



Министерство науки и высшего образования
Российской Федерации
Братский педагогический колледж
федерального государственного бюджетного
образовательного учреждения высшего
образования
«Братский государственный университет»

ОПЕРАЦИОННЫЕ СИСТЕМЫ И СРЕДЫ

основы администрирования операционной системы Ubuntu Linux

**методические рекомендации
по выполнению лабораторных работ**

для студентов II курса
очной формы обучения
специальности

09.02.07 Информационные системы и программирование

Автор: Ю.Н. Войтухов

Братск, 2021

Операционные системы и среды. Основы администрирования операционной системы Ubuntu Linux. Методические рекомендации по лабораторным работ /Сост. Ю.Н. Войтухов.- Братск, 2021.- 78 с.

Содержат указания к выполнению лабораторных работ по дисциплине «Операционные системы и среды». В лабораторных работах содержатся основные теоретические сведения, касающиеся администрирования операционной системы Ubuntu Linux, а так же настроек данной операционной системы для обеспечения информационной безопасности.

Предназначены для студентов специальности 09.02.07 Информационные системы и программирование.

Печатается по решению научно-методического совета
Братского педагогического колледжа ФГБОУ ВО «БрГУ»
665709, г. Братск, ул. Макаренко 40

ЛАБОРАТОРНАЯ РАБОТА №1

Установка и настройка виртуальной машины. Установка Linux

Цель работы: Установить Linux Ubuntu на виртуальную машину VM VirtualBox.

Задачи работы:

- Изучить интерфейс VM Virtual Box;
- Создать виртуальную машину;
- Установить Linux Ubuntu на виртуальную машину;
- Изучить способы разметки жесткого диска в Linux.

Теоретическая часть

Virtual Box 4.1.4

Oracle VM VirtualBox - программный продукт виртуализации для операционных систем Microsoft Windows, Linux, FreeBSD, Mac OS X, Solaris, ReactOS, DOS и других.

Основные характеристики:

- Кроссплатформенность;
- Модульность;
- Поддержка USB 2.0, когда устройства хост-машины становятся доступными для гостевых ОС (только в проприетарной версии);
- Поддержка 64-битных гостевых систем , даже на 32-битных хост-системах;
- Поддержка SMP на стороне гостевой системы;
- Встроенный RDP-сервер, а также поддержка клиентских USB-устройств поверх протокола RDP;
- Поддержка образов жёстких дисков VMDK (VMware) и VHD (Microsoft Virtual PC), включая snapshots;
- Поддержка iSCSI;
- Поддержка виртуализации аудиоустройств (эмуляция AC97 или SoundBlaster 16 или Intel HD Audio на выбор);
- Поддержка различных видов сетевого взаимодействия (NAT, Host Networking via Bridged, Internal);

- Поддержка цепочки сохраненных состояний виртуальной машины (snapshots), к которым может быть произведён откат из любого состояния гостевой системы;
- Поддержка Shared Folders для простого обмена файлами между хостовой и гостевой системами (для гостевых систем Windows 2000 и новее, Linux и Solaris);
- Поддержка интеграции рабочих столов (seamless mode) хостовой и гостевой ОС;
- Есть возможность выбора языка интерфейса (поддерживается и русскоязычный интерфейс).
- Описание интерфейса

Интерфейс Oracle VM VirtualBox Manager представлен на рис. 1.1. В левой панели располагается список всех виртуальных операционных систем. Для каждой из операционных систем создается виртуальный диск. При первом запуске программы список виртуальных операционных систем пуст.

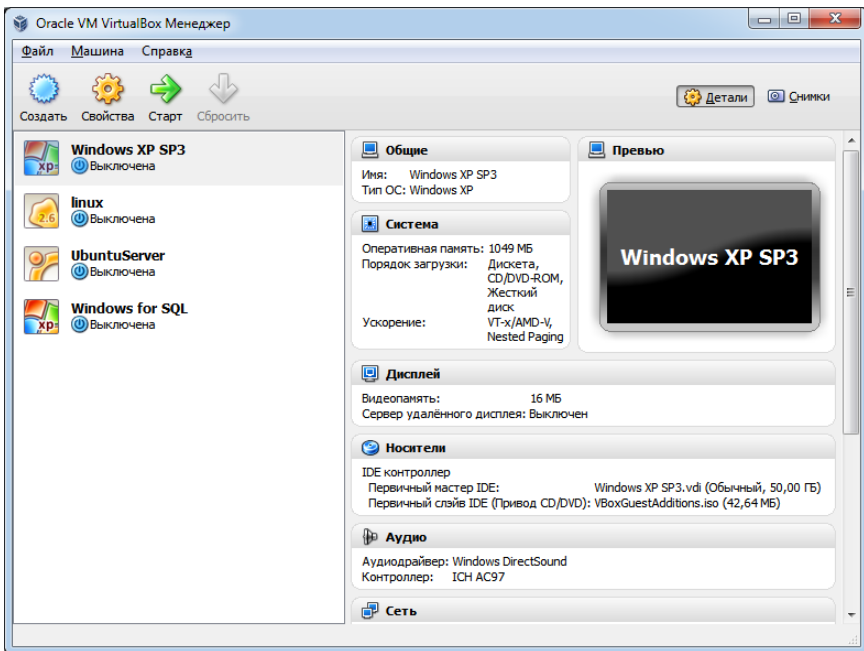


Рис. 1.1. Интерфейс Oracle VM VirtualBox

Справа от списка операционных систем отображается информация: наименование, объем оперативной памяти, порядок загрузки носителей и т.д. Свойства каждой операционной системы можно изменить, нажав на кнопку «Свойства».

Кнопка «Создать» вызовет мастер создания новой виртуальной операционной системы, а так же жесткого диска, если это потребуется.

Кнопка «Старт» позволяет загрузить выбранную операционную систему, а кнопка «Сбросить» - выключить.

На панели вверху находятся следующие пункты меню: Файл, Машина и Старт.

Пункт меню «Файл» позволяет работать с конфигурацией виртуальной машины (экспорт и импорт), изменять текущую конфигурацию виртуальной машины, а так же работать с виртуальными носителями (вызывает менеджер виртуальных носителей). Так же он содержит такой подпункт меню, как «Выход». При выборе данного пункта меню закроется окно Oracle VM VirtualBox Manager, однако запущенные виртуальные операционных системы продолжат функционировать.

Пункт меню «Машина» позволяет работать с виртуальными машинами (создавать, добавлять, изменять, копировать и удалять; запускать, сбрасывать и приостанавливать работу виртуальной машины).

Пункт меню «Справка» позволяет узнать информацию о программе, получить информацию о работе программы, обновить Oracle VM VirtualBox Manager и многое другое.

Создание виртуальной машины

Для вызова мастера создания новой виртуальной машины нажмите на кнопку «Создать».

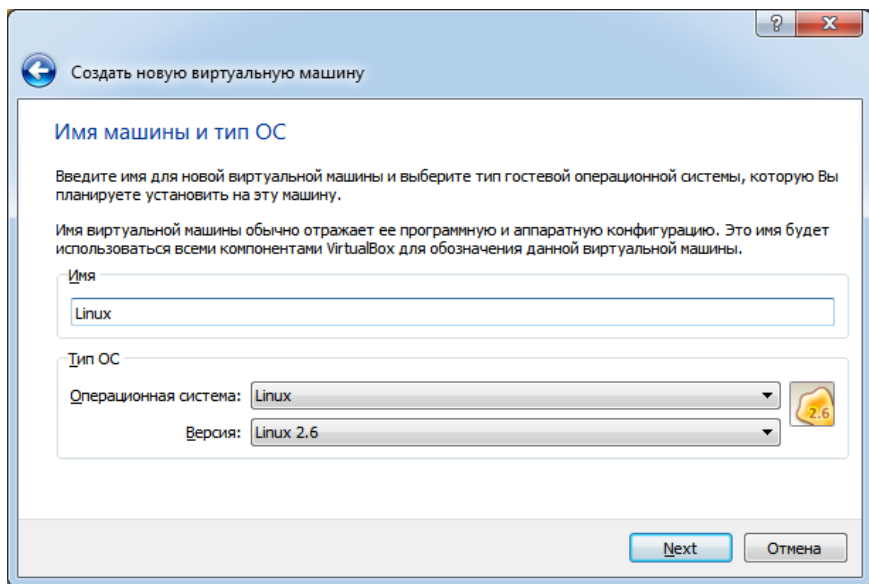


Рис. 1.2. Мастер создания виртуальной машины

Первая страница мастера – введение, которое содержит информацию о том, как пользоваться мастером. Нажмите «Далее» для перехода на следующую страницу. На второй странице необходимо ввести имя виртуальной операционной системы и выбрать её тип. Заполните данные поля так, как показано на рис. 1.2.

Третья страница позволяет выбрать количество основной памяти оперативного запоминающего устройства (далее – ОЗУ). Перетаскивая ползунок выберите количество необходимой памяти. Количество памяти не должно быть меньше 256Мб. Достаточное количество для Unix-систем – 512Мб, для Windows-систем – 1024Мб.

Следующий шаг позволяет выбрать виртуальный жесткий диск. Если такого диска нет, необходимо создать его, вызвав мастер создания нового виртуального носителя (рис. 1.3).

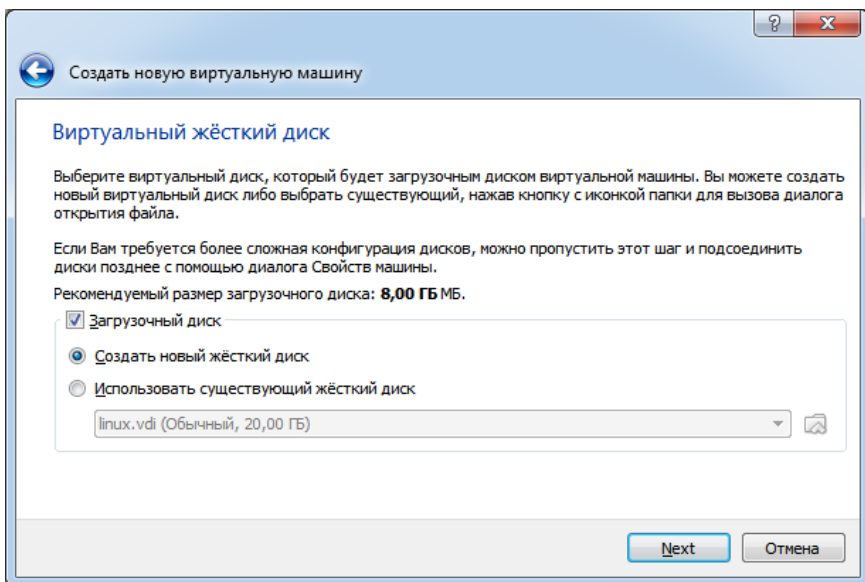


Рис. 1.3. Создание виртуального жесткого диска

Создание нового виртуального носителя начинается с выбора типа файла виртуального носителя. Начиная с версии VirtualBox 4.1.4 на выбор предлагаются четыре вида файла: VDI, VMDK, VHD, HDD. Нам нет необходимости использовать другие программы виртуализации, поэтому оставляем VirtualBox Disk Image (VDI). Следующая страница мастера предлагает нам выбор одного из двух способов создания файла виртуального диска: динамический виртуальный диск или фиксированный. Динамический виртуальный диск будет экономить место на вашем жестком диске, так как его размер будет динамически расширяться в зависимости от того, что мы будем делать с виртуальной операционной системой. Поэтому мы выберем динамический виртуальный диск.

