пакеты, установит новые версии присутствующих в системе пакетов, а также отследит ситуации с переименованиями пакетов или изменения зависимостей между старыми и новыми версиями программ. Все, что потребуется поставить (или удалить) дополнительно к уже имеющемуся в системе, будет указано в отчете **apt-get**, которым APT предварит само обновление.

При работе с *Sisyphus [http://www.altlinux.ru/index.php?module=sisyphus]* для обновления системы рекомендуется использовать команду **apt-get dist-upgrade**.

Поиск в репозитории

Для поиска нужного пакета можно воспользоваться утилитой **apt-cache**, которая позволяет искать не только по имени пакета, но и по его описанию.

Команда **apt-cache search подстрока** позволяет найти все пакеты, в именах или описании которых присутствует указанная подстрока. Например:

```
$ apt-cache search master

xcdroast - A GUI program for burning Cds
bluefish - A WYSIWYG GPLized HTML editor
xmess - X-Mess Multi Emulator Super System
mkisofs - Creates an image of an ISO9660 filesystem
```

В кратком описании каждого из перечисленных пакетов не присутствует слово “master”.

Для того, чтобы подробнее узнать о пакете, можно воспользоваться командой **apt-cache show**, которая покажет информацию о пакете из репозитория и в том числе:

```
Пакет: bluefish
Секция: Networking/WWW
Размер установленных пакетов: 2018
Упаковщик: AEN <aen@logic.ru>
Версия: 1:0.7-alt0.1
..
Предоставляет: bluefish
Архитектура: i586
..
Имя файла: bluefish-0.7-alt0.1.i586.rpm
Описание: A WYSIWYG GPLized HTML editor
Bluefish is a programmer's HTML editor, designed to save the
experienced webmaster some keystrokes.
It features a multiple file editor, multiple toolbars, custom menus,
image and thumbnail dialogs, open from the web, HTML validation and
lots of wizards.
It is in continuous development, but it's already one of the best
WYSIWYG HTML editors.
```

Наличие слова “webmaster” и объясняет наличие этого пакета в результате поиска по слову “master”.

Настройка APT

APT позволяет взаимодействовать с репозиторием с помощью различных протоколов доступа. Наиболее популярные — HTTP и FTP, именно они используются для работы с *Sisyphus [http://www.altlinux.ru/index.php?module=sisyphus]*. Однако существуют и некоторые дополнительные методы.

Настройка описаний репозиториев задается в файле `/etc/apt/sources.list` в следующем виде:

```
rpm [подпись] метод:путь база название
rpm-src [подпись] метод:путь база название
```

- `rpm` или `rpm-src` — тип репозитория (скомпилированные программы или исходные тексты);

- `подпись` — опциональная строка-указатель на сигнатуру разработчиков. Сигнатуры описываются в файле `/etc/apt/vendor.list`;
- `метод` — способ доступа к репозиторию: `ftp`, `http`, `file`, `rsh`, `ssh`, `cdrom`;
- `путь` — путь к репозиторию в терминах выбранного метода;
- `база` — относительный путь к базе данных репозитория;
- `название` — название репозитория;

Например, при установке *ALTLinux* в `/etc/apt/sources.list` записываются следующие настройки:

```
# Sisyphus
rpm [alt] ftp://ftp.altlinux.ru/pub/distributions/ALTLinux/Sisyphus i586/Mandrake sisyphus
rpm-src [alt] ftp://ftp.altlinux.ru/pub/distributions/ALTLinux/Sisyphus i586/Mandrake sisyphus
```

При этом, реальная структура репозитория по адресу `ftp://ftp.altlinux.ru/pub/distributions/ALTLinux/Sisyphus` выглядит следующим образом:

```
ftp://ftp.altlinux.ru/pub/distributions/ALTLinux/Sisyphus
|-- SRPMS
|-- i586
| |-- Mandrake
| | |-- RPMS
| | |-- RPMS.sisyphus -> RPMS
| | |-- SRPMS.sisyphus -> ../../SRPMS
| | |-- base
```

Более подробное описание команд программы **apt-get** можно найти в справочной системе дистрибутива на страницах `apt-get(8)` и `apt.conf(5)`.

Создание собственного репозитория

Вы можете создавать собственные репозитории и использовать их для обновления и/или установки собственных программ. Для этого необходимо создать структуру каталогов, подобную описанной выше. Вы можете выбирать из следующих компонентов (перечисляются по дереву выше):

i586	архитектура, под которую собраны пакет (совпадает с таковой в имени бинарных RPM-пакетов)
Mandrake	название подсистемы. Этот уровень в дереве может отсутствовать (то есть, каталоги `RPMS` и `base` могут идти сразу следом за архитектурой)
RPMS	каталог, в котором размещены бинарные пакеты
SRPMS	каталог, в котором размещены пакеты с исходными текстами программ
RPMS.sisyphus	ссылка на каталог `RPMS`. При этом `sisyphus` заменяется на собственное название репозитория, например, `local`
base	служебный каталог, в котором размещается база данных APT

Следующий шаг в создании своего репозитория заключается в помещении бинарных пакетов в каталог `RPMS`, а пакетов с исходными текстами — в каталог `SRPMS` и в генерации служебной информации для APT при помощи команды **genbasedir**; ее формат:
`genbasedir` [опции] { название подсистемы } { репозиторий 1 } [репозиторий 2 ...]

Из опций, список которых можно увидеть при запуске **genbasedir** без параметров, наиболее важной является опция `--topdir`, позволяющая указать путь к репозиторию. Все остальные параметры задаются относительно этого пути. Выглядит это следующим образом. Допустим, что наше дерево каталогов выглядит так:

```
/opt/repository/
|-- SRPMS
|-- SRPMS.security
|-- i386
| |-- MyDistro
| | |-- RPMS
| | |-- RPMS.local -> RPMS
| | |-- RPMS.security
| | |-- SRPMS.local -> ../../SRPMS
| | |-- SRPMS.security -> ../../SRPMS.security
| | |-- base
```

Тогда строка запуска **genbasedir** будет выглядеть так:

```
$ genbasedir --topdir=/opt/repository i386/MyDistro local security
```

Этой командой мы создадим информацию для APT в двух репозиториях — `local` и `security`. Для того, чтобы воспользоваться этой информацией, необходимо прописать доступ к репозиториям в `/etc/apt/sources.list`:

```
rpm file:/opt/repository i386/MyDistro local
rpm-src file:/opt/repository i386/MyDistro local
rpm file:/opt/repository i386/MyDistro security
rpm-src file:/opt/repository i386/MyDistro security
```

Репозиторий `MyDistro.security`, хранящий пакеты с исправлениями ошибок в системе безопасности, имеет смысл подписывать PGP-ключом, чтобы при установке пакета можно было проверить аутентичность репозитория и хранящихся в нем пакетов. Для этого необходимо создать соответствующий PGP-ключ, используя программу GnuPG (**gpg**) и запомнить его отпечаток (fingerprint) на клиентских машинах в файле `/etc/apt/vendors.list` в формате:

```
simple-key "краткое название ключа" {
Fingerprint "отпечаток ключа";
Name "Полное название ключа";
}
```

Примером может служить ключ службы безопасности *ALT Linux Team [http://www.altlinux.ru]*, которым подписаны пакеты репозитория *Sisyphus [http://www.altlinux.ru/index.php?module=sisyphus]* и обновления безопасности для различных дистрибутивов *ALTLinux*:

```
simple-key "alt" {
Fingerprint "BB1DD157A9722953847C5DB25B433A0EEAC91CA0";
```

```
Name "ALT Security Team <security@altlinux.ru>";
}
```

Для того, чтобы APT проверял аутентичность подписи, необходимо указать, что соответствующий репозиторий подписан PGP-ключом в `/etc/apt/sources.list`:

```
rpm [alt] file:/opt/repository i386/MyDistro security
rpm-src [alt] file:/opt/repository i386/MyDistro security
```

Необходимо также сгенерировать информацию для APT в репозитории с указанием опции `--sign` команды **genbasedir**. Дополнительно, можно указать идентификатор ключа, если он отличается от ключа по умолчанию, используя опцию `--uid=идентификатор`. Значением этой опции является идентификатор ключа в том виде, как он передается программе GnuPG в опции `--default-key`:

```
$ genbasedir --topdir=/opt/repository --sign \
--uid='ALT Security Team' i386/MyDistro security
```

Операцию создания служебной информации для APT необходимо производить каждый раз, когда в репозиторий вносятся изменения.

Часть III. Безопасность

Часть IV. Сеть

Глава 4. Общая информация

Сеть – это система, предназначенная для обмена информацией между различными ее узлами, в том числе компьютерами. Рассмотреть все способы работы с сетями, равно как и различные варианты их настройки, в данном руководстве невозможно – поэтому в данном разделе описываются способы работы дистрибутива ALT Linux Master с различными типами сетей.

Приведенная здесь информация является набором общих рекомендаций и советов, применимых к данному дистрибутиву. За более подробной документацией вы можете обратиться как к различной печатной литературе, так и к электронным документам из серии HOWTO, главным из которых является Networking-Overview-HOWTO, который помимо того, что содержит ссылки на другие источники информации, переведен на множество языков (в том числе русский).

Для работы с сетью в дистрибутиве ALT Linux Master можно использовать как общие для всех UNIX-подобных систем команды (например, **ifconfig**, **ping**, **traceroute** и т.д.), так и специальную систему скриптов, написанную специально для облегчения работы с сетью.

Фактически **draknet** всего лишь производит настройки путем изменения некоторых файлов конфигурации из дистрибутива, данные из которых потом используются различными программами. Опытным системным администраторам следует знать месторасположение и назначение этих файлов:

1. `/etc/sysconfig/network` – общие настройки сети;
2. `/etc/sysconfig/network-scripts` – файлы настроек и скрипты для работы с различными типами сетевых устройств и подключений. Например файл `/etc/sysconfig/network-scripts/ifcfg-eth0` содержит информацию о настройке сетевой Ethernet-карты с интерфейсом eth0;
3. `/etc/ppp` – файлы настройки протокола ppp;
4. `/etc/init.d/` – каталог с различными инициализационными скриптами, среди которых скрипты **network**, **firewall** и некоторые другие отвечают за настройку сети в момент загрузки и выключения компьютера.

В общем случае для запуска всех настроенных на данном компьютере соединений (настроенных на автоматический запуск при загрузке) необходимо дать команду **/etc/init.d/network start**, для останова и перезапуска соответственно используются ключи **stop** и **restart**.

Для запуска отдельно взятого интерфейса можно дать команду **ifup** интерфейс – например, **ifup ppp0**.

Для выключения интерфейса можно применить команду **ifdown** интерфейс.

Утилита draknet

Наиболее распространенные варианты настройки сети в Master можно произвести с помощью утилиты **draknet** – поэтому рекомендуется сначала попробовать воспользоваться ей, а при недостаточности фукнциональности этой утилиты попробовать произвести требуемые настройки вручную.

Сразу после запуска **draknet** предлагает попробовать определить автоматически сетевые устройства, установленные на вашем компьютере. Обычно стоит выбрать эту опцию за исключением случаев, когда она создает проблемы. После этого вам будет предложено настроить один из сетевых адаптеров, причем устройства, найденные автоматически, будут отмечены особо; далее необходимо ответить на вопросы (указать требуемые параметры), после чего ваша сеть будет настроена.

Глава 5. Подключение к сети

Локальная сеть

Как известно, локальная сеть обычно строится на основе технологии Ethernet; если ваша не является исключением, перед настройкой параметров сети стоит убедиться, что настроена ваша сетевая карта (см. стр.!!!).

Далее для настройки локальной сети можно запустить утилиту **draknet**, с помощью которой можно задать или изменить необходимые параметры.

Из них необходимо знать следующие:

1. IP-адрес данного компьютера;
2. маску подсети;
3. доменное имя данного компьютера;
4. IP-адрес(а) серверов DNS для данной локальной сети;
5. IP-адрес стандартного шлюза для данной сети (обычно это нужно для выхода в Internet).

Параметры, отмеченные звездочкой, являются обязательными.

Если в сети не используются специально выделенные IP-адреса для компьютеров – скорее всего, они раздаются автоматически DHCP-сервером. Для настройки сети с его использованием необходимо выбрать в утилите **draknet** соответствующий пункт.

Настройка локальных сетей, отличных от Ethernet, выходит за рамки этого описания; для получения информации на эту тему можно обратиться к HOWTO.

Для проверки работоспособности сети TCP/IP можно воспользоваться следующей схемой.

Для начала убедитесь в работоспособности только что настроенного вами интерфейса при помощи команды

```
$ ping ip_адрес_интерфейса
```

При получении ответов от него можно проверить командой **ping** доступность любого внешнего интерфейса из той же подсети, что и только что настроенный. После этого необходимо проверить работоспособность серверов DNS с помощью команды

```
$ host имя_хоста имя_сервера_DNS
```

Для проверки возможности доступа к Internet необходимо дать команду **ping** `интернет_сервер`, например, **ping www.altlinux.ru**.

Глава 6. Выход в Internet

Настройка модемного соединения

Таковая осуществляется выбором пункта "Обычное модемное соединение" (Normal modem connection). Если вы не выбрали автоматическое определение устройств – необходимо указать порт, к которому подключен модем. Далее надо ответить на следующие вопросы программы:

1. название соединения;
2. номер телефона провайдера;
3. тип набора номера – импульсный или тоновый. На большинстве АТС в России и СНГ доступен только импульсный тип набора;
4. идентификатор пользователя (Login ID), данный вам провайдером;
5. пароль для входа;
6. тип аутентификации. По умолчанию стоит оставить PAP (так как этот тип наиболее часто используется). Использовать тип CHAP стоит только в случаях, если на удаленном сервере требуется именно этот тип (обычно он используется системами на основе Windows NT). Если же ваш провайдер требует аутентификации вручную (т.е. ввод имени и пароля при доступе к системе), можно сделать этот процесс автоматизированным через скрипт доступа. Для полностью ручного доступа можно выбрать пункт Terminal-Based;
7. имя домена – необходимо заполнять только в том случае, если ваш провайдер требует этого. По умолчанию лучше всего оставить пустым;
8. первичный DNS – если ваш провайдер требует явного указания сервера DNS для выхода в Internet. Обычно сервера DNS назначаются провайдерами автоматически;
9. вторичный DNS – то же самое.

Определитесь, нужно ли, чтобы соединение устанавливалось автоматически при загрузке системы. Для модемного соединения это обычно не нужно.

Затем вы можете проверить новое соединение на работоспособность.

После настройки соединения производить подключение к Internet через модем возможно, например, следющими способами:

1. **/sbin/ifup ppp0** – этот способ работает как из консоли, так и из X-сессии;
2. посредством графической утилиты **kppp**, использующей графический интерфейс KDE.

Замечание

При настройке обратите внимание на то, что при выставленном значении default gateway PPP-соединение не установится; проверить маршрутизацию можно при помощи команды **/sbin/route -n**.

Организация шлюза

Для выхода в Internet должны выполняться следующие условия:

1. присутствие физического канала в Internet (например, модема);
2. функционирование этого канала на одном из компьютеров сети;
3. настроенная соответствующим образом маршрутизация для пересылки пакетов из внутренней локальной сети в Internet;
4. настройка всех компьютеров локальной сети на использование системы, имеющей физическое соединение с Internet, в качестве стандартного шлюза для данной сети; кроме того, должны быть правильно указаны DNS-серверы.

Автоматическую настройку данного варианта соединения можно сделать с помощью утилиты **drakgw**; она производит следующие действия:

1. присваивает интерфейсу `eth0` IP-адрес 192.168.0.1;
2. настраивает сервис DHCP для присвоения клиентам адресов из подсети 192.168.0.x;
3. конфигурирует маршрутизацию таким образом, что все клиентские компьютеры, получившие сетевые настройки через этот DHCP-сервер, будут получать доступ в Internet через указанную систему;
4. создает кэширующий сервер DNS.

В итоге всего этого после завершения работы программы **drakgw** при условии, что ваш компьютер имел настроенный ранее доступ к Internet, все клиентские компьютеры из локальной сети также получают разделяемый доступ к Internet через него.

Для получения более подробной информации по настройке разделяемого доступа к Internet можно прочитать Internet-Sharing-HOWTO.

Маршрутизация

Для настройки обычной маршрутизации в дистрибутиве ALT Linux Master используется стандартная утилита **route**; расширенная конфигурируется при помощи дополнительных утилит (например, **iproute2**).

Глава 7. Настройка почтового сервера Postfix

Михаил Забалуев
Mikhail Zabaluev

История переиздания

Издание 1.2 19 Nov 2002

Изменения, отражающие нынешнюю организацию пакетов. Добавлен абзац о пакетах Postfix и дополнена информация о SASL-enabled SMTP с учётом наличия пакета postfix-smtpd-sasl.

Издание 1.1 23 Oct 2002

Первая запись в истории изменений документа, существовавшего некоторое время и вошедшего в документацию ALT Linux Master 2.0, поэтому версия 1.1. Документ переработан, убраны численные character entities и "хитрые" пробелы, возвращена семантика разметки inline-элементов. Дополнены и исправлены некоторые пояснения и примеры. Добавлено пояснение о параметре mailbox_size_limit. Убран эпитет "прекрасно" в упоминании работы fetchmail: в последнее время эта программа проявляет нестабильность.

Возможно, вас удивит то, что сервер передачи электронной почты Postfix рекомендуется к установке в любой конфигурации *ALT Linux [http://www.altlinux.ru]*. Это объясняется тем, что в Unix-подобных системах способность отправлять почту с помощью простого вызова команды из командой оболочки практически обязательна. Некоторые программы (например, сервис cron) пользуются этим для отправки сообщений пользователям. Пересылкой всей электронной почты, проходящей через машину, занимается MTA (Mail Transport Agent), в нашем случае это Postfix. Хотя многие почтовые клиенты способны отправлять сообщения на удалённый SMTP-сервер, имеет смысл поручить и эту задачу системному процессу, чтобы достигнуть эффекта “отправил и забыл”. Существуют и другие популярные MTA (например qmail, exim), но они по разным причинам не вошли в данную версию дистрибутива. Sendmail, ветеран Интернета, проигрывает Postfix по ряду параметров, в том числе безопасности, к тому же он неоправданно сложен в настройке. В данном руководстве мы ограничимся рекомендациями по настройке Postfix для нескольких типичных конфигураций. Более полные сведения можно получить из превосходной документации на английском языке, которая входит в состав пакета `postfix`.

Пакеты Postfix

Базовый RPM-пакет для установки сервера Postfix в *ALT Linux [http://www.altlinux.ru]* носит, как нетрудно догадаться, имя `postfix`. Есть также несколько дополнительных пакетов, предоставляющих сервисы по приёму и доставке сообщений по сети с различной степенью защищённости. Один из пакетов SMTP-серверов, `postfix-smtpd` либо `postfix-smtpd-sasl`, нужен Postfix для того, чтобы принимать сообщения по протоколу SMTP (или ESMTP) как извне, так и локально. Второй из этих пакетов реализует расширения SASL; подробнее об этом см. далее. Есть также пакет `postfix-sasl`, который расширяет возможности доставки сообщений на случай, если какие-либо принимающие серверы, с которыми взаимодействует данный сервер, пользуются авторизацией по методу SASL.

Конфигурационные файлы

Файлы настройки Postfix располагаются в каталоге `/etc/postfix`. Основные параметры определяются в файле `main.cf`; в частности, параметры, о которых говорится далее в этой главе, устанавливаются в этом файле, если другой не указан специально.

В изначальном виде этот файл содержит конфигурацию, позволяющую серверу работать в пределах машины, а также развёрнутые комментарии с примерами. После редактирования конфигурации при работающем Postfix её нужно активизировать командой **service postfix reload** или просто **postfix reload**.

Доменная информация

Имя хоста и домена, которые считаются локальными при обработке email-адресов, необходимы для функционирования почтового сервера. Если эти имена для Postfix должны быть отличны от того, что выдаёт команда **hostname**, установите их с помощью параметров *myhostname* и *mydomain*.

Postfix на dialup-машине

Существует несколько проблем, возникающих при попытке отправки исходящей почты с машин, которые не являются полноценными узлами интернет, например, в системах с модемным и другими непостоянными соединениями не всегда возможно немедленно отправить сообщения удалённым адресатам по SMTP и их приходится держать в очереди до тех пор, пока соединение не будет установлено. Для этого используется параметр *defer_transports*, например:

```
defer_transports = smtp
```

Доставка активизируется командой **/usr/sbin/sendmail -q**, которая в *ALT Linux [http://www.altlinux.ru]* исполняется автоматически при установке PPP-соединения.

Будучи полноценным MTA, Postfix способен находить серверы, обслуживающие получателей сообщений, при помощи DNS. Тем не менее, для dialup-машин непосредственная доставка сообщений нежелательна, поскольку время соединения ограничено. К тому же это излюбленная тактика распространителей спама, поэтому многие серверы сверяют IP-адрес отправителя с базой известных адресов провайдерских пулов, после чего сообщения с таких адресов отвергаются. Поэтому целесообразно доверить доставку исходящей почты SMTP-серверу провайдера. Этим управляет параметр *relayhost*, например:

```
relayhost = [smtp.provider.net]
```

Postfix на клиентской машине локальной сети

Рабочие станции локальной сети или машины в провайдерской сети, отделённой от Интернета с помощью межсетевого экрана/NAT, должны переправлять исходящую почту на почтовый сервер, обслуживающий данную сеть. Для этого также используется параметр *relayhost*, описанный выше. Если сервер задан IP-адресом, можно отключить использование DNS для ускорения работы:

```
disable_dns_lookups = yes
```

Для того, чтобы в доменной части адреса отправителя фигурировал домен сети, а не имя конкретной машины, установите параметр *myorigin* в имя домена:

```
myorigin = $mydomain
```

Если почтовые ящики пользователей монтируются с сервера по NFS, Postfix на клиентских машинах служит лишь для отправки почты. В такой конфигурации следует отключить агенты `local` и `smtp` в файле `/etc/postfix/master.cf`.

Почтовый сервер для небольших доменов и сетей

Домены, для которых сервер получает почту, отличные от значения *mydomain* и не сконфигурированные как виртуальные домены Postfix (см. ниже), нужно перечислить с помощью параметра *mydestination* либо в дополнительном файле, на который ссылается этот параметр. Аналогичным образом параметр *mynetworks* описывает блоки IP-адресов, которые считаются внутренними и с которых разрешён приём исходящих сообщений. Не следует записывать в *mynetworks* блоки адресов, не принадлежащих сети, которую обслуживает сервер, поскольку этим могут воспользоваться распространители спама.

Для SMTP-аутентификации внешних пользователей, желающих отправлять сообщения через данный сервер, можно использовать поддержку авторизации SASL. Пакет `postfix-smtpd-sasl` предоставляет альтернативу `postfix-smtpd` со включенной поддержкой SASL; возможный недостаток этого расширения — включение кода, в меньшей степени проверенного в плане безопасности. Настройка аутентификации SASL описана в файле `SASL_README` в документации Postfix.

Преобразование глобальных адресов в локальные адреса назначения устанавливается с помощью таблиц типа `virtual` (см. virtual(5)):

```
virtual_maps = hash:/etc/postfix/virtual
```

Пример содержимого `/etc/postfix/virtual`:

```
domain1.ru # Домен в стиле Postfix (текст здесь игнорируется)
name1@domain1.ru user1
name2@domain2.ru user2@otherbox
@domain2.ru user3
```

После редактирования оттранслируйте таблицу в рабочий образ командой **postmap /etc/postfix/virtual**.

Если каким-либо пользователям сети почта должна доставляться по SMTP на их рабочие станции (это предполагает, что на их машинах работают MTA), подставляйте в доменной части их адресов имена машин в таблицах `virtual` либо `aliases` (см. ниже).

Алиасы и преобразования адресов

Имена локальных адресатов либо совпадают с именами пользователей системы, либо подставляются из таблицы `aliases` (см. aliases(5)):

```
alias_maps = hash:/etc/postfix/aliases
alias_database = hash:/etc/postfix/aliases
```

При установке Postfix “с нуля” в этой таблице создаётся алиас на имя `root` для доставки всей корреспонденции, предназначенной администратору и поступающей на другие системные адреса, на имя реального пользователя, который осуществляет функции администратора. Изначально им становится первый зарегистрированный в системе реальный пользователь. Таблица алиасов отличается от остальных таблиц, используемых Postfix; имена слева, которые являются ключами для поиска, отделяются от значений справа двоеточиями. Адресаты справа перечисляются через запятую и могут быть адресами, командами (обозначаются символом `|` в начале правой части; сообщение подаётся на стандартный поток ввода команды) и именами файлов:

```
John.Smith: john
chief: chief@bosscomputer
trio: stock, hausen, walkman
robot: | /usr/bin/robot --process-mail
filebox: /dir/file
```

Рабочий образ таблицы строится с помощью команд **postalias /etc/postfix/aliases** или **newaliases**. При отправке сообщения Postfix генерирует адрес отправителя из имени пользователя и собственного домена (или значения *myorigin*). Даже если почтовый клиент выставил заголовок `From:`, этот адрес попадает в служебную информацию сообщения и может быть использован

получателем, что не всегда желательно. Преобразование адресов отправителей к глобальным адресам можно задать в таблице типа `canonical` (см. canonical(5)):

```
sender_canonical_maps = hash:/etc/postfix/sender_canonical
```

Аналогичная таблица `recipient_canonical` и соответствующий параметр *recipient_canonical_maps* могут быть использована для преобразования адресов назначения. Для актуализации изменений таблиц используйте команду **postmap** *имя_таблицы*.