Жмем «Далее» и попадаем на страницу выбора директории хранения виртуального жесткого диска и выбора его емкости. Емкость жесткого диска не должна быть меньше 8Гб. В зависимости от ресурсов вашего компьютера укажите необходимую емкость от 10 до 20Гб. Большая емкость будет излишней. Здесь же укажите директорию, в которой будет храниться ваш виртуальный жесткий диск. С учетом того, что файл жесткого диска в процессе работы с

ним может весить до 20Гб, укажите тот раздел вашего жесткого диска, где данный файл сможет храниться (рис. 1.4). Так как раздел D:/ жесткого диска на рис. 1.4 имеет емкость 90Гб, мы указали именно его, в качестве директории для хранения файла виртуального жесткого диска.

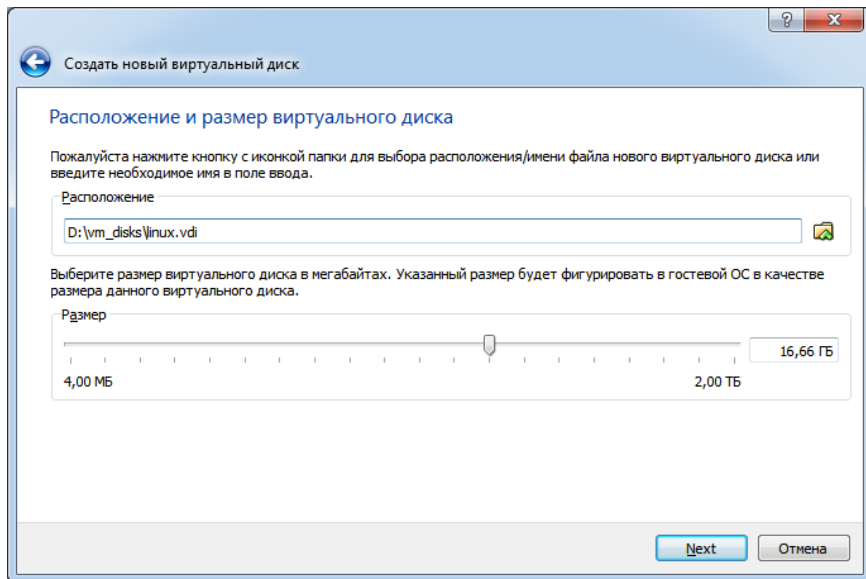


Рис. 1.4. Выбор директории хранения виртуального жесткого диска

На последней странице мастера создания виртуального жесткого диска приводится вся информация о том, что мы проделали за предыдущие шаги. Если все сведения верны, нажмите на кнопку «Создать», после чего по указанной выше директории будет создан файл linux.vdi.

Если не возникло никаких проблем, ваша виртуальная машина отобразится в списке слева.

Установка Ubuntu на виртуальную машину

В данной лабораторной работе используется дистрибутив Ubuntu 10.04. Скачать данный дистрибутив можно с официального сайта этой операционной системы: www.ubuntu.ru.

О разметке жесткого диска

Разделы диска бывают трёх типов: основные, расширенные и логические. Связаны они так: непосредственно диск делится на основные разделы, один из основных разделов может быть назначен расширенным и разделён на логические. При этом основных разделов может быть максимум четыре (с учётом расширенного), расширенный, если есть, то всегда один, а логических может быть сколько угодно. Другими словами: вы можете разделить диск максимум на 4 части, но одну из них вы можете спокойно поделить на сколько угодно частей.

Учитывайте вышесказанное при разметке. Некоторые программы, например, спокойно позволят вам создать не один расширенный раздел, а несколько. Однако ни Ubuntu, ни уж тем более Windows не увидят логические диски на таких разделах.

Об отношении Linux к разделам жесткого диска

Linux при работе с различными устройствами и источниками данных использует иную идеологию, в отличие от Windows. Для каждого объекта (источника данных или устройства) создаётся специальный файл, через который происходит «общение» этого объекта с системой. В частности, подобные файлы есть и для жестких дисков и разделов на них. И обычно при описании работы с дисками и разделами в качестве названий используются как раз имена этих файлов.

Жесткие диски называются sda, sdb, sdc и т.д. (sda - первый жесткий диск, sdb - второй и далее по аналогии). Кстати, подключаемые флеш-диски и другие USB устройства так же идентифицируются как жесткие диски и тоже получают имена вида sd*.

Разделы на дисках называются так: sda1, sda2, sda3 и т.д. Т.е. название раздела состоит из названия жесткого диска и цифры-номера раздела после него. Но тут есть некая особенность. Первые четыре цифры зарезервированы для основных разделов, а нумерация логических начинается всегда с пяти. Например, рассмотрим такое разбиение жесткого диска:

- sda1 – основной;
- sda2 – расширенный:
 - sda5 – логический;
 - sda6 – логический;

- sda7 – логический;
- sda3 – основной.

Как видно, у нас имеется 2 основных и 3 логических раздела, то есть в операционной системе у нас будет доступно 5 дисков на данном жестком диске. При этом четвертого основного раздела нет, соответственно, нет и специального файла sda4 в системе.

Обратите внимание, расширенный раздел - это всего лишь контейнер для логических, поэтому из операционной системы он недоступен и никакие данные на него записать нельзя.

Установка Ubuntu

После загрузки с установочного диска появится следующее диалоговое окно (рис. 1.5).

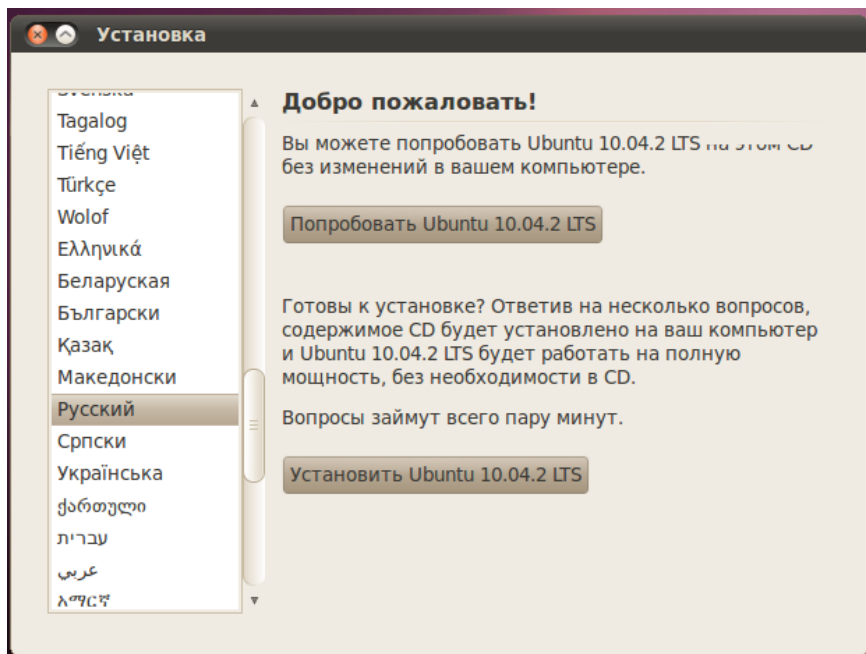


Рис. 1.5. Установка Linux, выбор языка

В данном окне необходимо выбрать русский язык и нажать на кнопку «Установить Ubuntu 10.04.2 LTS». Далее необходимо указать часовой пояс и нажать «Вперёд» (рис. 1.6):

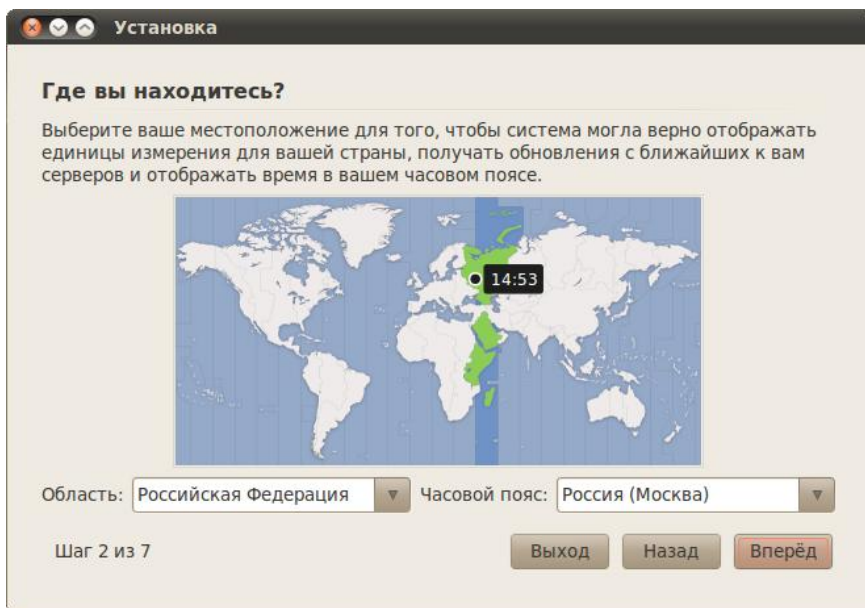


Рис. 1.6. Установка Linux, выбор часового пояса

Далее выберите свою раскладку клавиатуры. Если вы не знаете, что это такое, то, скорей всего, вам ничего не надо тут менять. В любом случае, вы можете проверить выбранную раскладку в специальном поле внизу окна (рис. 1.7).

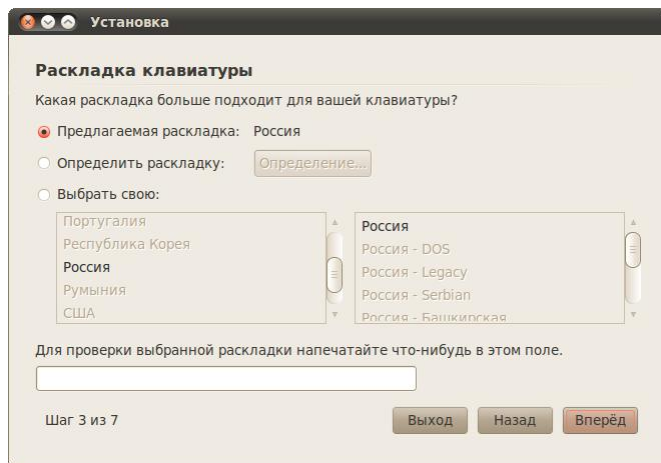


Рис. 1.7. Установка Linux, раскладка клавиатуры

Далее необходимо указать место для установки. Мастер установки покажет вам графическое представление вашего диска и предложит вам два варианта дальнейших действий (или три, если у вас уже установлена операционная система) так, как это показано на рис. 1.8.

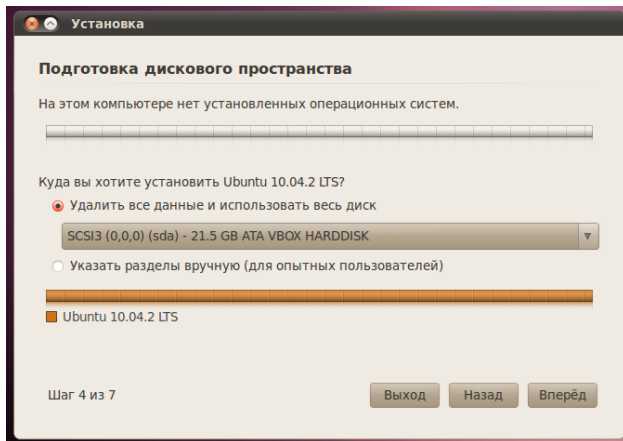


Рис. 1.8. Установка Linux, разметка жесткого диска

Первый пункт позволяют Ubuntu автоматически разметить диск для установки. При этом программа установки просто уничтожит всё содержимое жесткого диска и заново переразметит его под Ubuntu.

Однако первый вариант зачастую не подходит, поэтому вам (в целях обучения) нужен второй вариант: «Указать разделы вручную (расширенно)». Выберите его и нажмите кнопку «Вперёд».

Появится окно со списком жестких дисков (рис. 1.9).

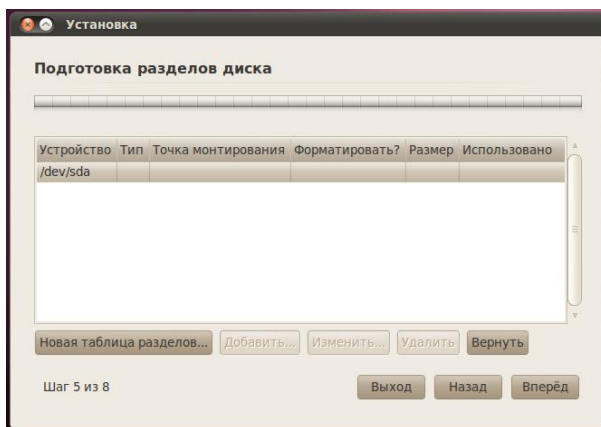


Рис. 1.9. Установка Linux, создание таблицы разделов

Выделите строку /dev/sda, нажмите кнопку «Новая таблица разделов...», появится следующее диалоговое окно (рис. 1.10).

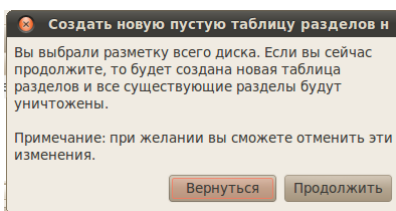


Рис. 1.10. Установка Linux, подтверждение создания новой разметки

Нажмите кнопку «Продолжить». После этого в таблице появится строка «Свободное место». Выделите её и нажмите кнопку «Добавить», после чего появится диалоговое окно (рис. 1.11).

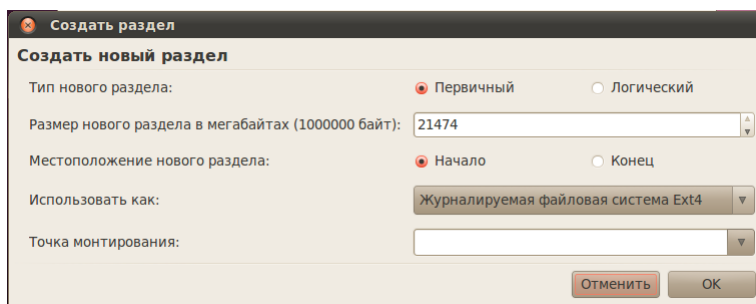


Рис. 1.11. Установка Linux, создание нового раздела

Первым делом необходимо создать системный раздел, поэтому в поле «Точка монтирования» необходимо указать «/». Кроме этого в поле «Использовать как» выбрать пункт «Журналируемая файловая система Ext4». Размер раздела «/» можно выбрать в пределах 10-15ГБ. В случае, если предполагается, что система будет использоваться в качестве рабочей станции, то больший раздел необходимо выделять под каталог «/home»; если это будет сервер, то под каталог «/var». Обязательно оставьте место для раздела подкачки «Swap», его размер должен быть не меньше размера оперативной памяти.

После создания разделов для данных, необходимо создать раздел подкачки. Для этого выберите свободное место на диске, нажмите кнопку «Создать» и в поле «Использовать как» выберите «Раздел подкачки». Точку монтирования указывать не нужно.

Далее жмём «Вперёд». В следующем окне (рис. 1.12) необходимо ввести имя первого пользователя. Указанный вами пользователь будет администратором с полным доступом к управлению системой. Уже после установки с его помощью вы сможете добавить обычных непривилегированных пользователей.

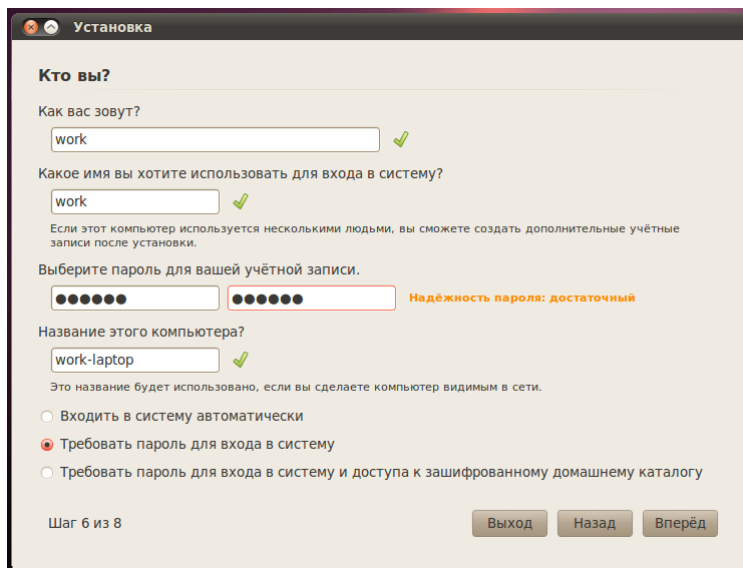


Рис. 1.12. Установка Linux, создание пользователя

Вам надо указать сначала своё имя в системе, потом свой логин и пароль. Логин желательно вводить маленькими латинскими буквами, а имя может быть любое. Имя компьютера можете оставить без изменений, а вот переключатель внизу может вас заинтересовать. Тут можно указать, будет ли ваша система спрашивать пароль при входе или нет, а так же можно выбрать третий вариант, позволяющий зашифровать ваши пользовательские данные, таким образом сделав их недоступными извне системы, при этом вам естественно всегда будет нужно вводить пароль при входе.

Остался фактически последний шаг, снова нажмите «Вперёд». Если у вас на компьютере установлены другие операционные системы, то появится окно импорта пользователей, однако через него не рекомендуется что-либо импортировать, поэтому снова спокойно жмите «Далее».

Вы увидите окно с указанием общей сводки действий для установки системы, проверьте, нет ли ошибок, если что, всегда можно вернуться назад и что-то поменять. Пока что никаких реальных операций ещё не производилось, программа установки просто собирала необходимые сведения, так что всё можно спокойно отменить. Сама установка начнётся только после нажатия на кнопку «Установить» (рис. 1.13):

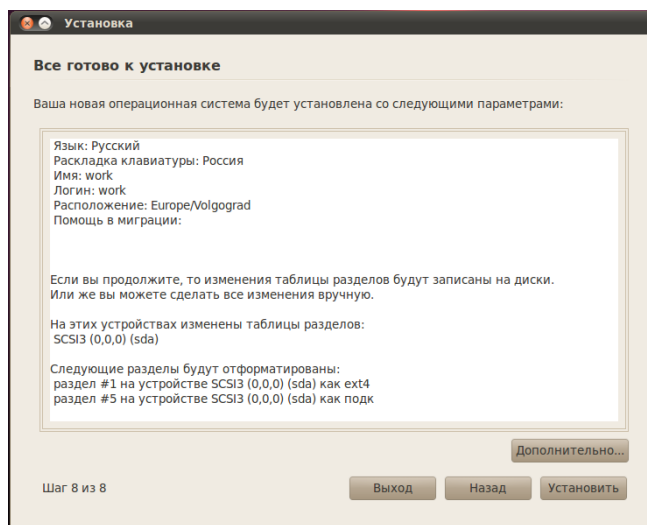


Рис. 1.13. Установка Linux, проверка сведений

Ну а теперь, пора нажать на кнопку «Установить». После того, как вы это сделаете, появится окно, показывающее ход установки (рис. 1.14).

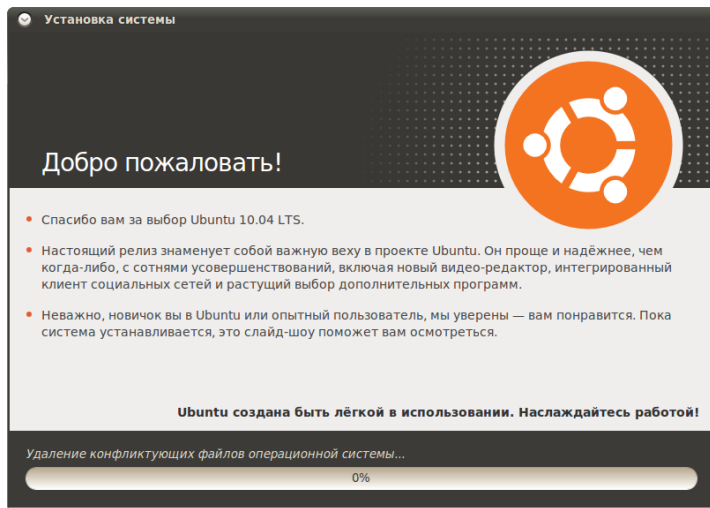


Рис. 1.14. Установка Linux, ход установки

Если Ubuntu смогла автоматически соединиться с интернетом или же если вы настроили подключение вручную, то большая часть файлов установки будет скачана с Интернета. Если вы по каким-то причинам хотите запретить ей это делать, то либо отключитесь от интернета перед установкой, либо нажмите на кнопку «Пропустить».

Для удобства работы в виртуальной машине, необходимо поставить драйвера Гостевой операционной системы. Для этого в пункте «Устройства» необходимо выбрать «Установить дополнения гостевой ОС».