Борьба со спамом и почтовыми вредителями

Противодействие спаму (массовым рассылкам непрошенной корреспонденции) — отдельная большая тема, которую невозможно полностью раскрыть в этом руководстве; здесь даны лишь несколько практических советов применительно к конфигурации Postfix. По умолчанию сервер сконфигурирован так, что отвергает попытки переслать сообщения извне на другие удалённые серверы. Со спамом, адресованным локальным получателям, дело обстоит сложнее. Хорошо зарекомендовали себя служба *MAPS RBL* и ей подобные, организованные по принципу "чёрного списка" IP-адресов; чтобы задействовать эти сервисы, предварительно ознакомившись с условиями их использования, занесите имена доменов, работающих по принципу RBL, в конфигурацию:

```
smtpd_client_restrictions = permit_mynetworks, reject_maps_rbl
maps_rbl_domains = relays.ordb.org, blackholes.mail-abuse.org
```

В некоторых случаях требуется адресная работа с отдельными нарушителями почтового этикета. Адресная работа заключается в блокировании SMTP-соединений с их адресов, сетей либо доменов. Для этого предусмотрены таблицы типа `access` (см. access(5)):

```
smtpd_client_restrictions = permit_mynetworks, hash:/etc/postfix/access
```

Пример таблицы:

```
1.2.3.4 550 No more canned meat, please
1.2.5 REJECT
goodguy.generallybad.com OK
.generallybad.com REJECT
```

Как и с другими таблицами, после редактирования приведите кары в действие командой **postmap /etc/postfix/access**.

Прочие настройки

По умолчанию размер файла почтового ящика при локальной доставке ограничен `51200000` байтами. Это ограничение можно изменить с помощью параметра *mailbox_size_limit*. Установка параметра в `0` снимает ограничение.

Использование Postfix

После того, как Postfix настроен и запущен как сервис с предсказуемым именем **postfix**, в настройках почтовых клиентов можно указывать имя или адрес машины (например, `localhost`) как

SMTP-сервер. Программа **fetchmail** работает в связке с Postfix, опрашивая внешние почтовые ящики пользователей по протоколам POP3 или IMAP и передавая полученные сообщения системному MTA для локальной доставки. Лог-файлы Postfix находятся в каталоге `/var/log/mail`.

Глава 8. Объединенная служба каталога

Что такое служа каталога и что такое LDAP?

Служба каталога (Directory Service) – это программный комплекс для хранения и каталогизирования информации. По своей сути это очень похоже на обычную базу данных, но с "уклоном" скорее на чтение данных, нежели на их добавление или модификацию. Обычно служба каталога базируется на клиент-серверной архитектуре. Одна из наиболее известных таких систем – это DNS (Domain Name Service): DNS-сервер производит взаимную "трансляцию" имен машин и их IP-адресов. Другие машины в сети могут обращаться к такому серверу за информацией о соответствии имени и адреса. Однако это очень простой пример каталогизации информации. Объекты в такой базе имеют ограниченное количество атрибутов – таких как имя, адрес и еще несколько дополнительных параметров. Разумеется, служба каталога реального предприятия будет содержать более разнообразные данные и иметь гораздо более сложную структуру.

В общем случае служба каталога должна предоставлять простой, централизованный доступ к данным, которые могут использоваться различными приложениями. Протокол, по которому могла бы работать такая служба, был разработан в ISO (International Standartization Organization), получил номер X.500 и назывался DAP (Directory Access Protocol). В соответствии с этим протоколом любое приложение может получить доступ к информации в каталоге. Там же была предложена гибкая и легко расширяемая информационная структура которая позволяла хранить в принципе любой тип данных. К сожалению, X.500 имел и ряд ограничений, одним из которых была зависимость от коммуникационного уровня, который не являлся стандартным протоколом TCP и запутанность требований к правилам именования объектов. В результате решение на базе этого протокола становилось очень дорогим при обслуживании.

Позже появился протокол LDAP (Lightweight Directory Access Protocol), который позволил реализовать доступ по TCP/IP и мог легко расширяться. В результате появилось решение, позволяющее организовать службу каталога на предприятии любого масштаба.

Сегодня существует несколько реализаций данного протокола от различных фирм. Наиболее известные из них – это Netscape Directory Service, Microsoft Active Directory, Novell Directory Service. Из некоммерческих реализаций LDAP наибольшее распространение получил проект OpenLDAP. Именно его мы и будем рассматривать в данной главе, хотя большинство понятий и определений применимо и к другим реализациям сервера LDAP.

Основные понятия

Для понимания работы службы каталога необходимо усвоить несколько ключевых понятий.

- Данные каталога хранятся в виде объектов или записей (от англ. *entry*), состоящих из специальных полей называемых *атрибутами* (*attributes*). Набор атрибутов, их синтаксис и правила заполнения определяются *схемой каталог*а (*scheme*).
- Данные в каталоге можно представить в виде древовидной структуры – DIT (Directory Information Tree). Это очень похоже на структуру, используемую многими файловыми системами.

- Каждый объект в структуре каталога идентифицируется специальным атрибутом DN (Distinguished Name). По аналогии с файловой системой DN описывает путь, по которому можно найти объект в дереве каталога. Отличие в данном случае в том, что DN формируется не слева направо, как путь к файлу, а наоборот – справа налево.

- Любой DN в дереве должен заканчиваться специальным DN, называемым *суффиксом каталога* (*suffix*) и являющимся корнем дерева.

- *Корневой объект* (*Root Entry*) – это первый элемент дерева. DN корневого объекта полностью соответствует суффиксу.

- *DN администратора каталога* (*Root Distinguished Name*) – это специальный объект, описывающий администратора каталога. Обычно такой объект не имеет суффикса и к нему не применяются списки доступа (ACL).

- *База поиска* (*Base Distinguished Name*) – объект каталога, начиная с которого производится поиск. Дело в том, что не всегда есть необходимость производить поиск по всему дереву каталога; ограничить область поиска можно указнием в запросе базы поиска. По умолчанию этот параметр соответствует суффиксу.

Объекты и атрибуты

Серверы LDAP могут поставляться с несколькими вариантами *бэкенда* (*backend*). Например, OpenLDAP имеет такие варианты, как LDBM – собственный формат хранения данных в текстовых файлах; SHELL – интерфейс к базе данных, использующий команды UNIX; PASSWD – простейшая база, использующая стандартные файлы `/etc/passwd` и `/etc/group`; SQL – интерфейс к любой базе данных, использующей SQL.

Для процедур импорта и экспорта всеми серверами LDAP поддерживается единый формат обмена данными – LDIF. Вот пример такого файла:

```
dn: dc=altlinux,dc=ru
objectClass: top
objectClass: organization
o: ALTLinux Team
o: altlinux.ru
dn: ou=People,dc=altlinux,dc=ru
objectClass: top
objectClass: organizationalUnit
ou: People
description: ALT workers
description: Stuff area
```

Описание каждого объекта в таком файле начинается с атрибута DN, который идентифицирует данный объект в каталоге. Специальный атрибут `objectClass` указывает, к каким классам относится данный объект и, следовательно, какие атрибуты он может иметь. В нашем случае принадлежность к классу top означает, что объект обязательно должен иметь атрибут `objectClass`, а принадлежность к классу `organization` предполагает наличие нескольких атрибутов, из которых атрибут o является обязательным. Второй объект находится на одну ступеньку ниже по иерархии и поэтому в его DN включен DN объекта верхнего уровня.

Классы, характеризующие объекты, описываются схемой базы – ее пример приведен ниже.

```
attributetype ( 2.5.4.10 NAME ( 'o' 'organizationName' )
  SUP name
)
attributetype ( 2.5.4.13 NAME 'description'
```

```
  EQUALITY caseIgnoreMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{1024}
)
objectclass ( 2.5.6.4 NAME 'organization' SUP top STRUCTURAL
  MUST o
  MAY ( userPassword $ searchGuide $ seeAlso $ businessCategory $
    x121Address $ registeredAddress $ destinationIndicator $
    preferredDeliveryMethod $ telexNumber $ teletexTerminalIdentifier $
    telephoneNumber $ internationaliSDNNumber $
    facsimileTelephoneNumber $ street $ postOfficeBox $ postalCode $
    postalAddress $ physicalDeliveryOfficeName $ st $ l $ description
  )
)
```

В данном фрагменте приводятся описания двух атрибутов и одного класса. Вот что означают эти записи:

- Атрибут `o` (его можно также называть `organizationName`) является расширением атрибута `name`.
- Атрибут `description` – это строка длиной до 1024 байт; при поиске в ней регистр символов не учитывается.
- Класс `organization` является расширением класса `top` и имеет единственный обязательный атрибут `o`. Кроме того имеется большое количество необязательных атрибутов таких как `userPassword`, `businessAddress`, `street`, `postOfficeBox` и т.д.

Много полезной информации о схемах можно найти по ссылкам на сайте OpenLDAP [http://www.openldap.org/faq/data/cache/219.html].

Установка и настройка

Процесс сборки и установки сервера OpenLDAP не отличается от сборки и установки другого программного обеспечения, поставляемого с исходными кодами. Кроме того, практически во всех современных дистрибутивах Linux он поставляется в виде готового пакета. Поэтому уделим больше внимания настройке.

Настройка сервера

Сервер LDAP состоит из двух серверных процессов **slapd** и **slurpd**. Процесс **slapd** занимается приемом и обработкой запросов от клиентов; это основной процесс, который непосредственно работает с базой данных. Сервис **slurpd** используется в тех случаях, когда данные нужно реплицировать на другие сервера – он контролирует изменения в базе и при необходимости пересылает их на подчиненные сервера.

Приведем пример конфигурационного файла:

```
include /etc/openldap/schema/core.schema
include /etc/openldap/schema/nis.schema
```

В первых строках мы подключаем необходимые схемы; в поставке OpenLDAP их около полутора десятков. Подключите только те, которые будете использовать. В принципе, схемы являются частью конфигурационного файла, но для наглядности они вынесены в отдельные фрагменты.

```
database ldbm
```

В качестве способа хранения используется собственный формат ldbm. Если предполагается обычная конфигурация сервера, то данный формат предпочтителен.

```
suffix "dc=altlinux,dc=ru"
```

Корнем информационной структуры будет являться объект `dc=altlinux,dc=ru`. В принципе, суффикс для каталога можно взять любой, например, `o=ALTLinux,c=RU` – это не накладывает абсолютно никаких ограничений на функциональность. Однако последнее время все чаще используется именно первый вид суффикса, который подчеркивает, что информационная структура данного предприятия тесно связана со структурой его домена.

```
rootdn "cn=admin,dc=altlinux,dc=ru"
rootpw secret
```

DN, описывающий администратора базы данных и пароль. В данном случае пароль записан в открытом виде, поэтому файл конфигурации сервера должен иметь соответствующие права доступа, ограничивающие его чтение обычными пользователями. Пароль можно записать и в виде хэша DES или MD5 – тогда строка будет иметь вид

```
rootpw $1$slJFPHzI$x2hWBQDHqzvMaziAoq2bq/
index objectClass eq
```

Формат ldbm поддерживает простейшие индексы с целью ускорения операций поиска. Желательно создать такие индексы по тем атрибутам, по которым предполагается наибольшее количество обращений.

```
access to attr=userPassword
  by self write
  by anonymous auth
  by * none

access to * by * read
```

Не всегда данные каталога находятся в публичном доступе. Для управления доступам могут использоваться *списки доступа* (*access lists*). В данном примере приводятся два списка – в первом из них ограничивается доступ к атрибуту `userPassword` (полный доступ к нему могут иметь только сам объект либо администратор базы; для всех остальных доступ запрещен). Второе правило гласит, что всем дается доступ на чтение любых данных (кроме ограниченного предыдущим правилом).

```
TLSCipherSuite HIGH:MEDIUM:+SSLv2
TLSCertificateFile /etc/openldap/ssl/slapd.pem
TLSCertificateKeyFile /etc/openldap/ssl/slapd.pem
```

Часто LDAP используется для централизованной авторизации пользователей сети. В таких случаях из каталога может запрашиваться конфиденциальная информация, например, пароль. Для предотвращения перехвата этих данных желательно использовать протокол LDAPS (LDAP с SSL/TLS).

После настройки можно сразу запустить процесс **slapd** – например, такой командой:

```
slapd -u ldap -h ldap://127.0.0.1/ ldaps://ldap.altlinux.ru/
```

Первые объекты, которые нужно создать в базе – это *корневой элемент* (*root entry*) и *администратор базы* (*root dn*), которые указаны в конфигурационном файле как `suffix` и `rootdn`.

Настройка репликации

Одной из важных особенностей LDAP являются встроенные средства репликации данных. Этот механизм реализован в виде отдельного серверного процесса, контролирующего изменения в базе данных и пересылающего эти изменения на другие сервера. Прежде чем включать такую репликацию, необходимо убедиться, что соответствующие данные на обоих серверах идентичны. Это связано с тем, что **slurpd** пересылает именно изменения на текущем сервере – он не проверяет и не анализирует состояние данных на удаленном сервере. Настройки **slurpd** находятся в том же файле, что и настройки **slapd** – поэтому перечислим, что нужно добавить к перечисленным выше параметрам:

```
replica /var/log/slapd.replog
```

Прежде всего укажем файл, в который **slapd** будет записывать все свои действия и из которого **slurpd** будет их читать.

```
replica
  host=ldap2.altlinux.ru
  tls=yes
  bindmethod=simple
  binddn="cn=slurpd,ou=lug,dc=altlinux,dc=ru"
  credentials=secret
```

Для каждого подчиненного сервера описывается так называемая реплика. На вторичном сервере нужно создать соответствующий объект и указать, что он имеет права на изменение информации с помощью параметров `updatedn` и `updateref`.