Установка дополнительных виртуальных машин

Для дальнейшей работы понадобятся дополнительные виртуальные машины. Поэтому создайте ещё две виртуальные машины – с операционной системой Linux и операционной системой Windows XP.

Для соединения данных виртуальных машин по сети необходимо изменить настройки нашей виртуальной машины с Linux Ubuntu: в категории «Сеть» добавляем второй адаптер. При этом тип подключения «Внутренняя сеть», имя: «intent».

Меняем настройки виртуальной машины с Windows XP: в категории «Сеть» для адаптера 1 тип подключения «Внутренняя сеть», имя: «intent». Аналогично меняем конфигурацию виртуальной машины с Linux.

В самих операционных системах приписываем статический IP-адрес, например, для Linux Ubuntu 192.168.0.1, для Windows XP – 192.168.0.2, для второй машины с Linux – 192.168.0.3.

Задание к лабораторной работе:

1. Создайте виртуальную машину со следующими характеристиками: объем ОЗУ от 512Мб, объем жесткого диска: от 10Гб;
2. Скачайте дистрибутив Linux Ubuntu 10.04 и выше с официального сайта;
3. Установите ОС Linux Ubuntu 10.04 на виртуальный жесткий диск, при этом при разметке жесткого диска создайте два основных раздела и один раздел подкачки. Для файла-подкачки выделите место объемом 1Гб;
4. При запросе создать пользователя, введите имя пользователя «work» и пароль.

Контрольные вопросы

1. Перечислите основные характеристики VM VirtualBox.
2. Перечислите преимущества и недостатки виртуализации.
3. Какие типы разделов жесткого диска существуют?
4. Опишите принцип работы Linux с различными устройствами.
5. Назначение раздела подкачки.
6. Какие файловые системы существуют в Linux?

ЛАБОРАТОРНАЯ РАБОТА №2

Работа с пользователями. Управление правами доступа. Управление файлами и каталогами.

Ссылки.

Цель работы: Изучить возможности Linux при работе с пользователями и управлении правами доступа.

Задачи работы:

- Рассмотреть концепцию Linux при работе с пользователями;
- Изучить управление базами данных пользователей;
- Рассмотреть возможности манипулирования доступом к данным.

Теоретическая часть

Работа с пользователями

Linux - это многопользовательская операционная система. Каждый пользователь в Linux принадлежит одной основной группе и одной или нескольким дополнительным группам. В Linux, как и в большинстве других операционных системах, работа с пользователями заключается в наборе следующих манипуляций: добавление пользователя/группы, удаление пользователя/группы, модификация настроек пользователя/группы. Данные манипуляции производятся с помощью команд: `useradd`, `groupadd`, `userdel`, `groupdel`, `usermod`, `groupmod`, а так же `passwd`, `gpasswd`, `id`. Существуют так же и графические средства администрирования пользователями, обычно они расположены в оболочке X в разделе Администрирование - Пользователи и группы. Однако, при администрировании Linux использование графических оболочек не приветствуется.

UID, GID

Каждый пользователь в системе имеет свой уникальный идентификационный номер (`user-ID`, или `UID`). Также пользователи могут объединяться в группы, которые в свою очередь имеют `group-ID`, или `GID`. Чтобы узнать свой `UID` и `GID`, т.е. уникальный номер пользователя и номер группы, к которой вы принадлежите, необходимо ввести команду `id` (рис. 2.1).

```
work@work:~$ id
uid=1000(work) gid=1000(work) группы=4(adm
,20(dialout),24(cdrom),46(plugdev),105(lpada
min),119(admin),122(sambashare),1000(work)
```

Рис. 2.1. Отообразить UID и GID

Пример добавления пользователя (рис. 2.2.):

```
work@work:~$ sudo groupadd test
work@work:~$ sudo useradd -c "Test Test" -g test -m test
work@work:~$ sudo passwd test
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
work@work:~$ sudo it test
sudo: it: command not found
work@work:~$ sudo id test
uid=1001(test) gid=1002(test) groups=1002(test)
work@work:~$ sudo ls -ld /home/test/
drwxr-xr-x 2 test test 4096 2011-12-17 15:13 /home/test/
```

Рис. 2.2. Добавление нового пользователя

В примере мы добавляем группу для нового пользователя (`groupadd`), далее создаем нового пользователя с полным именем `Test Test`, имеющего основную группу `test` и логин `test`, далее задаем пароль для пользователя `test` (`passwd test`) и проверяем параметры созданного пользователя (`id` и созданный каталог пользователя `/home/test/`). На рис. 2.2 видно, что UID и GID - более 1000. Данная особенность является признаком обычного пользователя. Значения ниже (меньше) 1000 (а в некоторых версиях - меньше 500) указывают на то, что пользователь является системным пользователем.

В соответствии с соглашением, системные пользователи обычно имеют `id` меньше, чем 100, а пользователь `root` имеет `id`, равный 0. Автоматическая нумерация обычных пользователей начинается со значения `UID_MIN`, установленного в файле `/etc/login.defs`. Это значение обычно установлено в 500 или 1000.

Помимо учетных записей обычных пользователей и учетной записи пользователя `root`, в системе бывает несколько учетных записей специального назначения для демонов, таких как `FTP`, `SSH`, `mail`, `news` и т.д. Такие учетные записи часто управляют файлами, но к ним невозможно получить доступ путем обычной регистрации в системе. Поэтому обычно они имеют `login shell`, определенный как

`/sbin/nologin` или `/bin/false`, чтобы попытки зарегистрироваться в системе терпели неудачу.

Управление базами данных пользователей и групп в Linux

Основные файлы, содержащие информацию о пользователях и группах, - это четыре файла в каталоге `/etc`.

1. `/etc/passwd` - файл паролей, содержащий основную информацию о пользователях;
2. `/etc/shadow` - файл теневых шифрованных паролей, содержащий зашифрованные пароли;
3. `/etc/group` - файл групп, содержащий основную информацию о группах и принадлежащих этим группам пользователях;
4. `/etc/gshadow` - файл теневых групп, содержащий зашифрованные пароли групп.

Данные файлы редактировать обычным текстовым редактором крайне не рекомендуется. Они, обновляются при выполнении вышеуказанных команд, при этом при изменении - блокируются и синхронизируются.

Если все же есть острая необходимость в редактировании указанных файлов, то при помощи команды `vipw` можно безопасно редактировать файл `/etc/passwd`, а при помощи команды `visg` безопасно редактировать файл `/etc/group`. Эти команды блокируют необходимые файлы на то время, пока при помощи редактора `vi` будут производиться изменения. Если вы вносите изменения в файл `/etc/passwd`, команда `vipw` подскажет, что необходимо проверить, не нужно ли обновить и файл `/etc/shadow`. Подобным образом, если вы обновляете файл `/etc/group` при помощи команды `visg`, вы получите подсказку, что необходимо обновить и файл `/etc/gshadow`. Если необходимо удалить администраторов группы, необходимо использовать команду `visg`, поскольку команда `grpasswd` позволяет только добавлять администраторов.

В современных системах, файлы `passwd` и `group` не хранят пароли в открытом виде. Это сделано из соображений безопасности. Сами файлы `passwd` и `group` должны быть доступными для чтения для всех, а зашифрованные пароли - недоступными для чтения для всех. Поэтому зашифрованные пароли хранятся в теневых файлах, и эти файлы доступны для чтения только пользователю `root`. Необходимый доступ для изменения аутентификационных данных обеспечивается

при помощи `suid`-программы, которая имеет полномочия пользователя `root`, но может быть запущена любым пользователем.

Права доступа

Операционная система Linux - это многопользовательская система, которая дает огромные возможности манипулирования доступом к данным для каждого пользователя отдельно. Это позволяет гибко регулировать отношения между пользователями, объединяя их в группы, что позволит защитить данные одного пользователя от нежелательного вмешательства других.

Бессмысленно считать, что файловая система это не самая важная часть операционной системы, поскольку все данные пользователей хранятся именно в файлах.

В UNIX-подобных системах файлы также обеспечивают доступ к периферийным устройствам, дисковым накопителям, принтерам и т.п.

Права доступа к файлам

В свою очередь файлы имеют двух владельцев: пользователя (`user owner`) и группу пользователей (`group owner`). Для каждого файла есть индивидуальные права доступа, которые разбиты на три группы:

- Доступ для пользователя-владельца файла (`owner`).
- Доступ для группы-владельца файла (`group`).
- Доступ для остальных пользователей (`others`).

Для каждой категории устанавливаются три вида доступа: (`x`) - право на запуск файла, (`r`) - право на чтение файла, (`w`) - право на изменение (редактирование) файла.

Для того, чтобы увидеть права доступа к файлам необходимо ввести команду `ls` с ключом `-l` (рис. 2.3).

```
work@work:~$ ls -l Картинки/Kubuntu_leaflet.jpg
-rw-r--r-- 1 work work 658825 2010-03-26 15:21
Картинки/Kubuntu_leaflet.jpg
```

Рис. 2.3. Просмотр прав доступа к файлу

Для данного примера мы видим, что владелец имеет права на чтение, запись (первые две буквы `rw`), группа пользователей может лишь читать этот файл (следующая `r--`), остальные пользователи могут также только читать данный файл (`r--`).

Изменение прав доступа

Права пользователя могут быть изменены только владельцем файла или пользователем с правами администратора системы. Для изменения прав используется команда:

```
chmod[u|g|o|a] [+|-|=] [r|w|x] name1 [name2 ...]
```

В качестве аргументов команда принимает указание классов доступа («u» - владелец-пользователь, «g» - владелец-группа, «o» - остальные пользователи, «a» - все вышеперечисленные группы вместе), права доступа («r» - чтение, «w» - запись, «x» - выполнение) и операцию, которую необходимо произвести («+» - добавить, «-» - убрать, «=» - присвоить).

Таким образом, чтобы разрешить выполнение файла ip, который находится в директории /home/work/Загрузки всем пользователем необходимо выполнить команду (рис. 2.4):

```
work@work:~$ chmod a+x Загрузки/ip
```

Рис. 2.4. Команда, выдающая права на исполнение файла

Далее, чтобы оставить права записи только для владельца файла необходимо выполнить (рис.2.5):

```
work@work:~$ chmod go-w Загрузки/ip
```

Рис. 2.5. Команда, позволяющая оставить права записи только для владельца файла

Рассмотрим еще несколько примеров:

- chmod go=w ip - установить право на запись для всех пользователей кроме владельца;
- chmod a+x ip - предоставить право на запись для всех пользователей;
- chmod g+x-w ip - добавить для группы право на выполнения файла, но снять право на запись.