Настройка клиента

Существует огромное количество клиентов, работающих с LDAP. Это могут быть почтовые программы, которые обращаются к каталогу в поисках адреса электронной почты сотрудника или за информацией о маршрутизации почты, FTP-сервер, который берет информацию для авторизации своего клиента и многие другие программы – однако все они имеют схожие настройки. Прежде всего это адрес сервера и порт, на котором работает LDAP (обычно это 389 либо 636, если сервер поддерживает протокол LDAPS). Вторым важным параметром является база поиска (Base DN) – в большинстве случаев этот параметр соответствует суффиксу сервера. Третий важный параметр – фильтр поиска. Кроме того, существуют параметры, позволяющие ограничить поиск снизу – например, только самой базой или базой и ее подобъектами первого уровня, управляющие поиском в алиасах (alias) и т.п.

Трех этих параметров в большинстве случаев достаточно, чтобы выполнить запрос к любому серверу LDAP. Однако если на сервере существуют ограничения на доступ к данным, то может потребоваться авторизация. Авторизоваться в LDAP можно, указав DN одного из объектов базы данных LDAP; пароль для такого объекта будет искаться в его атрибуте `userPassword`.

Ниже приводится фрагмент настройки почтового сервера Postfix:

```
canonical_maps = ldap:canonical
canonical_server_host = localhost
canonical_search_base = ou=People,dc=altlinux,dc=ru
canonical_query_filter = (&(uid=%s)(objectClass=posixAccount))
canonical_result_attribute = mail
canonical_scope = sub
canonical_bind = yes
canonical_bind_dn = cn=mta,dc=altlinux,dc=ru
canonical_bind_pw = supersecret
```

В данном фрагменте описывается, что в процессе канонизации почтового адреса необходимо сделать запрос в LDAP и найти данные по атрибуту `mail` для объекта, у которого `uid` соответствует искомой строке и который относится к классу `posixAccount`. Поиск ограничивается только объектом

ou=People,dc=altlinux,dc=ru и его подобъектами первого уровня. Для того, чтобы получить доступ к этим данным, необходимо авторизоваться как объект cn=mta,dc=altlinux,dc=ru, используя пароль supersecret.

Использование LDAP

LDAP может использоваться в самых различных ситуациях; здесь мы рассмотрим несколько наиболее распространенных применений. Поскольку в LDAP хранится полная информация об сотрудниках предприятия, мы можем брать справочную информацию для почтовых программ прямо оттуда. Для начала настроим сервер:

```
include /etc/openldap/schema/core.schema
include /etc/openldap/schema/cosine.schema
include /etc/openldap/schema/inetorgperson.schema
include /etc/openldap/schema/nis.schema

TLSCipherSuite HIGH:MEDIUM:+SSLv2
TLSCertificateFile /etc/openldap/ssl/slapd.pem
TLSCertificateKeyFile /etc/openldap/ssl/slapd.pem

pidfile /var/run/slapd.pid
argsfile /var/run/slapd.args
directory /var/lib/ldap
database ldbm
index objectClass,uid,uidNumber,gidNumber eq
index cn,name,surName,givenName eq,subinitial
password-hash {MD5}

suffix "dc=altlinux,dc=ru"
rootdn "cn=admin,dc=altlinux,dc=ru"
rootpw {md5}$1$ION4SIII$EYyGEeYt4g2hEe9tjICac.

access to attr=userPassword
  by self write
  by anonymous auth
  by * none
access to attr=shadowLastChange
  by self read
  by anonymous auth
  by * none
access to attr=shadowFlag
  by self read
  by anonymous auth
  by * none
access to attr=shadowMax
  by self read
  by anonymous auth
  by * none
access to attr=shadowMin
  by self read
  by anonymous auth
  by * none
access to attr=shadowWarning
  by self read
  by anonymous auth
  by * none
access to attr=shadowInactive
  by self read
  by anonymous auth
```

```
  by * none
access to attr=shadowExpire
  by self read
  by anonymous auth
  by * none
access to * by * read
```

После этого создадим пользователя `ldap`, от имени которого будет работать наш сервер, сертификат с помощью программы **openssl** и запустим процесс **slapd** следующей командой:

```
# slapd -u ldap -h 'ldap://127.0.0.1/ ldap//ldap.altlinux.ru/ ldaps://ldap.altlinux.ru'
```

Тепрь можно создать базу данных – например, с помощью утилиты **ldapadd**:

```
$ ldapadd -xWD cn=admin,dc=altlinux,dc=ru -H ldaps://ldap.altlinux.ru -f initial.ldif
```

Содержимое файла initial.ldif будет такое:

```
dn: dc=altlinux,dc=ru
objectClass: top
objectClass: organization
o: ALTLinux Team
o: altlinux.ru

dn: cn=admin,dc=altlinux,dc=ru
objectClass: top
objectClass: organizationalRole
cn: admin
description: ALTLinux LDAP manager

dn: ou=People,dc=altlinux,dc=ru
objectClass: top
objectClass: organizationalUnit
ou: People
description: Stuff area

dn: uid=migor,ou=People,dc=altlinux,dc=ru
objectClass: top
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
objectClass: inetOrgPerson
cn: Igor Muratov
sn: Muratov
givenName: Igor
uid: migor
uidNumber: 1000
gidNumber: 1000
homeDirectory: /home/migor
loginShell: /bin/bash
userPassword: {md5}$1$ION4SIII$EYyGEeYt4g2hEe9tjICac.
mail: migor@altlinux.ru
mail: migor@linux.ru.net

....

dn: ou=Group,dc=altlinux,dc=ru
objectClass: top
objectClass: organizationalUnit
ou: Group
description: Groups of users
```

```
dn: cn=luser,ou=Group,dc=altlinux,dc=ru
objectClass: top
objectClass: posixGroup
cn: luser
gidNumber: 1000
description: Default group for users presented by LDAP
```

Проверим, что сервер работает сделав к нему анонимный запрос:

```
$ ldapsearch -xLLL "(uid=migor)"
dn: uid=migor,ou=People,dc=altlinux,dc=ru
objectClass: top
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
objectClass: inetOrgPerson
cn: Igor Muratov
sn: Muratov
givenName: Igor
uid: migor
uidNumber: 1000
gidNumber: 1000
homeDirectory: /home/migor
loginShell: /bin/bash
mail: migor@altlinux.ru
mail: migor@linux.ru.net
```

Поскольку, согласно нашим настройкам, доступ к атрибуту `userPassword` имеют только сам пользователь и администратор, то этот атрибут мы не получили. Собственно, пока он нам и не нужен.

Адресная книга

На сегодняшний день почти все почтовые программы поддерживают возможность использовать LDAP как адресную книгу. В качестве примера возьмем пакет Mozilla; установите пакеты libldap, mozilla, mozilla-mail и запустите программу, далее:

- откройте окно настройки (Edit->Preferences...);
- выберите слева категорию Mail & Newsgroups, подкатегорию Addressing;
- справа в опциях Address Autocompletion включите Directory Server и нажмите кнопку Edit Directories...;
- в новом окне нажмите кнопку Add и на вкладке General заполните поля Name: `ExampleLDAP` , Hostname: `ldap.altlinux.ru` и BaseDN: `dc=altlinux,dc=ru`;
- при желании на вкладке Advanced можно указать ограничение на количество возвращаемых записей (по умолчанию это 100) и фильтр поиска.

После этого сохраните изменения – и теперь при заполнении поля To: можно писать не адрес, а имя получателя из атрибута `cn`. Программа произведет соответствующий поиск и предложит варианты атрибута `mail`, которые найдет в базе.

Для настройки другого пакета обратитесь к руководству пользователя вашей программы.

Маршрутизация почты в Postfix

Предположим, что наше предприятие не имеет своего POP3/IMAP-сервера либо для некоторых сотрудников удобнее получать почту через другой сервер. Для этого нам необходимо принять почту пользователя, приходящую в наш домен, и переправить ее на тот адрес который для сотрудника удобнее. Решений для этой задачи существует несколько: в простейшем варианте можно создать в домашнем каталоге пользователя файл `.forward`, в котором он сам мог бы указать нужный ему адрес. Однако усложним задание – предположим, что на нашем почтовом сервере нет учетной записи для данного пользователя; тогда получается, что этот файл некуда поместить. Второй вариант – настроить пересылку на самом сервере; для этого создается файл `/etc/postfix/virtual` приблизительного такого вида:

```
migor@altlinux.ru migor@linux.ru.net
```

а в конфигурационном файле Postfix указывается

```
virtual_maps = hash:/etc/postfix/virtual
```

Теперь остается только создать хэш и перезапустить Postfix; однако, если мы имеем много таких пользователей и если почтовых серверов существует несколько, то отслеживать синхронное изменение файлов `/etc/postfix/virtual` становится нелегкой задачей.

Немного модифицируем наше последнее решение. Перенесем данные из файла `/etc/postfix/virtual` в LDAP; для этого модифицируем приведенную выше базу следующим образом: добавим пользователю класс `inetLocalMailRecipient` и новый атрибут `mailRoutingAddress`. Атрибут `mail` у нас теперь имеет только одно значение.

```
dn: uid=migor,ou=People,dc=altlinux,dc=ru
changetype: modify
objectClass: top
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
objectClass: inetOrgPerson
objectClass: inetLocalMailRecipient
mail: migor@altlinux.ru
mailRoutingAddress: migor@linux.ru.net
После этого изменим настройки Postfix:
virtual_maps = ldap:virtual
virtual_server_host = ldap.altlinux.ru
virtual_search_base = ou=People,dc=altlinux,dc=ru
virtual_query_filter = (&(mail=%s)(objectClass=inetLocalMailRecipient))
virtual_result_attribute = mailRoutingAddress
virtual_scope = sub
```

Теперь почта для данного пользователя будет пересылаться на адрес из атрибута `mailRoutingAddress`, тем не менее в адресной книге все останется без изменений и там будет показываться “официальный” адрес пользователя из атрибута `mail`.

Централизованная авторизация

Разобравшись с почтой, хочется перенести в LDAP и авторизацию. Обычно для этих целей используют NIS+, однако хочется использовать для этого более совершенную технологию – в конце концов, у нас уже есть LDAP-сервер, содержащий все необходимые данные по нашим пользователям. Для того, чтобы система искала своих пользователей не только в файле `/etc/passwd`, необходимо установить пакеты `nss_ldap` и `pam_ldap`. Оба пакета имеют общий конфигурационный файл `/etc/ldap.conf` (в других дистрибутивах это могут быть другие файлы, но синтаксис у них одинаковый).

```
uri ldaps://ldap.altlinux.ru
ldap_version 3
base dc=altlinux,dc=ru
rootbinddn cn=admin,dc=altlinux,dc=ru
timelimit 15
ssl on
```

Поскольку нам необходимо получить доступ к атрибуту пароля userPassword, потребуется авторизованный доступ. Для этого укажем опцию rootbinddn, а в файле /etc/ldap.secret запишем пароль администратора базы. Затем подправим файл `/etc/nsswitch.conf`:

```
passwd: file ldap
shadow: file ldap
group: file ldap
```

Теперь проверяем, подключены ли пользователи из базы:

```
$ id migor
uid=1000(migor) gid=1000(luser) groups=1000(luser)
```

Обратите внимание на то, что сейчас мы обращаемся к серверу по защищенному протоколу LDAPS. Поскольку теперь мы берем из базы крайне важную информацию – пароль пользователя, дополнительная степень защиты будет весьма кстати.

Приложения

Ссылки

Список ссылок на информационные ресурсы Internet, посвященных LDAP:

- University of Michigan LDAP Page [http://www.umich.edu/dirsvcs/ldap/index.html]
- University of Michigan LDAP Documentation Page [http://www.umich.edu/dirsvcs/ldap/doc/]
- OpenLDAP Administrator's Guide [http//www.openldap.org/doc/admin]
- Manually Implementing Roaming Access [http://help.netscape.com/products/client/communicator/manual_roaming
- Customizing LDAP Settings for Communicator 4.5 [http://developer.netscape.com/docs/manuals/communicator/ldap
- Introducing to Directory Service (X.500) [http://www.nic.surfnet.nl/surfnet/projects/x500/introducing/]
- Linux Directory Service [http://www.rage.net/ldap/]

RFC

Список RFC, поддерживающих LDAP:

- RFC 1558: A String Representation of LDAP Search Filters
- RFC 1777: Lightweight Directory Access Protocol
- RFC 1778: The String Representation of Standard Attribute Syntaxes
- RFC 1779: A String Representation of Distinguished Names
- RFC 1781: Using the OSI Directory to Achieve User Friendly Naming
- RFC 1798: Connectionless LDAP
- RFC 1823: The LDAP Application Programming Interface
- RFC 1959: An LDAP URL Format
- RFC 1960: A String Representation of LDAP Search Filters
- RFC 2251: Lightweight Directory Access Protocol (v3)
- RFC 2307: LDAP as a Network Information Service

Глава 9. Служба FTP

FTP, второй по популярности в интернет протокол после HTTP, предназначен для обмена файлами. Он предназначен только для передачи файлов, зато делает это хорошо. К сожалению, изначально протокол спроектирован так, что пароли, данные и все прочее передаются открытым текстом и их можно легко перехватить – однако большинство серверов предоставляют только анонимный доступ, так что это не проблема.

В этом документе изложены рекомендации, которые помогут вам правильно настроить FTP-сервер и свести к минимуму угрозу проникновения в систему злоумышленников через этот вид сервиса.

Установка анонимного FTP-сервера

При установке FTP-серверов, входящих в комплект поставки дистрибутива ALT Linux Master, автоматически регистрируется в системе новый пользователь, от имени которого будет происходить работа FTP-сервера. Бюджет ftp не должен использоваться кем-либо для входа в систему, поэтому реальный пароль для него не задаётся; группа пользователя также не имеет каких-либо прав на файловой системе. Домашним каталогом для него устанавливается `/var/ftp`, а вместо командного интерпретатора указывается `/dev/null`. Таким образом, бюджет пользователя ftp, в файле `/etc/passwd`, имеет примерно следующий вид:

```
ftp:x:14:50:FTP User:/var/ftp:/dev/null
```

Итак, после того, как FTP-сервер установлен, можно обратить внимание и на его рабочий каталог. Домашним каталогом пользователя ftp является `~ftp` – это полный путь к каталогу, который будет “корневым” для всех анонимных пользователей. В нашем случае, как вы уже догадались, это `/var/ftp`. Его можно создать как автоматически, установив пакет `anonftp`, так и вручную. Владельцем этого каталога будет пользователь root, группа ftpadmin. Да-да, именно он, а не ftp. Это делается в целях вашей же (FTP-сервера и системы в целом) безопасности – иначе однажды вы можете узнать неприятную новость, что больше не являетесь хозяином в своей системе.

Для нормального функционирования будущего сервера неплохо было бы создать дерево подкаталогов, в которых будут размещаться файлы. Минимально, что вам может понадобиться, это `~ftp/pub`. Для него необходимо установить права доступа 755. Владельцем для этого каталога также будет root. Это делается с той целью, чтобы содержимое `~ftp/pub` было доступно всем пользователям FTP-сервера для беспрепятственного (разумеется, в рамках дозволенного) использования тех файлов, которые вы хотите сделать публично-доступными. Группу, которой принадлежит `~ftp/pub`, лучше сменить с root на специальную, в которую включены пользователи, имеющие право изменения содержимого этого каталога – этим не стоит заниматься от имени root.

Для того, чтобы разрешить пользователям вашего сервера доступ на запись, создайте каталог `~ftp/pub/incoming` с правами доступа 733 (владелец – root), тем самым предоставив право записи в этот каталог, но лишив возможности просмотра его содержимого.

Замечание

Совет: для предотвращения атаки на ваш сервер через ftp, путём переполнения диска информацией, с целью заблокировать работу всей системы, старайтесь размещать каталог `~ftp/pub/incoming` на отдельном разделе файловой системы.

Особенности FTP-сервера vsftpd из поставки ALT Linux Master

В комплект поставки дистрибутива входит `vsftpd` (Very Secure FTP Daemon) – полнофункциональный FTP-сервер, позволяющий обслуживать как анонимные запросы, так и запросы от пользователей, имеющих полноценный доступ к ресурсам сервера.

Разумеется, “Very Secure” в его названии не является гарантией, однако свидетельствует о том, что при написании кода целью стояло создание максимально безопасной и аккуратно выполненной программы, минимально чувствительной к атакам со стороны.

Если вам необходим анонимный FTP-сервер, то вам подойдет `vsftpd` в сочетании с пакетом `anonftp`. Этот пакет содержит дерево каталогов, необходимых для организации сервера с анонимным доступом к ресурсам, который не требует особо изощрённой настройки, если только вы не захотите, например, предоставить пользователям доступ на запись. Этот сервер в состоянии осуществлять всю передачу данных в пассивном режиме, что в высшей степени безопасно, однако не всегда удобно.

Если же вам необходим надёжный, защищённый и одновременно чрезвычайно быстрый и масштабируемый FTP-сервер, предоставляющий не только анонимный доступ к ресурсам вашего сервера, но и доступ для пользователей, зарегистрированных локально, то и в этом случае вам безусловно подойдёт `vsftpd`. Примером такого использования может послужить серверный пул ftp.redhat.com, обрабатывающий по 15000 соединений одновременно.

Что же придаёт ему такую популярность? Во-первых, конечно же безопасность работы. Каждая строка его кода неоднократно подвергалась самым жестким проверкам со стороны специалистов в вопросах безопасности. Другой стороной его привлекательности безусловно является простота и гибкость настройки. Все необходимые настройки осуществляются путем редактирования единственного конфигурационного файла `/etc/vsftpd.conf`.

В целях безопасности по умолчанию сервер сконфигурирован для предоставления только анонимного доступа. Запрещены любые команды записи. От администратора при установке требуется только удалить знак комментария перед директивой `nopriv_user`, которая задаёт имя непривилегированного пользователя, используемое `vsftpd` для организации безопасных соединений. Это должен быть абсолютно изолированный и лишённый каких-либо привилегий пользователь, наподобие ftp, описанного ранее. С этой целью `vsftpd` при установке автоматически регистрирует бюджет пользователя novsftpd.

Для предоставления доступа к FTP-серверу локально зарегистрированным пользователям необходимо внести изменения в файл конфигурации `/etc/vsftpd.conf`, который на самом деле является ссылкой на файл `/etc/vsftpd/conf`. Для этого достаточно удалить знак комментария перед директивой `local_enable=YES` – тем самым предоставляя доступ пользователям к их домашним и системным каталогам. Для ограничения возможности пользователя перемещаться по дереву каталогов достаточно убрать знак комментария из строки, содержащей директиву `chroot_list_file`, чтобы сообщить `vsftpd` о необходимости использования изолированной среды выполнения для пользователей, зарегистрированных локально. Теперь, в случае успешной регистрации пользователя в системе, “свобода” его перемещения по каталогам будет ограничена в пределах только его “домашнего” каталога. Указание регистрационного имени пользователя в файле `/etc/vsftpd/chroot_list`, позволит сделать исключение для пользователя, чьё имя указано в этом файле, предоставив ему свободу перемещения.

Общие рекомендации

Еще большей безопасности в работе FTP-сервера можно добиться при помощи `xinetd`. Этот сервер позволяет ограничить количество одновременно выполняемых процессов как по системе в целом, так и для каждого отдельного пользователя, указать пользователя, от имени которого будет выполняться сервис, задать приоритет процесса (nice), указать адреса, с которых разрешено

подключение к данной службе, а также время доступа и множество других параметров. Вот наглядный пример файла конфигурации `xinetd` для `vsftpd`:

```
# default: off
 # description: The vsftpd FTP server.
 service ftp
 {
      disable = no # включает службу
      socket_type = stream
      protocol = tcp
      wait = no
      user = root
      nice = 10
      rlimit_as = 16M # устанавливает лимит адресного пространства
      server = /usr/sbin/vsftpd # путь к исполняемому файлу
      only_from = 192.168.0.0 # предоставляем доступ из всей подсети 192.168.0
      only_from = 207.46.197.100, 207.46.197.101 # доступ с указанных адресов
 #     only_from = 0.0.0.0 # неограниченный по адресам доступ
      access_times = 2:00-9:00 12:00-24:00 # время, когда возможен доступ
 }
```

Для получения дополнительной информации по использованию `xinetd` смотрите страницы руководства `xinetd` и `xinetd.conf`.

Глава 10. Samba

Олег Власенко

<cornet@altlinux.ru>

Игорь Вергейчик

<i.viarheichyk@sam-solutions.net>

Александр Боковой

<ab@altlinux.ru>

История переиздания
Издание 0.2 16 Nov 2002
Изменения в структуре документа - выделение раздела для клиентской части и перенос разделов по безопасности и локализации из серверного раздела. Мелкие багфиксы.
Издание 0.1 26 May 2002
Начальная версия.

Аннотация

Данный раздел документации предназначен прежде всего для тех, кто только начинает знакомиться с Samba, но между тем уже имеет достаточные знания в области TCP/IP и сетей Microsoft™.

Все, что сказано ниже, относится непосредственно к пакету `samba-2.2.5` , входящему в состав ALTLinux Master; тем не менее, многое будет справедливо как для предыдущих, так и для последующих версий.

Общие сведения о Samba

Данный продукт представляет собой комплект серверного и клиентского программного обеспечения для осуществления связи UNIX-машин с сетями Microsoft™ и LanManager, которые сами по себе представляют собой подклассы [8] сетей SMB.

Исходно сети SMB были разработаны фирмой IBM™, базировались на протоколе NetBIOS, предназначались прежде всего для сетей Token Ring и были в полной мере реализованы в OS/2 Warp LanServer. Позднее в Windows 95 этот протокол был заменён на NetBEUI (несколько упрощённая версия NetBIOS).

Чуть ранее в OS/2 Warp и NT 3.5 была реализована более удобная для сложных гетерогенных сетей реализация, работающая поверх TCP/IP — “NetBIOS over TCP/IP”. Ввиду явных преимуществ данного подхода он используется и поныне. Когда где-либо в Windows вы организовываете работу с сетевыми разделяемыми ресурсами по TCP/IP, то на самом деле используется “NetBIOS over TCP/IP” (о чем, например, в Win95 в свойствах TCP/IP в закладке NetBIOS есть соответствующая отметка).

Samba также использует протокол “NetBIOS over TCP/IP”, что позволяет ей успешно взаимодействовать с такими реализациями SMB, как входящие в OS/2 3-4, Windows 9X-ME, NT3.5-4/2000/XP, UNIX-системами с Samba и, возможно, другими подобными. Менее очевидно то, что

[8] На самом деле Microsoft™ существенно расширила исходную спецификацию SMB.

Samba не может работать без использования TCP/IP (на NetBIOS и NetBEUI). Об этом не стоит забывать при проектировании сетей.

Итак, для работы в сетях SMB необходимы:

- клиент;
- сервер;
- средства администрирования.

Все это есть в пакетах `samba-client`, `samba-client-cups`, `samba-common`, `samba`, `samba-swat`, входящих в состав дистрибутива.

При использовании SMB доступны следующие ресурсы:

- сетевые диски;
- прямые пути к дискам;
- принтеры;
- доменная авторизация и управление.

Первые три пункта поддерживаются Samba в полном объёме, последний — частично, но это направление стремительно развивается и весьма полно реализовано в Samba 3.0, описанной ниже.

Также доступен весьма объёмный комплект документации в пакете `samba-doc`; большинство ссылок данного раздела будут указывать именно на содержимое этого пакета.

Краткий обзор каталогов и файлов

Все файлы конфигурации и авторизации Samba расположены в каталоге `/etc/samba` и его под-каталогах. Рассмотрим их несколько подробнее.

`MACHINE.SID` системный идентификатор машины, формируется автоматически при старте сервера и предназначен для идентификации компьютера в домене сети Microsoft™;

`codepages/` каталог, содержащий файлы с таблицами перекодировки;

`lmhosts` то же, что и `/etc/hosts`, но предназначен для преобразования IP<=>NetBIOS. Как правило содержит только одну запись:

```
127.0.0.1 localhost
```

но можно считать удачной идеей [9] заносить туда хосты из других под-сетей (когда по ряду причин невозможно надёжно провести преобразование IP<=>NetBIOS ни широковещательными запросами, ни с использованием WINS) или наоборот — ключевые сервера собственного домена;

[9] Как и с `/etc/hosts`, увлекаться содержанием распределённых данных в локальных файлах не стоит.