Права доступа можно представить в виде битовой строки, в которой каждые 3 бита определяют права доступа для соответствующей категории пользователей, как представлено в таблице 2.1:

Таблица 2.1

Представление прав доступа в виде битовой строки

gwx	gwx	gwx
421	421	421
user	group	others
владелец	группа	остальные

Таким образом, для команды `chmod 666 ip` имеем (рис. 2.6):

```
work@work:~$ chmod 666 Загрузки/ip
work@work:~$ ls -l Загрузки/ip
-rw-rw-rw- 1 work work 226568 2010-01-18 11:11 Загрузки/ip
work@work:~$ chmod 644 Загрузки/ip
work@work:~$ ls -l Загрузки/ip
-rw-r--r-- 1 work work 226568 2010-01-18 11:11 Загрузки/ip
```

Рис. 2.6. Пример использования команды `chmod`

Команда:

`chmod 644 имя_файла`

устанавливает «обычные» права доступа, т.е. владелец может читать и записывать в файл, а все остальные пользователи - только читать.

Особенности прав доступа для каталогов

Права доступа для каталогов не столь очевидны. Это в первую очередь связано с тем, что система трактует операции чтения и записи для каталогов отлично от остальных файлов. Право чтения каталога позволяет Вам получить имена (и только имена) файлов, находящихся в данном каталоге. Чтобы получить дополнительную информацию о файлах каталога (например, подробный листинг команды `ls -l`), системы придется «заглянуть» в метаданные файлов, что требует права на выполнения для каталога. Право на выполнение также потребуется для каталога, в который Вы захотите перейти (т.е. сделать его текущим) с помощью команды `cd`.

T-бит, SUID и SGID

Помимо стандартных «gwx» значений существуют еще и буквы «s» и «t». В действительности, битовая маска прав доступа к файлам содержит 4 группы по 3 бита в каждой. Таким образом, команда `chmod 755` это всего лишь краткая запись полной формы команды: `chmod 0755`.

T-бит обычно используется с каталогами. Обычно, когда t-бит для каталога не установлен, файл в данном каталоге может удалить

любой пользователь, имеющий доступ на запись к данному файлу. Устанавливая t-бит на каталог мы меняем это правило таким образом, что удалить файл из каталога может только владелец этого каталога или файла.

Установить t-бит можно при помощи команд:

```
chmod a+tw имя_файла
```

```
chmod 1777 имя_файла
```

Атрибуты SUID и SGID позволяют изменить права пользователя при запуске на выполнения файла, имеющего эти атрибуты.

Запускаемая программа получает права доступа к системным ресурсам на основе прав доступа пользователя, запустившего программу. Установка же флагов SUID и SGID изменяет это правило таким образом, что назначает права доступа к системным ресурсам исходя из прав доступа владельца файла. Т.е. запущенный исполняемый файл, которым владеет суперпользователь, получает права доступа к системным ресурсам на уровне суперпользователя (фактически неограниченные). При этом установка SUID приведет к наследованию прав владельца-пользователя файла, а установка SGID - владельца-группы.

Пользоваться такими мощными атрибутами как SUID и SGID нужно с крайней осторожностью, особенно подвергать пристальному вниманию программы и скрипты, владельцем которых является root (суперпользователь), т.к. это потенциальная угроза безопасности системы.

Управление файлами

В ОС Linux следует различать физическую файловую систему, которая отвечает за управление дисковым пространством и размещение файлов в физических адресах диска и логическую файловую систему, которая обеспечивает логическую структуру хранения файлов - пространство имен файлов. ОС Unix и Linux могут работать с различными физическими файловыми системами (Ext2, ext3, ufs), логическое же представление файловой системы в Unix/Linux структурировано. Все файлы в логической файловой системе располагаются в виде дерева, промежуточные вершины которого соответствуют каталогам, и листья - файлам и пустым каталогам. Реально на каждом логическом диске (разделе

физического дискового пакета) располагается отдельная иерархия каталогов и файлов. Для получения общего дерева в динамике используется «монтирование» отдельных иерархий к фиксированной корневой файловой системе в качестве ветвей общего дерева. Самым верхом иерархии является корень, который имеет предопределенное имя «/» (слэш). Этот же символ используется как разделитель имен в пути. Далее в корне находятся папки с определенными для каждого дистрибутива именами (etc, home, bin, mnt, proc и т.д.).

Полное имя файла, например, /bin/sh означает, что в корневом каталоге должно содержаться имя каталога bin, а в каталоге bin должно содержаться имя файла sh. Коротким или относительным именем файла называется имя, задающее путь к файлу от текущего рабочего каталога. В каждом каталоге содержатся два специальных имени, имя «.» - ссылка на текущий каталог, и имя «...» - ссылка «родительский» каталог данного текущего каталога, т.е. каталог, непосредственно предшествующий данному в иерархии каталогов. Так, например, для структуры, показанной на рис. 2.7 доступ к файлу file2 из текущего каталога (laba) возможен по полному имени: /home/myvar/file2 или по относительному имени: ../../../../myvar/file2.

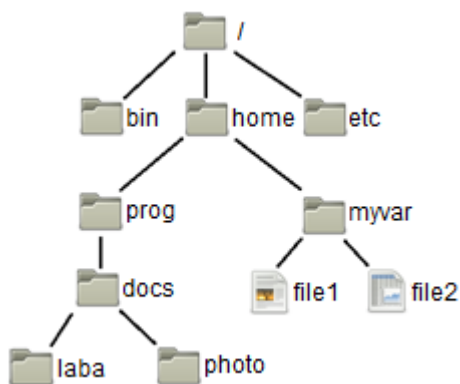


Рис. 2.7. Пример дерева каталогов

Типы файлов

ОС LINUX поддерживают несколько типов файлов:

– Обычные файлы (или регулярные) - представляют собой последовательность байтов. Это текстовые, исполняемые файлы и т.д. Данный тип файла отображается командой ls -l в виде «-» (черточка).

– Каталоги - представляют собой особый вид файлов, которые хранятся во внешней памяти подобно обычным файлам, но их структура поддерживается самой файловой системой. Данный тип файла отображается командой `ls -l` в виде символа «d».

– Специальные файлы устройств, бывают блочные и символьные. Данный тип файла отображается командой `ls -l` в виде символа «b» или «c» соответственно. Специальные файлы не хранят данные. Они обеспечивают механизм отображения физических внешних устройств в имена файлов файловой системы. Каждому устройству, поддерживаемому системой, соответствует, по меньшей мере, один специальный файл. При выполнении чтения или записи по отношению к специальному файлу, производится прямой вызов соответствующего драйвера устройства. При этом имена специальных файлов можно использовать практически всюду, где можно использовать имена обычных файлов.

– Ссылка (link). Данный тип файла отображается командой `ls -l` в виде символа «l». Файловая система UNIX/LINUX обеспечивает возможность связывания одного и того же файла с разными именами.

– Именованный программный канал (pipe) - одно из средств межпроцессных взаимодействий (IPC) в ОС UNIX/LINUX. Данный тип файла отображается командой `ls -l` в виде символа «p». Именованному программному каналу обязательно соответствует элемент некоторого каталога.

– Сокет (socket)- предоставляют весьма мощный и гибкий IPC. Данный тип файла отображается командой `ls -l` в виде символа «s». Они могут использоваться для организации взаимодействия программ на одном компьютере, по локальной сети или через Internet, что позволяет создавать распределённые приложения различной сложности. Кроме того, с их помощью можно организовать взаимодействие с программами, работающими под управлением других операционных систем.

Ссылки

Существуют жесткие и мягкие ссылки.

Жесткая ссылка является просто еще одним именем для исходного файла и не является типом файла. Она прописывается в индексном дескрипторе исходного файла (в структуре, хранящей метаданные файла). После создания жесткой ссылки невозможно различить, где исходное имя файла, а где ссылка. Если вы удаляете

один из этих файлов (точнее одно из этих имен), то файл еще сохраняется на диске (пока у него есть хоть одно имя - жесткая ссылка). Очень трудно различить первоначальное имя файла и позже созданные жесткие ссылки на него. Поэтому жесткие ссылки применяются там, где отслеживать различия и не требуется. Одно из применений жестких ссылок состоит в том, чтобы предотвратить возможность случайного удаления файла. Особенностью жестких ссылок является то, что они прямо указывают на номер индексного дескриптора, а, следовательно, такие имена могут указывать только на файлы внутри той же самой файловой системы (т. е., на том же самом носителе, на котором находится каталог, содержащий это имя).

Мягкие (символические) ссылки тоже могут рассматриваться как дополнительные имена файлов, но в то же время они представляются отдельными файлами - файлами типа мягких ссылок и являются самостоятельным типом файла. Однако блоки данных файла в системе представляются в одном экземпляре, у файла-ссылки адреса блоков данных те же, что и у исходного файла. В отличие от жестких ссылок мягкие ссылки могут указывать на файлы, расположенные в другой файловой системе, например, на монтируемом носителе, или даже на другом компьютере. Если исходный файл удален, мягкая ссылка не удаляется, но становится бесполезной. Используйте мягкие ссылки в тех случаях, когда хотите избежать путаницы, связанной с применением жестких ссылок.

Создание любой ссылки внешне подобно копированию файла, но фактически как исходное имя файла, так и ссылка указывают на один и тот же реальный файл на диске. Поэтому, например, если вы внесли изменения в файл, обратившись к нему под одним именем, вы обнаружите эти изменения и тогда, когда обратитесь к файлу по имени-ссылке.

Для создания ссылки, используется команда `ln` (рис. 2.8):

```
ln [-f] файл1 [файл2 ...] целевой_файл
```

Команда `ln` делает целевой_файл ссылкой на файл1. Файл1 не должен совпадать с целевым_файлом. Если целевой_файл является каталогом, то в нем создаются ссылки на файл1, файл2,... с теми же именами. Только в этом случае можно указывать несколько исходных файлов. Если целевой_файл существует и не является каталогом, его старое содержимое теряется. Аргументы: `-f` - удаление

существующего целевого файла; -s – создание мягкой ссылки (по умолчанию создается жесткая ссылка).

```
work@work:~$ ls
examples.desktop  sitel      Музыка
mysite           Видео     Общедоступные
Navicat          Документы Рабочий стол
PHP редактор     Загрузки  Шаблоны
shares           Картинки
work@work:~$ ln -s Картинки/Kubuntu_leaflet.jpg
work@work:~$ ls
examples.desktop  shares     Картинки
Kubuntu_leaflet.jpg sitel      Музыка
mysite           Видео     Общедоступные
Navicat          Документы Рабочий стол
PHP редактор     Загрузки  Шаблоны
```

Рис. 2.8. Пример создания ссылки

Файловый менеджер

Управление файлами также можно выполнять с помощью Midnight Commander (mc) - один из файловых менеджеров с текстовым интерфейсом типа Norton Commander для UNIX-подобных операционных систем. Запуск mc из консоли выполняется с помощью команды mc.

Установить файловый менеджер mc так, как показано на рис. 2.9.

```
work@work:~$ sudo apt-get install mc
```

Рис. 2.9. Установка mc

Достоинство mc том, что есть встроенные средства редактирования и просмотра текстовых файлов (какими являются конфигурационные файлы). При работе с mc необходимы права суперпользователя. Общая последовательность действий по редактированию конфигурационного файла:

- запустить программу (ввод команды mc);
- используя клавиши управления курсором и клавишу «Enter», добраться до нужного файла и выбрать его;
- нажатием клавиши «F4» открыть файл для редактирования;
- внести необходимые изменения;

- сохранить их (клавиша «F2»);
- выйти из режима редактирования (клавиша «F10»).

Просмотр файла выполняется аналогично, только клавишей «F3».

Выйти из mc можно тоже клавишей «F10». Общий вид mc приведен на рис. 2.10.

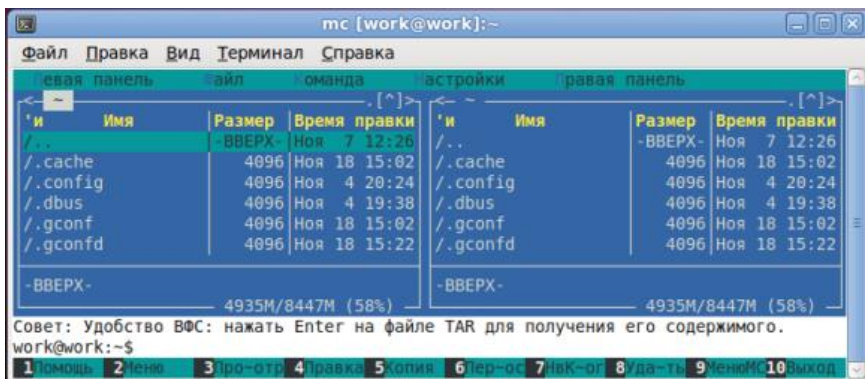


Рис. 2.10. Midnight Commander

Задания к лабораторной работе:

1. Выведите на экран UID и GID своего пользователя;
2. Создайте группу пользователей с именем usersGroup;
3. Создайте пользователя myUser в группе usersGroup;
4. Выведите на экран UID и GID пользователя myUser;
5. В своем пользователе work создайте текстовый файл, и ограничьте к нему доступ на чтение для всех других групп пользователей. Затем, зайдите в систему от имени пользователя myUser и проверьте возможность открыть этот текстовый файл;
6. От имени пользователя work изменить права доступа к данному файлу, и проверьте изменения;
7. Для одного и тоже файла создайте мягкую и жесткую ссылку в домашнем каталоге. Попробуйте создать ссылки одновременно для нескольких файлов;
8. Установите файловый менеджер mc и проверьте его работу.

Контрольные вопросы

1. Расскажите про идентификационные номера пользователей и групп в Linux.
2. Расскажите о файлах Linux, содержащих информацию о пользователях и группах системы.
3. Как система Linux хранит пароли пользователей и групп?
4. Как организуется разграничение доступа к файлам в Linux?
5. Как изменить права доступа к файлу? Как сделать это с помощью битовой строки?
6. Расскажите о T-бит, SUID и SGID.
7. Расскажите про файловые системы Linux.
8. Какие типы файлов существуют в Linux?
9. Расскажите о мягких и жестких ссылках.

ЛАБОРАТОРНАЯ РАБОТА №3

Управление сетью в Linux. Сетевые интерфейсы. Межсетевой экран.

Цель работы: Научится управлять сетевыми подключениями в ОС Linux.

Задачи работы:

- Ознакомиться с основными конфигурационными файлами сети;
- Настроить сетевые подключения различными способами;
- Изучить содержимое файла сетевой фильтрации;
- Настроить межсетевой экран.

Теоретическая часть

В ОС Linux присутствуют следующие файлы конфигурации сети вне зависимости от версии дистрибутива:

- /etc/hosts - в этом файле можно прописать IP-адреса и имена узлов локальной сети, но обычно здесь указывается только IP-адрес узла localhost (127.0.0.1), потому что сейчас даже в небольшой локальной сети устанавливается собственный DNS-сервер;
- /etc/hosts.allow – содержит IP-адреса узлов, которым разрешен доступ к сервисам данного узла;
- /etc/hosts.deny – содержит IP-адреса узлов, которым запрещен доступ к сервисам данного узла;
- /etc/iftab – содержит таблицу интерфейсов, т. е. соответствие имен интерфейсов и их MAC-адресов;
- /etc/motd – файл задает сообщение дня (Message of the day). Данный файл используется многими сетевыми сервисами, например, FTP-, SSH-серверами, которые при регистрации пользователя могут выводить сообщение из этого файла;
- /etc/resolv.conf – задает IP-адреса серверов DNS;
- /etc/services – база данных сервисов, задающая соответствие символического имени сервиса (например, pop3) и номера порта (110/tcp, tcp - это наименование протокола).

Прежде чем начать работу, убедитесь, что драйвер сетевого устройства корректно установлен, кабель (при проводном соединении) исправен и подсоединен.

Команда

```
$ sudo lshw -C network
```

позволяет посмотреть подключенные сетевые устройства. Пример вывода команды (рис. 3.1):

```
work@work:~$ sudo lshw -c network
*-network:0
    description: Ethernet interface
    product: 82540EM Gigabit Ethernet Controller
    vendor: Intel Corporation
    physical id: 3
    bus info: pci@0000:00:03.0
    logical name: eth0
    version: 02
    serial: 08:00:27:61:28:dc
    size: 1GB/s
    capacity: 1GB/s
    width: 32 bits
    clock: 66MHz
    capabilities: pm pcix bus_master cap_list ethernet phy
    sical tp 10bt 10bt-fd 100bt 100bt-fd 1000bt-fd autonegotiat
    ion
    configuration: autonegotiation=on broadcast=yes drive
    r=e1000 driverversion=7.3.21-k5-NAPI duplex=full firmware=N/
    A ip=10.0.2.15 latency=64 link=yes mingnt=255 multicast=yes
    port=twisted pair speed=1GB/s
    resources: irq:19 memory:f0000000-f001ffff ioport:d01
    0(size=8)
```

Рис. 3.1. Просмотр подключенных сетевых устройств

При выводе информации вы можете увидеть следующие свойства устройства: `description` – тип устройства, `product` – название адаптера, `vendor` - производитель устройства, `logical name` - имя сетевого интерфейса, `serial`- физический адрес устройства (mac-адрес), `driver` - используемый драйвер, `driverversion` - версия драйвера, `link` - наличие ссылки, `speed` - текущая скорость подключения.

Обратите внимание на имя сетевого интерфейса - `eth0`. Это имя будет далее применяться для настройки именно данной сетевой карты. Где `eth` обозначает что используется Ethernet интерфейс, а `0` - номер устройства. Если у вас установлено несколько сетевых устройств то соответственно им будет присвоено имена: `eth0`, `eth1`, `eth2` и т.д.

Различные сетевые утилиты, предназначенные для автоматического конфигурирования сети должны быть выключены. Для отключения запущенного Network Manager введите команду, представленную на рис. 3.2.

```
work@work:~$ sudo /etc/init.d/network-manager stop
Rather than invoking init scripts through /etc/init.d,
use the service(8)
utility, e.g. service network-manager stop

Since the script you are attempting to invoke has been
converted to an
Upstart job, you may also use the stop(8) utility, e.g.
stop network-manager
```

Рис. 3.2. Остановка NetworkManager

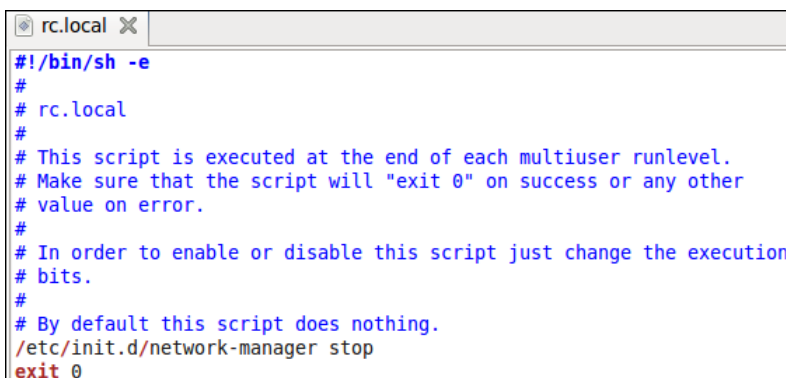
Network Manager после этого не выгрузится, но не будет видеть никаких соединений.

Отключение автоматического запуска Network Manager:

1. Откройте для редактирования файл `/etc/rc.local`, например командой:

```
$ sudo gedit /etc/rc.local
```

2. Добавьте в него (перед строкой со словом `exit 0`) выключение NM (рис. 3.3):



```
rc.local X
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.
/etc/init.d/network-manager stop
exit 0
```

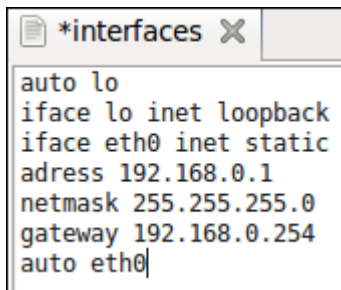
Рис. 3.3. Редактирование файла `rc.local`

Настройка проводной сети

Для настройки IP адреса, шлюза по умолчанию, маски подсети, отредактируйте файл конфигурации /etc/network/interfaces, например так:

```
$ sudo gedit /etc/network/interfaces
```

Для статического IP отредактируйте данный файл так, как представлено на рис. 3.4:



```
auto lo
iface lo inet loopback
iface eth0 inet static
address 192.168.0.1
netmask 255.255.255.0
gateway 192.168.0.254
auto eth0
```

Рис. 3.4. Настройка сетевого интерфейса

Где:

- `iface eth0 inet static` - указывает, что интерфейс (`iface eth0`) находится в диапазоне адресов IPv4 (`inet`) со статическим `ip (static)`;
- `address 192.168.0.1` - указывает что IP адрес (`address`) нашей сетевой карты `192.168.0.1`;
- `netmask 255.255.255.0` - указывает что маска подсети (`netmask`) имеет значение `255.255.255.0`;
- `gateway 192.168.0.254` - адрес шлюза (`gateway`) по умолчанию `192.168.0.254`;
- `auto eth0` - указывает системе что интерфейс `eth0` необходимо включать автоматически при загрузке системы с вышеуказанными параметрами.

`eth0` - имя подключаемого своего интерфейса. Список интерфейсов можно посмотреть набрав (рис. 3.5):

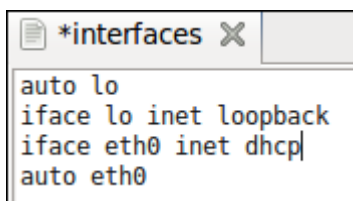
```
$ ifconfig -a
```

```
work@work:~$ ifconfig -a
eth2      Link encap:Ethernet  HWaddr 08:00:27:e8:d3:74
          BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:3 errors:0 dropped:0 overruns:0 frame:0
          TX packets:76 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1770 (1.7 KB)  TX bytes:13316 (13.3 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:78 errors:0 dropped:0 overruns:0 frame:0
          TX packets:78 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:5936 (5.9 KB)  TX bytes:5936 (5.9 KB)
```

Рис. 3.5. Вывод списка интерфейсов

Пример конфигурации для динамического IP приведен на рис. 3.6:



```
*interfaces
auto lo
iface lo inet loopback
iface eth0 inet dhcp
auto eth0
```

Рис. 3.6. Конфигурация для динамического IP

Временная настройка IP адреса и маски подсети

При необходимости задать пробные настройки, выполните (рис. 3.7):

```
work@work:~$ sudo ifconfig eth0 192.168.0.1 netmask
255.255.255.0 up
```

Рис. 3.7. Временные настройки адаптера

Где 192.168.0.1 - IP адрес, 255.255.255.0 - маска подсети. eth0 - подключаемый сетевой интерфейс.