`secrets.tdb` — ключевой файл для идентификации машины в домене сети Microsoft™. С точки зрения безопасности имеет ту же ценность, что и файлы `/etc/tcb/*/shadow` — а потому права доступа должны быть `root.root 0600`;

`smb.conf` — основной конфигурационный файл Samba. Он нужен не только серверной части, но и всем остальным компонентам этой системы;

`smbpasswd` — аналог `/etc/passwd` и `/etc/tcb/*/shadow` — файл пользователей сервера Samba с паролями. С точки зрения безопасности имеет ту же ценность, что и `/etc/tcb/*/shadow` — а потому права доступа должны быть `root.root 0600`. Соответствие пользователей Samba и системных производится на основе общего UID; данный файл используется Samba при отсутствии данных о пользователе на PDC или при отсутствии самого PDC;

`smbusers` — файл соответствий имён сетевых и локальных пользователей SMB; это удобный метод для организации административных и гостевых входов на сервер. Соответствие пользователей Samba и системных производится на основе символьных имён;

`/var/log/samba/*` — лог-файлы серверной части Samba. Из них `log.smbd`, `log.nmbd`, `log.winbind` — журналы соответствующих процессов, а все прочие — логи взаимодействия сервера с отдельными клиентскими хостами в формате именования по умолчанию `log.<Client_NetBIOS_NAME>`. При превышении заданного в `smb.conf` предела производится ротация логов и формируются файлы `*.old`;

`/var/spool/samba` — каталог динамического спулинга печати сервера Samba. На не сильно загруженных серверах печати он обычно пуст; наличие там множества файлов в то время, когда ни один из клиентов не печатает — явный признак сбоев сервера печати;

`/var/cache/samba/*` — файлы (как правило, двоичные базы данных), формируемые в процессе работы различных компонентов Samba. Наиболее примечательны:

- `browse.dat` и `wins.dat` — текстовые файлы, их названия говорят сами за себя;
- `winbindd*.tdb` — базы данных доменных пользователей, формируемых winbind (см. “Использование winbind”). Время от времени их необходимо архивировать: если при апгрейде, “переезде” или переустановке сервера winbind сгенерирует эти файлы с нуля, то соответствия системных и доменных символьных и числовых имён изменятся и права доступа на восстановленные из архива файлы окажутся заведомо перепутанными. Поэтому настоятельно рекомендуется архивировать файлы `/var/cache/samba/winbindd*.tdb`;

`/var/lib/samba/*` служебные каталоги для администратора сервера.

Список выполняемых файлов Samba можно получить командой:

```
$ rpm -ql ‘rpm -qa | grep samba‘ | grep bin/
```

и подробно ознакомиться с каждым, прочитав соответствующие разделы документации.

Здесь же мы остановимся лишь на самых важных и наиболее часто используемых компонентах.

1. серверные компоненты:

   `/usr/sbin/nmbd` сервер преобразования имён и адресов;

   `/usr/sbin/smbd` файловый сервер;

   `/usr/sbin/winbindd` сервер импорта пользователей и групп с PDC;

   `/usr/sbin/swat` средство конфигурирования Samba с web-интерфейсом;

   `/etc/init.d/smb` и `/etc/init.d/winbind` управляющие скрипты инициализации сервисов.

   Следует отметить, что у скрипта `/etc/init.d/smb` есть два режима рестарта — `restart` и `reload`, которые радикально отличаются следующими особенностями:

   a. `restart` производит полный рестарт процессов **smbd** и **nmbd** со сбросом текущих соединений. Как правило, клиенты сами производят автоматический реконнект к ресурсам, однако если в момент рестарта были открыты файлы, то возможны проблемы с клиентскими приложениями (например, MS Office и 1C);

   b. `reload` заставляет **smbd** и **nmbd** только лишь перечитывать файлы конфигурации без рестарта и сброса соединений. При этом старые соединения продолжают существовать по старым правилам, а ко всем новым соединениям будут применены уже новые правила на основании файлов конфигурации.

2. клиентские компоненты:

   `/usr/bin/smbclient` интерактивное приложение для просмотра сетевых ресурсов;

   `/sbin/mount.smb`, `/sbin/mount.smbfs`, `/usr/bin/smbumount`, `/usr/sbin/smbmnt`, `/usr/bin/smbmount` средства монтирования/размонтирования сетевых файловых систем.

3. утилиты:

   `/usr/bin/smbpasswd` управление пользователями и подключением к домену;

   `/usr/bin/wbinfo` отображение списка пользователей, импортированных **winbindd**;

`/usr/bin/testparm`	проверка синтаксиса конфигурационных файлов;
`/usr/bin/smbstatus`	отображение статуса процессов **smbd** и **nmbd**;
`/usr/bin/nmblookup`	программа разрешения имён WINS (аналог **nslookup** для DNS).

Настройка сервера

В большинстве случаев настройка Samba заключается в редактировании основного конфигурационного файла `/etc/samba/smb.conf` и управлении пользователями с помощью **smbpasswd**. Если это непривычно — попробуйте использовать web-интерфейс SWAT (Samba Web Administration Tool); для этого установите пакет `samba-swat` и откройте URL `http://localhost:901/` в браузере.

Обычный сервер

Под таковым мы понимаем компьютер, предоставляющий в сеть файловые ресурсы. Фактически это простейший независимый файловый сервер, имеющий собственную базу авторизации пользователей.

Для того, что бы создать такой сервер, необходимо лишь немного подправить стандартный конфигурационный файл `smb.conf` (подставить требуемые имя рабочей группы и имена ресурсов) и создать учётные записи пользователей, как описано ниже, а также учесть рекомендации по безопасности, изложенные в конце параграфа.

Вот основные записи в smb.conf, которые создадут нам “обычный сервер”.

```
[global]

# Секция [global] определяет общие настройки серверной части Samba в
# целом для всех ресурсов.
# Имя рабочей группы OFFICE
workgroup = OFFICE

# Уровень определения прав доступа на уровне пользователей
security = user

# Приоритет данного сервера среди других компьютеров рабочей группы:
# определяет, кто именно будет главной машиной, отвечающей за
# отображение ресурсов сети. Для сравнения, у Win9X os level = 34, а
# у NT4 os level = 64.
os level = 65

# Очевидно, что раз нет домена - нет и мастера.
domain master = no

# Не стоит становиться сервером паролей для окрестных машин. Так что
# если к Вам прибежал разъярённый администратор соседнего NT-сервера с
# жалобами что его не пускают на его собственный сервер - поставьте
# domain logons = no ;-)
domain logons = no

# Обычно в простейшей сети WINS не нужен, мы его отключаем и у себя то
# же.
wins support = no
```

Ну а теперь надо определить, какие именно каталоги мы предоставим в сеть. Для каждого ресурса существует отдельная секция.

Самый простейший вариант для обычных ресурсов - обычный каталог с именем `public` [10] :

```
# имя ресурса, видимое в сети

[public]

# комментарий, видимый в сети как комментарий к ресурсу
comment = Public Stuff

# путь к каталогу ресурса
path = /home/samba/public

# отметка о доступе на чтение всем авторизованным пользователям (в том
# числе и гостевым, если они определены)
public = yes

# запрещение работы на запись всем пользователям
writable = no

# разрешение работы на запись всем пользователям, входящим в системную
# группу staff
write list = @staff
```

Подобным образом можно создать различные сетевые ресурсы сервера с различными правами доступа; за более подробной справкой по директивам и их синтаксису обратитесь к справочному руководству.

Поскольку Samba исполняется не в `chroot`, внутри ресурсов можно использовать любые символические ссылки на расположенные локально и в сети (NFS, SMB, Coda и т.д.) файловые объекты, что очень удобно в плане администрирования системы.

Особые ресурсы — например, домашние каталоги пользователей:

```
# имя ресурса, которое автоматически будет заменено именем
# домашнего каталога пользователя, под которым подключился клиент
# и именно название его домашнего каталога будет отображено в сети
# как имя ресурса.

# Для получения доступа к этому ресурсу клиент должен предоставить
# серверу соответствующие имя и пароль, все прочие пользователи к
# этому ресурсу доступа не имеют вовсе.

[homes]

# комментарий, видимый в сети как комментарий к ресурсу
comment = Home Directories

# признак невидимости - данный ресурс виден в сети только тому
# пользователю,который является его владельцем. К этому
# ресурсу можно обратиться непосредственно задав его имя, но в
# браузинге сети он будет виден только владельцу.
browseable = no
```

[10] Так называемая “файлопомойка” :-).

```
# Разрешение на запись.
writable = yes
```

Принтеры:

```
# имя ресурса, которое будет видно в сети. Кроме него, в сети будут
# также видны и локальные принтеры под теми же именами, что и в
# системе по команде lpq.

[printers]

# комментарий, который игнорируется.
comment = All Printers

# Путь к каталогу, в котором располагается спул принтеров,
# предоставляемых в сеть через Samba
path = /var/spool/samba

# невидимость ресурса в браузинге, он подменяется системным
# ресурсом.
browseable = no

# разрешение на печать для гостевого захода.
guest ok = yes

# запрещение на запись, поскольку в спул пишет сама Samba, а не
# пользователь.
writable = no

# признак того, что это именно принтер, а не файловый ресурс
printable = yes

# маска для создания файлов заданий на печать
create mode = 0700

# Команды, выполняемые Samba для того, что бы напечатать документ.
# использование драйвера клиента, применяется для не-UNIX
# клиентов.
print command = lpr-cups -P %p -o raw %s -r

# Использование драйвера CUPS на стороне сервера (на стороне
# клиентов используется generic PostScript драйвер).
; print command = lpr-cups -P %p %s

# Следующие команды являются стандартными при установке printing=cups,
# их можно изменить в случае необходимости.
lpq command = lpq -P %p
lprm command = cancel %p-%j
```

Сервер в составе существующего домена NT

Подключим вновь созданную машину Samba с именем `COMP` к существующему домену `DOM`, администратором которого является пользователь `Administrator` и PDC этого домена реализован на другом компьютере.

Первым делом необходимо убедиться, что машины с таким же именем, как и та, которую мы собираемся подключить, в домене ещё нет. В противном случае эту машину необходимо удалить из состава домена средствами самого PDC или выбрать другое имя.

На машине `COMP` в `/etc/samba/smb.conf` необходимо внести следующие изменения:

```
[global]

workgroup = DOM
netbios name = COMP
security = domain
password server = *
allow trusted domains = yes
nt acl support = yes
```

После чего необходимо остановить Samba-сервер, если он работает, командой **service smb stop**.

Теперь необходимо послать запрос на PDC с целью авторизации нового члена домена с помощью следующей команды:

```
$ smbpasswd -j DOM -r DOMPDC -U Administrator
```

и в ответ на запрос ввести пароль пользователя `Administrator` — тот самый, с которым этот пользователь зарегистрирован в домене.

Если получено сообщение:

```
Joined domain DOM.
```

все работает; иначе в `smb.conf` надо написать:

```
[global]

log level = 4
```

повторить последнюю команду и по подробным логам разбираться, что не так. При таком уровне `log level` в `log.smbd` содержится подробный отчёт об обмене с PDC. Вполне возможно, что были допущены ошибки в написании имён или ошибочно введён пароль; также возможны какие-либо неполадки на стороне PDC.

С этого момента, когда к Samba обратился пользователь “`user123`” с паролем “`passw`”, она:

- сначала ищет его в `/etc/samba/smbpasswd`, если пароль и имя совпадают — пускает, иначе отказывает в авторизации или считает гостем (в зависимости от настройки);
- если такого имени в упомянутом файле нет — смотрит в `/etc/passwd` (проверив соответствия через файл `/etc/samba/smbusers`) и
- если такой пользователь есть — спрашивает PDC, числится ли за пользователем “`user123`” полученный пароль “`passw`”;
- если это так — пускает, иначе отказывает в авторизации либо переключает на гостевой заход, в соответствии с настройкой.

Обычно при работе в домене на рядовых рабочих станциях `/etc/samba/smbpasswd` должен быть абсолютно пустым либо содержать только административные учётные записи, с доменом никак не связанные.

Данная логика работы применима только в том случае, если не используется winbind. Для того, чтобы доменные пользователи автоматически оказывались в `/etc/passwd` при первом же удачном обращении (правильность паролей была подтверждена PDC), в `/etc/samba/smb.conf` необходимо написать одну строку

```
[global]

add user script = /usr/sbin/useradd -d /home/domain/%u -g 600 -m\
-k /etc/skel_domain -s /bin/false %u
```

соответственно каталоги `/home/domain` и `/etc/skel_domain`, а также группа `600` должны уже существовать. Все конкретные имена и параметры **useradd** можно менять в зависимости от конкретных применений.

По директиве `add user script`, которая активизируется в тех случаях, когда пользователь ещё не зарегистрирован на данной машине, можно вызывать не только `/usr/sbin/useradd` с ключами, но и любые другие программы; если подойти с фантазией, то с помощью данной директивы можно делать очень интересные вещи.

Не стоит забывать и о безопасности — программы, запущенные при помощи `add user script`, будут выполняться от всемогущего в пределах системы пользователя `root`, а параметры их вызова частично определяются пользователем, что потенциально опасно!

Теперь можно включить сервер Samba командой: **service smb start** и работать в домене сети Windows на правах рядового члена домена.

Сервер как PDC домена

Для создания Primary Domain Controller (PDC) необходимо в `smb.conf` внести/изменить следующие записи

```
[global]

# Имя сервера; если данный параметр не определён,
# то он примет значение, соответствующее имени хоста.
netbios name = COOLSERVER

# Имя домена
workgroup = COOLDOMAIN

# Режим работы системы авторизации сервера.
security = user

# Разрешение на использование шифрованных паролей
encrypt passwords = yes

# Путь к локальному файлу паролей
smb passwd file = /etc/samba/smbpasswd

# Стать мастер-браузером для домена
local master = yes
```

```
# Быть PDC
domain master = yes

# Сразу при старте постараться стать мастер-браузером домена
preferred master = yes

# Быть сервером паролей домена
domain logons = yes

# Расположение профайла пользователей домена
logon path = \\%L\Profiles\%U

# Административная группа домена, присутствие в списке
# пользователя "administrator" весьма желательно, без
# этого данный пользователь не получит административных
# прав на клиентских машинах Windows.
domain admin group = root @wheel administrator

# Быть WINS-сервером. WINS-сервер имеет смысл когда в сети более 10
# машин, работающих по протоколу SMB. Наличие такого сервера в сложных
# сетях существенно снижает широковещательный трафик.
wins support = yes

# Порядок разрешения имён NetBIOS, по аналогии с записью в
# /etc/host.conf для разрешения имён DNS. Значение wins
# имеет смысл только при наличии в сети wins-сервера,
# в противном случае оно замедлит работу.
name resolve order = wins lmhosts bcast
```

Также необходимо создать ресурсы для работы домена.

Ресурс `netlogon` необходим для работы PDC и домена в целом. Он просто должен существовать.

```
[netlogon]

comment = Network Logon Service
path = /var/lib/samba/netlogon
guest ok = yes
writable = no
write list = admin, administrator
```

Данный ресурс необходим для создания и хранения профайлов пользователей домена:

```
[Profiles]

path = /var/lib/samba/profiles
browseable = no
read only = no
create mask = 0600
directory mask = 0700
```

При создании пользователя домена в `/var/lib/samba/profiles` автоматически создаётся каталог с именем, идентичным имени создаваемого пользователя и принадлежащий ему (с правами `0700`). В этом каталоге будут храниться личные настройки пользователя.

Для того, чтобы включить клиентскую машину в домен, необходимо произвести следующие действия.

Первый метод — вручную:

Прежде всего необходимо создать локального пользователя системы с именем, соответствующим `NetBIOS-name` подключаемой к домену машины. К имени на конце добавляется символ “$”. Для добавления машины с именем `machine_name` необходимо от имени пользователя `root` выполнить следующие команды:

```
# /usr/sbin/useradd -g machines -d /dev/null -c "machine nickname" -s
/bin/false machine_name$

# passwd -l machine_name$
```

Теперь, когда создан пользователь (символ “$” в конце имени означает что это NetBIOS-имя компьютера, а не имя пользователя), можно добавить его в домен, выполнив от имени `root` команду: **smbpasswd -a -m machine_name**.

Теперь компьютер подключён к домену.

Второй метод — автоматический:

Работу со созданию машинного акаунта можно переложить на Samba, включив в `smb.conf` следующую запись:

```
[global]

add user script = /usr/sbin/useradd -d /dev/null -g machines -s
/bin/false -M %u
```

Теперь Samba будет принимать от клиентских машин запросы на включение в домен и автоматически регистрировать их аналогично NT Server.

С этого момента начинает существовать домен и PDC на базе Samba-сервера. Пользователи могут входить под своими именами и паролями с любой машины домена с сохранением настроек, а также самостоятельно менять свои пользовательские пароли без помощи администратора сети.

Учётные записи пользователей

Все учётные записи хранятся в файле `/etc/samba/smbpasswd`.

Учётные записи пользователей, используемые Samba делятся на две категории:

- записи о компьютерах, входящих в домен;
- записи о пользователях, зарегистрированных на данном сервере.

Следует учитывать, что для того, что бы создать и использовать любую учётную запись в `/etc/samba/smbpasswd`, предварительно необходимо создать соответствующую запись в `/etc/passwd`. Общее правило — для каждого пользователя в `/etc/samba/smbpasswd` обязательно должен существовать пользователь в `/etc/passwd`. Обратное утверждение неверно.

Для управления учётными записями предназначена утилита **smbpasswd**; полный список её возможностей можно узнать из соответствующей man-страницы, здесь же рассмотрим наиболее частые методы использования.

Создание нового пользователя:

```
# smbpasswd -a <User_name>
```

Смена пароля у существующего пользователя:

```
# smbpasswd <User_name>
```

Удаление существующего пользователя:

```
# smbpasswd -x <User_name>
```

Приостановление учётной записи без удаления:

```
# smbpasswd -d <User_name>
```

Подключение данного компьютера к существующему домену:

```
# smbpasswd -j <Domain_name> -U <Administrator_name>
```

Использование winbind

Сервис winbind является новым средством, предназначенным для более полной интеграции Samba в домены Windows; он появился, начиная с Samba 2.2.0. Данный сервис считывает свою конфигурацию из `/etc/samba/smb.conf` и динамически взаимодействует с PDC домена, автоматически синхронизируя списки пользователей и групп домена и машины Samba. Таким образом, winbind является весьма удобным средством для автоматического поддержания актуальности базы пользователей домена на рабочих станциях Samba.

Работа данного сервиса происходит без изменения содержимого каких либо авторизационных файлов в `/etc` и при перезагрузке машины доменные пользователи появляются в системе только после запуска **winbindd**. Если во время работы остановить **winbindd**, то доменные пользователи и группы не исчезнут из системы до перезагрузки, однако динамического обновления списков имён и паролей происходить не будет.

Для того, что бы при рестарте компьютера (или только сервиса **winbindd**) не нарушались соответствия внутренних UID и доменных SID, он сохраняет текущее состояние списков в файлах `/var/cache/samba/winbindd*.tdb`.

Для нормального функционирования **winbindd** в файле `/etc/samba/smb.conf` обязательно должны быть объявлены следующие директивы:

```
[global]

# Диапазон номеров локальных пользователей, который будет
# использован для динамического создания пользователей домена.
winbind uid = 10000-20000

# Диапазон номеров локальных групп пользователей, который будет
# использован для динамического создания групп пользователей
# домена.
winbind gid = 10000-20000

# Символ-разделитель, используемый для составления доменных имён
# пользователей и располагающийся между именем домена и именем
# пользователя.
winbind separator = +

# Интервал времени (в секундах) между запросами winbind к PDC
# в целях синхронизации списков пользователей и групп.
winbind cache time = 10

# Шаблон имени домашних каталогов доменных пользователей,
# автоматически присваиваемых каждому пользователю. Сами каталоги,
# однако, динамически не создаются. Вместо переменой %D подставляется
# имя домена, а вместо %U подставляется имя пользователя.
template homedir = /home/%D/%U

# Командный интерпретатор, назначаемый по умолчанию для
# пользователей, авторизованных через winbindd.
template shell = /bin/bash
```

Также необходимо внести изменения в файле `/etc/nsswich.conf` в разделы `passwd` и `group`, вписав директиву `winbind` — например, таким образом:

```
passwd: files winbind
group: files winbind
```

С этого момента можно использовать имена доменных пользователей в `/etc/samba/smb.conf` с целью разграничения доступа, в правах на файлы и каталоги, для подключения к сетевым ресурсам данного хоста со стороны других хостов.

Принт-сервер на CUPS

По умолчанию Samba сконфигурирована на использование CUPS в качестве спулера печати. Подразумевается, что CUPS уже настроен и запущен. В `/etc/samba/smb.conf` присутствуют следующие директивы:

```
[global]

printcap name = lpstat
load printers = yes
printing = cups
```

Также необходимо создать ресурс `[printers]`; его создание и назначение директив подробно описано в разделе “Обычный сервер” в части Особые ресурсы.

Настройка клиента

Для подключения компьютера Linux к сетям SMB существуют клиентские функции Samba.

Обычный клиент

Клиентские функции Samba представлены средствами просмотра сетевого окружения и монтирования файловых систем **/usr/bin/smbclient** и **/usr/bin/smbmount** соответственно. Также доступны **mount.smb** и **mount.smbfs**, являющиеся символическими ссылками на `/usr/bin/smbount`.

При запуске эти программы считывают текущую конфигурацию из файла `/etc/samba/smb.conf` и используют доменные функции в случае, если машина подключена к домену Windows.

Также файловые системы возможно монтировать системной командой mount, указав в качестве типа файловой системы smbfs, и использовать эти записи в `/etc/fstab` для автоматического монтирования при загрузке системы.

Например, для того что бы смонтировать в каталог `/mnt/disk` ресурс `public` с машины `SMALLSERVER` под именем `cooluser`, нужно выполнить команду: **smbmount //smallserver/public /mnt/disk -o username=cooluser**

Регистр написания имён компьютеров, ресурсов и пользователей роли не играет. Для того, что бы получить список Samba-ресурсов данной машины и список машин рабочей группы или домена достаточно выполнить команду: **smbclient -L localhost -N**

Более подробные сведения можно прочесть в man-страницах по **smbclient** и **smbmount**.

В составе дистрибутива поставляются два графических клиентских приложения — LinNeighborhood и gnomba [11], которые работают поверх утилит **smbclient** и **smbmount**.

По адресу http://www.public.iastate.edu/~chadspen/homepage.html можно получить весьма качественное графическое клиентское приложение xSMBrowser.

Клиент в составе существующего домена NT

Подключение происходит аналогично рассмотренному в п. “ Сервер в составе существующего домена NT ”. Далее вся работа происходит точно так же, как описано в предыдущем пункте.

Особенности локализации клиента и сервера

Для того, чтобы все компоненты Samba правильно работали с русскими именами файловых объектов и ресурсов, в /etc/samba/smb.conf необходимо добавить следующие директивы:

```
[global]

client code page =
character set =
```

Далее приводятся наборы значений этих директив и системных кодировок, наиболее часто используемых в России, Белоруссии и на Украине:

```
$LANG = ru_RU.KOI8-R
client code page = 866
```

[11] Предпочтительнее использовать первый из них.

```
characte set = koi8-r

$LANG = ru_RU.CP1251
client code page = 866
characte set = 1251

$LANG = be_BY.CP1251
client code page = 866
character set = 1251

$LANG = uk_UA.KOI8-U
client code page = 1125
character set = koi8-u

$LANG = uk_UA.CP1251
client code page = 1125
character set = 1251U

$LANG = ru_UA.CP1251
client code page = 1125
character set = 1251U
```

В двух последних случаях `1251U` — специальное обозначение внутри Samba для комбинации локально "`1251` — удалённо `1125`". В Samba определение удалённой кодировки делается по имени локальной [12].

Также необходимо проследить, чтобы на тех компьютерах Windows, с которыми предполагается взаимодействие через Samba, были установлены соответствующие системные настройки локализации. В противном случае велика вероятность, что вместо кириллических символов будут отображены знаки "?" либо другие непрошенные символы.

Указанные директивы `/etc/samba/smb.conf` воздействуют на работу всех компонентов Samba — и серверных, и клиентских. На данный момент поддерживаются кириллические написания имён — файлов, каталогов и ресурсов.

Некоторые вопросы безопасности

Данный раздел относится в основном к серверной части Samba.

Прежде всего необходимо определить, какие интерфейсы должны прослушиваться Samba в ожидании запроса на соединение (по умолчанию прослушиваются все имеющиеся в системе).

Например, для того, чтобы ограничить прослушивание локальным хостом и первой сетевой картой, необходимо написать в `/etc/samba/smb.conf`:

```
[global]

interfaces = 127.0.0.1 eth0
bind interfaces only = Yes
```

Далее можно ограничить диапазоны адресов, с которых позволительно обращаться к данному серверу. Действие данных директив аналогично воздействию `/etc/hosts.allow` и `/etc/hosts.deny` на `xinetd` и `ssh`: если IP-адрес хоста не подпадает под разрешающее правило, то соединение не будет

[12] Например, кодовая страница `1251` в качестве локальной однозначно задаёт удалённую `866`.

установлено вовсе. Для того, что бы ограничить доступ двумя подсетями и локальной системой, дополнительно исключив при этом один хост, можно написать:

```
[global]

hosts allow = 192.168.1. 192.168.2. 127.
hosts deny = 192.168.1.12
```

Все вышеперечисленные директивы ограничивают соединения на уровне интерфейсов и IP-адресов до какой либо авторизации. Следующие директивы управляют режимом авторизации пользователей.

Во избежание перехвата чувствительных данных при передаче их по сети открытым текстом принято шифровать пароли. Samba и все версии Windows, начиная с версии Win98, по умолчанию используют шифрование паролей. Данная директива включает его в Samba:

```
[global]

encrypt passwords = yes
```

Файл переопределений имён пользователей является весьма мощным средством управления пользовательскими акаунтами, однако при неразумном использовании это средство опасно и поэтому по умолчанию отключено. Внимательно ознакомьтесь с содержимым файла `/etc/samba/smbusers`, прежде чем использовать его.

```
[global]

; username map = /etc/samba/smbusers
```

Особенности использования Samba 3.0

Samba 3.0 имеет заметные отличия от более ранних версий; наиболее выдающимися из них являются улучшенная по сравнению с версией 2.2 поддержка Unicode, поддержка гораздо большего количества кодовых страниц, новая утилита администрирования **net**, призванная заменить **smbpasswd**.

В поставку входят пакеты `samba3-client`, `samba3-client-cups`, `samba3-common`, `samba3`, `samba3-swat`.

Задание кодовых страниц

Для задания кодировок используются следующие новые параметры `smb.conf`:

```
unix charset = <charset>
dos charset = <charset>
display charset = <charset>
```

где `<charset>` — любая кодировка, поддерживаемая `iconv`. Список возможных кодировок можно узнать, выполнив команду **iconv -list**.