Данные настройки пропадут после перезагрузки системы и не повлияют на файл /etc/network/interfaces.

Межсетевой экран

Ядро linux содержит подсистему (модуль) Netfilter, которая используется для управления приходящими или проходящими через сервер пакетами. Все современные брандмауэры используют эту систему для фильтрации tcp пакетов. Без интерфейса эта система мала полезна для администратора. Для управления используется iptables. При получении пакета вашим сервером, он передается Netfilter для принятия им решения: принять, обработать, или отбросить его. Таким образом, iptables это все, что нужно для управления брандмауэром. Для управления iptable существует множество программ, в том числе и ufw, которая в Ubuntu используется по умолчанию.

Ufw представляет из себя простой механизм для создания правил фильтрации пакетов IPv4 и IPv6. Данный пакет, после установки, по умолчанию отключен. Для включения ufw необходимо ввести команду:

```
work@work:~$ sudo ufw enable
Межсетевой экран активен и будет запущен при запуске системы
```

Рис. 3.8. Команда включение ufw

После запуска межсетевого экрана необходимо открыть все необходимые порты. Это делается командой, приведенной на рис. 3.9.

```
work@work:~$ sudo ufw allow 22
Правило добавлено
```

Рис. 3.9. Открытие порта

В данном примере межсетевого экран открывает 22 порт, который используется ssh. Для того, что бы закрыть открытый порт, необходимо ввести команду, приведенную на рис. 3.10.

```
work@work:~$ sudo ufw deny 22
Правило обновлено
```

Рис. 3.10. Закрытие порта

При вводе данной команды, мы получили сообщение, что правило обновлено. Так же какое-либо правило можно удалить, для этого необходимо воспользоваться командой (рис.3.11):

```
work@work:~$ sudo ufw delete deny 22
Правило удалено
```

Рис. 3.11. Удаление правила

Возможно разрешить доступ для определенных хостов или сетей. На рис. 3.12 показано как разрешить доступ хосту с ip адресом 192.168.0.2 на хост с любым ip по протоколу ssh. Если заменить 192.168.0.2 на 192.168.0.0/24 то мы разрешим протокол ssh для любого хоста этой локальной сети.

```
work@work:~$ sudo ufw allow proto tcp from 192.168.0.2 to any port 22
Правило добавлено
work@work:~$ sudo ufw allow proto tcp from 192.168.0.0/24 to any port 22
Правило добавлено
```

Рис. 3.12. Определение доступа для определенных хостов

При указании опции `--dry-run` будет выводить результат применения правила, но применяться они не будут. Например, если набрать команду:

```
sudo ufw --dry-run allow http
```

будет показана цепочка применяемых правил для открытия порта HTTP.

Для отключения `ufw`, просмотра состояния брандмауэра и вывода дополнительной информации о нем, необходимо воспользоваться соответствующими командами, приведенными на рис. 3.13.

```
work@work:~$ sudo ufw disable
Фаервол остановлен и деактивирован при загрузке
work@work:~$ sudo ufw status
Состояние: неактивен
work@work:~$ sudo ufw status verbose
Состояние: неактивен
```

Рис. 3.13. Команды остановки и просмотра состояния ufw

Если порт который вы хотите открыть или закрыть определен в файле `/etc/services`, вы можете указывать текстовое имя порта вместо его номера. Например, в приведенных выше примерах, можно заменить 22 на `ssh`.

Практическая работа

1. Выведет на экран все подключенные сетевые интерфейсы;
2. Отключите Network Manager, и отключите автоматический запуск Network Manager'a;
3. Настройте свой адаптер, задав следующие параметры:
IP: 192.168.0.1
Маска сети: 255.255.255.0
4. Включите межсетевой экран и добавьте в него правило: запретить входящий трафик по 80му порту;
5. Запретите любой исходящий трафик по 20му порту;
6. Разрешить доступ по 20му порту с ip-адреса 192.168.0.1.

Контрольные вопросы

1. Какие файлы конфигурации сети существуют в Linux?
2. Каким образом присваиваются имена интерфейсам в Linux?
3. Какого назначения модуль Ufw?
4. Каким образом осуществляется фильтрация пакетов в Linux?

ЛАБОРАТОРНАЯ РАБОТА №4

Настройка FTP-сервера. Удаленное управление операционной системой. Веб-сервер.

Цель работы: Изучить основы удаленного управления в Linux Ubuntu.

Задачи работы:

- Произвести настройки ftp-сервера;
- Настройки и использование защищенного терминала ssh;
- Получить основные сведения о протоколе Telnet;
- Установить и настроить веб-сервер Apache.

Теоретическая часть

Настройка ftp-сервера

FTP (англ. File Transfer Protocol - протокол передачи файлов) - протокол, предназначенный для передачи файлов в сетях передачи данных. FTP позволяет подключаться к серверам FTP, просматривать содержимое каталогов и загружать файлы с сервера или на сервер; кроме того, возможен режим передачи файлов между серверами.

FTP является одним из старейших прикладных протоколов, появившимся задолго до HTTP, в 1971 году. Он и сегодня широко используется для распространения программного обеспечения и доступа к удалённым хостам.

Протокол FTP относится к протоколам прикладного уровня и для передачи данных использует транспортный протокол TCP. Команды и данные, в отличие от большинства других протоколов, передаются по разным портам. Исходящий порт 20, открываемый на стороне сервера, используется для передачи данных, порт 21 для передачи команд. Порт для приема данных клиентом определяется в диалоге согласования. В случае, если передача файла была прервана по каким-либо причинам, протокол предусматривает средства для докачки файла, что бывает очень удобно при передаче больших файлов.

Vsftpd (Very Secure FTP Daemon или Очень Защищенный FTP Демон) является одним из самых простых в конфигурировании и наиболее часто используемым FTP сервером. Vsftpd обслуживает ftp серверы debian, redhat, ubuntu и прочих крупных компаний. Благодаря

предельной простоте настройки, поднятие ftp сервера с помощью vsftpd редко занимает более 5 - 10 минут.

В данной лабораторной работе предполагается показать принцип создания файлового сервера, на который все пользователи смогут складывать файлы, удалять их, создавать директории и т.д.

Установка vsftpd

Установка vsftpd приведена на рис. 4.1. Перед установкой необходимо проверить, что есть соединение с Internet.

```
work@work:~$ sudo apt-get install vsftpd
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
```

Рис. 4.1. Установка ftp

Настройка vsftpd

Конфигурирование vsftpd осуществляется редактированием файла /etc /vsftpd.conf (Рис. 4.2). Комментариев (при минимальном знании английского) обычно достаточно, чтобы разобраться что к чему:

- anon_root - директория для анонимных пользователей (/var/ftp/ по умолчанию в большинстве дистрибутивов);
- anonymous_enable - разрешить доступ анонимным пользователям;
- local_enable - разрешить доступ локальным пользователям;
- write_enable - разрешить запись;
- anon_upload_enable - разрешить запись анонимным пользователям

Таким образом, можно отредактировать эти записи в конфиге следующим образом (не стоит удалять остальные опции, если вы не знаете, что они делают):

```
#возможность работы в автономном режиме
listen=YES
#позволяем анонимных пользователей, учетки
anonymous и ftp являются синонимами
anonymous_enable=YES
#разрешаем локальных пользователей (локальные
пользователи - это те, которые
```



```
#зарегистрированы в системе, то есть на них есть
учетные записи)
local_enable=YES
#разрешаем любые формы записи на FTP сервер
write_enable=YES
#разрешаем анонимным пользователям upload
anon_upload_enable=YES
#разрешаем анонимным пользователям создавать
директории
anon_mkdir_write_enable=YES
#разрешаем анонимным пользователям
переименовывать файлы
anon_other_write_enable=YES
#у анонимов пароль спрашивать не будем
no_anon_password=YES
#директория для доступа анонимных пользователей
(если пользователь присутствует)
anon_root=/home/ftp/
#разрешаем соединение по 20 порту
connect_from_port_20=YES
#поддержка древних FTP клиентов
async_abor_enable=YES
#используем родное время, а не GMT
use_localtime=YES
#небольшое приветствие
ftpd_banner=Hello! We come in peace!
#возможность работы как фоновый процесс
background=YES
# Должны ли пользователи находится только в своих
директориях
YES/NO chroot_local_user=YES
```

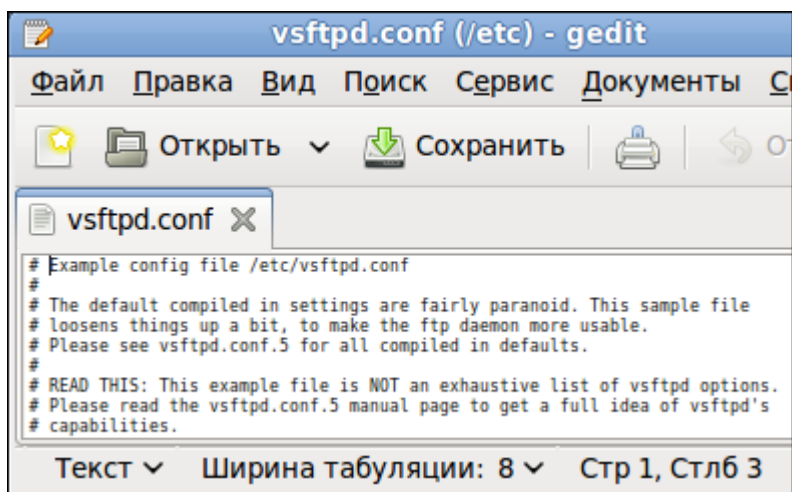


Рис. 4.2. Файл конфигурации ftp

Telnet

TELNET (англ. TErминаL NETwork) - сетевой протокол для реализации текстового интерфейса по сети (в современной форме - при помощи транспорта TCP). Название «telnet» имеют также некоторые утилиты, реализующие клиентскую часть протокола. Современный стандарт протокола описан в RFC 854.

Выполняет функции протокола прикладного уровня модели OSI.

Назначение протокола TELNET в предоставлении достаточно общего, двунаправленного, восьмибитного байт-ориентированного средства связи. Его основная задача заключается в том, чтобы позволить терминальным устройствам и терминальным процессам взаимодействовать друг с другом. Предполагается, что этот протокол может быть использован для связи вида терминал-терминал («связывание») или для связи процесс-процесс («распределенные вычисления»).