Параметры `client code page` и `character set` больше не поддерживаются. Параметр `unix charset` указывает кодировку, в которой будут храниться файлы на диске, в которой заданы параметры в `smb.conf`. Наконец-то появилась возможность хранить имена файлов в UTF-8!

Параметр `dos charset` указывает кодировку, в которой Samba будет общаться с клиентами, не поддерживающими Unicode. Все версии Windows, начиная с 95, понимают Unicode — но все же стоит установить `dos charset = cp866`, что соответствует `client code page = 866` в более старых версиях.

Параметр `display charset` указывает в какой кодировке должны выводить информацию программы, непосредственно обменивающиеся информацией с пользователем, например **smbclient**, **net**, **wbinfo** и другие.

Утилита net

Утилита **net** призвана заменить **smbpasswd** и обеспечивает гораздо большие возможности по получению информации о сети и управлению сетью. Формат команд утилиты очень похож на формат одноимённой команды Windows NT/2000.

Основные применения команды **net**:

- создание и удаление пользователей: **net user**
- включение машины в домен: **net ads join** — Active Directory; **net rpc join** — NT Domain;
- получение информации о домене, машине, открытых файлах, сессиях: **net info**, **net ads status**, **net rpc status**;
- создание и удаление разделяемых ресурсов на удалённых машинах: **net share**;
- синхронизация времени с windows-сервером: **net time**

Управление машиной с Samba из Microsoft Management Console

Начиная с версии 2.2, Samba имеет возможность удалённого администрирования из MMC (Microsoft Management Console). Эта возможность полезна, когда Samba является членом NT-домена или AD. Администратор домена может создавать, удалять и изменять сетевые ресурсы на UNIX-машине с запущенной Samba.

Как сконфигурировать Samba для удалённого администрирования? Для управления ресурсами служат параметры `/etc/samba/smb.conf`:

```
[global]

add share command = <add script>
```

Параметр указывает скрипт, который будет вызван при попытке создания нового ресурса в MMC. Скрипту передаётся четыре параметра:

- имя конфигурационного файла (например, `/etc/samba/smb.conf`);
- имя создаваемого ресурса;
- путь к существующей директории на диске;
- комментарий.

Скрипт должен завершаться с кодом `0` в случае успешного создания и ненулевым в случае ошибки.

```
change share command = <change script>
```

Параметр указывает скрипт, который будет вызван при попытке изменения существующего ресурса в MMC. Скрипту передаётся четыре параметра:

- имя конфигурационного файла (например, `/etc/samba/smb.conf`);
- имя создаваемого ресурса;
- путь к существующей директории на диске;
- комментарий.

Скрипт должен завершаться с кодом `0` в случае успешного создания и ненулевым в случае ошибки.

```
delete share command = <delete script>
```

Параметр указывает скрипт, который будет вызван при попытке удаления существующего ресурса в MMC (Stop sharing). Скрипту передаётся два параметра:

- имя конфигурационного файла (например, `/etc/samba/smb.conf`);
- имя создаваемого ресурса;

Скрипт должен завершаться с кодом `0` в случае успешного создания и ненулевым в случае ошибки.

Чтобы скрипты могли изменять конфигурационные файлы Samba, они должны выполняться с правами root. Для этого нужно установить отображение пользователей домена, имеющих право изменять ресурсы, в root. Это можно сделать либо с помощью файла /etc/samba/smbusers, прописав там строку вида

```
root = administrator <user 1> ... <user n>
```

либо с помощью параметра admin users в `/etc/samba/smb.conf`:

```
admin users = administrator
```

При создании нового ресурса Windows позволяет просматривать дерево директорий. Для этого в `/etc/samba/smb.conf` нужно задать служебные ресурсы, заканчивающиеся символом “$”, например:

```
[C$]

path = /drives/c
```

После этого при создании нового ресурса можно будет просматривать и выбирать все директории ниже `/drives/c`.

Работа в среде Active Directory

Для объединения компьютеров в домены Widows 2000 Server использует схему, отличную от NT-доменов, которая называется Active Directory; эта схема обладает гораздо большей масштабируемостью и позволяет централизованно администрировать машины, входящие в домен. Active Directory базируется на протоколе авторизации Kerberos, при котором имя пользователя и пароль не передаются по сети, а используется механизм так называемых билетов, выдаваемых сервером на определённое время. Получив билет, машина, входящая в домен, может авторизоваться на других машинах домена без участия сервера.

Установка Samba

Samba 3.0, в отличие от более ранних версий Samba, имеет возможность работать в сетях Wndows, работающих в режиме Active Directory (или Windows 2000 native mode). Если требуется эта функциональность, следует установить пакет `samba3-3.0` вместо `samba-2.2`.

Active Directory имеет другую схему именования доменов, компьютеров и пользователей, основанную на DNS. Допустим, существует сеть с именем `my.firm.com` и компьютерами `host1.my.firm.com`, `host2.my.firm.com`, `host3.my.firm.com`; тогда домен Active Directory будет называться `my.firm.com`, а пользователи Active Directory будут иметь имена вида `user@my.firm.com`.

Настройка

`/etc/krb5.conf` должен содержать по крайней мере следующие строки:

```
[realms]

MY.FIRM.COM = {
kdc = your.kerberos.server
}
```

где `MY.FIRM.COM` - имя домена (или “царства”, в терминологии Kerberos; задаётся обязательно в верхнем регистре), а `your.kerberos.server` — имя или IP-адрес KDC (Kerberos Domain Controller), аналог PDC (Primary Domain Controller) в доменах Windows NT — например, `server.my.firm.com` или `192.168.117.11`.

Правильность указания параметров можно проверить, выполнив команду (замените имя пользователя на актуальное — например, `administrator@MY.FIRM.COM`):

```
# kinit username@REALM
```

и убедившись, что пароль был принят сервером. `REALM` всегда задаётся в верхнем регистре.

Вы также должны убедиться, что возможно получить имя KDC по его IP адресу (так называемый Reverse DNS lookup). Имя KDC должно либо совпадать с NetBIOS-именем компьютера (имя машины в сети Windows без указания домена) либо состоять из NetBIOS-имени и имени домена. Если получить имя KDC по адресу невозможно, вы получите ошибку “local error” при попытке войти в домен.

Если ваш DNS не поддерживает Reverse lookup либо KDC не зарегистрирован в DNS, вы можете указать соответствие IP-адреса и имени в `/etc/hosts`.

Редактирование `/etc/samba/smb.conf`

Для работы в Active Directory `smb.conf` должен содержать следующие параметры:

```
[global]
```

```
# Задаёт Kerberos realm, обычно совпадает с именем домена в
# верхнем регистре, например realm = MY.FIRM.COM
realm = <REALM>

# Это обычно часть реалма до первой точки, например
# workgroup = MY
workgroup = <WORKGROUP>

# Тип домена - Active Directory.
security = ADS

# В случае Active Directory пароли всегда шифруются.
encrypt passwords = true

# Обычно этот параметр указывать не обязательно, т.к. Samba сама
# определяет адрес KDC, если в сети есть WINS-сервер и он указан
# в smb.conf
ads server = <your.kerberos.server>
```

Регистрация компьютера в Active Directory домене

Убедитесь что Samba не запущена. Если запущена, её нужно остановить:

```
# service smb stop; service winbind stop
```

Чтобы включить компьютер в домен, выполните команду:

```
# net ads join -U administrator
```

где `administrator` — имя пользователя домена, имеющего право создавать новые учётные записи.

Если не было выдано сообщение об ошибке, то машина успешно зарегистрирована в домене — иначе проверьте правильность задания параметров в `/etc/samba/smb.conf` и `/etc/krb5.conf`. Убедитесь, что пользователь, указанный после `-U` в **net ads join**, имеет необходимые права на создание новых учётных записей.

Теперь можно запустить необходимые службы:

```
# service smb start; service winbind start
```

Проверка правильной работы в Active Directory

Для работы с компьютерами, зарегистрированными в Active Directory, не требуется указывать имя пользователя и пароль. Попробуйте выполнить команду:

```
$ smbclient -k -L <имя компьютера в домене>
```

Вы должны получить список доступных ресурсов, при этом **smbclient** не должен запрашивать имя пользователя и пароль.

Чтобы проверить, что доступ к ресурсам вашей машины возможен с других машин домена, вы можете попробовать выполнить все ту же команду:

```
$ smbclient -k -L <имя вашего компьютера в домене>
```

и получить список доступных ресурсов. Можно также попробовать открыть какой-нибудь ресурс на вышей машине с Windows-машины, входящей в домен. В любом случае имя пользователя и пароль запрашиваться не должны.

Некоторые особенности работы в Active Directory

В отличие от доменов Windows NT, авторизация в Active Directory производится не по имени и паролю, а с помощью билетов протокола Kerberos. Из-за этого работа с **smbclient** может поначалу показаться необычной.

Во-первых, при вызове **smbclient** нужно указывать параметр `-k`. Во-вторых, билеты Kerberos даются на определённое время (обычно на сутки, но это зависит от настроек сервера). Поэтому их нужно периодически обновлять с помощью команды **kinit**:

```
# kinit username@REALM
```

где `username` — ваше имя в Actiev Directory домене `REALM`.

Так что если **smbclient** вдруг перестаёт подключаться к доменным ресурсам, попробуйте обновить билет — скорее всего, дело именно в этом.

Литература

[inet] Официальный сайт проекта `http://www.samba.org` [http://www.samba.org] .

[local] `/usr/share/doc/samba-*/docs/*` — комплект документации .

[man] man-страницы `samba(7)`, `smb.conf(5)`, `smbmount(8)`, `smbclient(1)`, `smbpasswd(8)`, `winbindd(8)` .

Глава 11. WWWOFFLE

Алексей Турбин

<at@turbinal.org>

История переиздания
Издание 0.2 03 ноября 2002
Создана разметка в DocBook/XML v.4.2. Внесена литправка.

Вступление

WWWOFFLE (World Wide Web Offline Explorer) — это прокси-сервер, предназначенный для использования на компьютерах с ограниченным доступом к Интернету (dial-up).

Сервер работает в двух режимах: `online` (при установленном соединении с Интернетом) и `offline` (при отсутствии соединения). В режиме `offline` доступны для просмотра страницы, закэшированные ранее в режиме `online`. Кроме того, в режиме `offline` можно "заказать" для просмотра страницы, которые будут загружены (**fetch**) при следующем подключении к сети. Поддерживается регулярный мониторинг страниц, а также рекурсивная загрузка. Сервер имеет много возможностей и настроек.

Установка и настройка

Установку проще всего произвести стандартными средствами (**apt-get install wwwoffle**). Сразу после установки нужно проверить, включена ли загрузка сервера по умолчанию (с помощью утилиты chkconfig), и запустить сервер (**service wwwoffle start**).

В настройках браузера необходимо указать адрес сервера:

```
localhost:8080 (HTTP, FTP)
```

Сервер также поддерживает протокол *finger* и *HTTPS*-туннелирование).

Для того, чтобы при подключении к Интернету (а также при установленном соединении) сервер автоматически менял режим работы, небходимо отредактировать конфигурационные скрипты `/etc/ppp/ip-up.local` и `/etc/ppp/ip-down.local` (если они не существуют, их нужно создать с правами доступа `0755 root:root`). В первый из них нужно поместить команды:

```
#!/bin/sh
wwwoffle -online
wwwoffle -fetch
```

Во второй файл следует поместить команды:

```
#!/bin/sh
wwwoffle -offline
```

Еженедельно на компьютере будет запускается процесс удаления (**purge**) устаревших документов в кэше.

Настройку сервера можно осуществлять двумя способам: с помощью редактирования конфигурационного файла (`/etc/wwwoffle.conf`) и с помощью web-интерфейса (http://localhost:8080). После редактирования конфигурационного файла нужно перезагрузить конфигурацию сервера (**service wwwoffle reload**).

Далее перечислены некоторые конфигурационные директивы сервера:

- `request-changed` — время, по истечении которого сервер будет считать документы считать устаревшими, то есть не будет загружать одну и ту же страницу слишком часто; для этой и других директив можно задавать разные значения по шаблону адреса документа: например, рисунки можно кэшировать “сильнее”, чем html-страницы;
- `request-expired`, `request-no-cache` — должен ли сервер строго придерживаться правил обновления страниц при повторной загрузке;
- `FetchOptions`— позволяет настроить автоматическую загрузку рисунков, стилей и скриптов вместе со страницами;
- `add-cache-info` — добавляет к страницам информацию о кэшировании;
- `DontGet` — список шаблонов для игнорирования файлов (позволяет отключать баннерную рекламу);
- `disable-webbug-images`, `disable-dontget-iframes` — позволяет отключить загрузку рисунков и фрэймов размером 1x1;
- `LocalNet`, `AllowedConnectHosts`, `AllowedConnectUsers` — опции управления доступом к серверу.
- `Purge` — управление очисткой кэша; есть возможность хранить на диске закэшировнные страницы в сжатом виде.

Предостережение

В настоящее время WWWOFFLE не рекомендуется использовать на серверах с повышенными требованиями к безопасности.