В протоколе не предусмотрено использование ни шифрования, ни проверки подлинности данных. Поэтому он уязвим для любого вида атак, к которым уязвим его транспорт, то есть протокол TCP. Для функциональности удалённого доступа к системе в настоящее время применяется сетевой протокол SSH (особенно его версия 2), при создании которого упор делался именно на вопросы

безопасности. Так что следует иметь в виду, что сессия Telnet весьма незащищена, если только не осуществляется в полностью контролируемой сети или с применением защиты на сетевом уровне (различные реализации виртуальных частных сетей). По причине ненадёжности от Telnet как средства управления операционными системами давно отказались.

Сетевой протокол ssh

SSH - это специальный сетевой протокол, позволяющий получать удаленный доступ к компьютеру с большой степенью безопасности соединения.

В основном, ssh реализован в виде двух приложений – ssh-сервера и ssh-клиента. В Ubuntu используется свободная реализация клиента и сервера ssh - OpenSSH. При подключении клиент проходит процедуру авторизации у сервера и между ними устанавливается зашифрованное соединение. OpenSSH сервер может работать как с протоколом ssh1, так и с протоколом ssh2. В настоящее время протокол ssh1 считается небезопасным, поэтому его использование крайне не рекомендуется.

Установить OpenSSH можно так:

```
work@work:~$ sudo aptitude install ssh
```

Рис. 4.3. Установка OpenSSH

Метапакет ssh содержит в себе и клиент и сервер, при этом скорее всего будет установлен только сервер, т. к. клиент часто бывает установлен в Ubuntu по умолчанию.

SSH сервер автоматически прописывается в автозагрузку при установке. Управлять его запуском/остановкой или перезапуском можно при помощи команд:

```
sudo service ssh stop|start|restart
```

Основным файлом конфигурации ssh-сервера является файл /etc/ssh/sshd_config, который должен быть доступным для чтения/редактирования только суперпользователю. После каждого изменения этого файла необходимо перезапустить ssh-сервер для применения изменений.

Сам по себе, неправильно настроенный ssh сервер - огромная уязвимость в безопасности системы, т. к. у возможного злоумышленника есть возможность получить практически

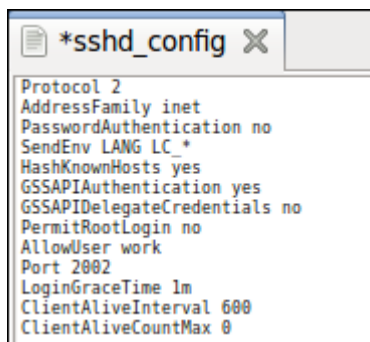
неограниченный доступ к системе. Помимо этого, у `sshd` есть много дополнительных полезных опций, которые желательно включить для повышения удобства работы и безопасности.

Для правильной настройки `ssh` с точки зрения безопасности необходимо отредактировать всего семь параметров:

1. `PermitRootLogin` – отключение возможности авторизации под суперпользователем;
2. `AllowUsers`, `AllowGroups` - предоставление доступа только указанным пользователям или группам;
3. `DenyUsers`, `DenyGroups` - блокировка доступа определенным пользователям или группам;
4. `Port` - изменение порта `SSHD`;
5. `LoginGraceTime` - изменение времени ожидания авторизации;
6. `ListenAddress` - ограничение авторизации по интерфейсу;
7. `ClientAliveInterval` - рассоединение при отсутствии активности в шелле.

Сменить стандартный порт (22) на котором слушает `sshd`. Это связано с тем, что многочисленные сетевые сканеры постоянно пытаются соединиться с 22-м портом и как минимум получить доступ путем перебора логинов/паролей из своей базы. Даже если у вас и отключена парольная аутентификация - эти попытки сильно засоряют журналы и (в большом количестве) могут негативно повлиять на скорость работы `ssh`-сервера. Если же вы по какой либо причине не желаете изменить стандартный порт вы можете использовать как различные внешние утилиты для борьбы брутфорсерами, например `fail2ban`, так и встроенные, такие как `MaxStartups`.

По умолчанию `root`-доступ разрешен. Это означает, что клиент при подключении в качестве пользователя может указать `root`, и во многих случаях получить контроль над системой. При условии, что по умолчанию в `Ubuntu` пользователь, добавленный при установке системы имеет возможность решать все административные задачи через `sudo`, создавать возможность `root` доступа к системе как минимум странно. Рекомендуется отключить эту опцию совсем.

A screenshot of a text editor window titled '*sshd_config'. The window contains the following configuration lines:

```
Protocol 2
AddressFamily inet
PasswordAuthentication no
SendEnv LANG LC_*
HashKnownHosts yes
GSSAPIAuthentication yes
GSSAPIDelegateCredentials no
PermitRootLogin no
AllowUser work
Port 2002
LoginGraceTime 1m
ClientAliveInterval 600
ClientAliveCountMax 0
```

Рис. 4.4. Файл конфигурации ssh

Разрешенная по умолчанию парольная аутентификация является практически самым примитивным способом авторизации в ssh. С одной стороны это упрощает конфигурацию и подключение новых пользователей (пользователю достаточно знать свой системный логин/пароль), с другой стороны пароль всегда можно подобрать, а пользователи часто пренебрегают созданием сложных и длинных паролей. Специальные боты постоянно сканируют доступные из интернета ssh сервера и пытаются авторизоваться на них путем перебора логинов/паролей из своей базы. Настоятельно не рекомендуется использовать парольную аутентификацию.

Как уже было сказано, ssh может работать с протоколами ssh1 и ssh2. При этом использование небезопасного ssh1 крайне не рекомендуется.

В конечном итоге файл конфигурации должен выглядеть так, как на рис. 4.4.

Для удаленного доступа с операционной системы Windows необходимо установить на ней специальный клиент – putty (рис 4.5).

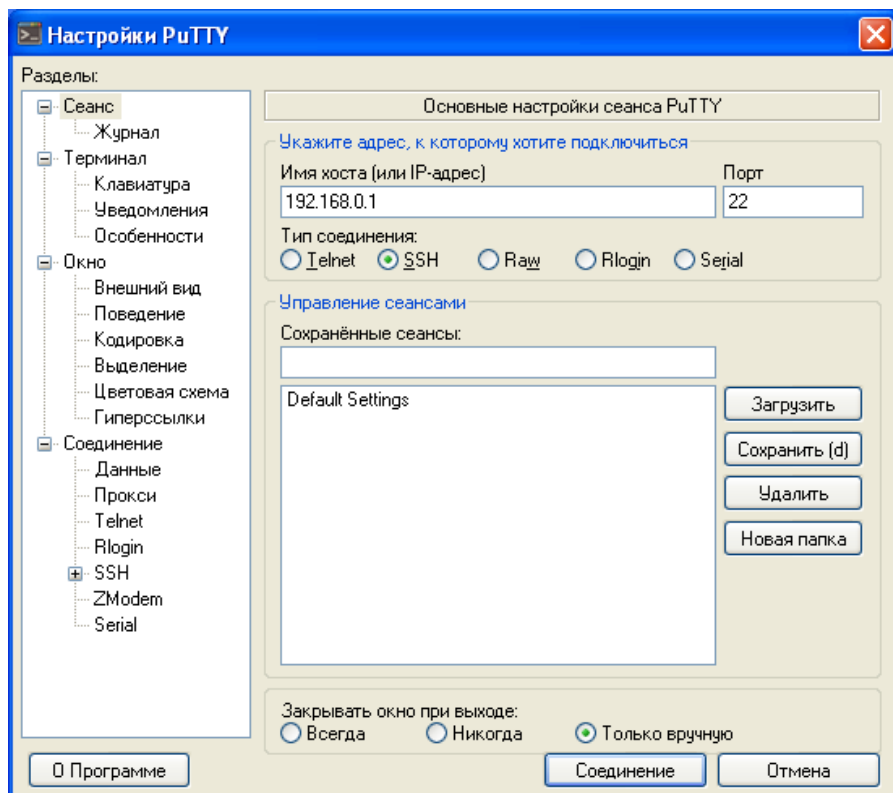


Рис. 4.5. PuTTY

Для настройки сессии введите IP хоста (192.168.0.1). Так же настройте кодировку в пункте Translation, поменяв её на UTF-8.

Веб-сервер

Apache HTTP-сервер – свободный веб-сервер. Apache является кроссплатформенным программным обеспечением, поддерживает операционные системы Linux, BSD, Mac OS, Microsoft Windows, Novell NetWare, BeOS.

Основными достоинствами Apache считаются надёжность и гибкость конфигурации. Он позволяет подключать внешние модули для предоставления данных, использовать СУБД для аутентификации пользователей, модифицировать сообщения об ошибках и т. д. Поддерживает IPv6.

Для установки apache2 введите команду, представленную на рис. 4.6.

```
work@work:~$ sudo apt-get install apache2
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
```

Рис. 4.6. Установка apache2

Файлы конфигурации Apache2 находятся в директории: /etc/apache2:

- conf.d/
- sites-available/
- sites-enabled/
- mods-available/
- mods-enabled/
- apache2.conf
- envvars
- httpd.conf
- ports.conf

В Ubuntu основным файлом настройки Apache2 является apache2.conf. Он играет роль системного файла, в котором собраны основные и самые важные настройки сервера.

Файл httpd.conf - пустой и предназначен для добавления дополнительных настроек, он включен в основной файл настройки apache2.conf

В файле envvars описаны переменные среды, необходимые для функционирования Apache-сервера.

В ports.conf вынесены настройки портов на которые можно будет подключиться к серверу или конкретному сайту на нем.

В папке conf.d находятся дополнительные конфигурационные файлы.

Для описания всех доступных сайтов используется папка sites-available в которой расположены файлы с описанием виртуальных хостов - VirtualHosts, опубликованные же сайты находятся в папке sites-enabled в виде ссылок на файлы доступных сайтов из папки sites-available.

Таким же образом в папках mods-available и mods-enabled настраивается доступность модулей используемых сервером.

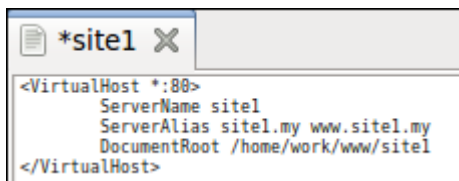
Теперь необходимо подготовить компьютер к работе веб-сервера. Прежде всего необходимо создать единую папку для всех сайтов, которые будут там размещаться, например /home/user/www. Лучшее место для такой папки это домашний каталог пользователя. Далее в этой папке необходимо создать папку сайта. Например, /home/user/www/site1. И в эту папку кинуть файлы сайта.

Следующая команда (рис. 4.7) создает запись виртуального хостинга копируя стандартную запись из файла конфигурирования Apache:

```
work@work:~$ sudo cp /etc/apache2/sites-available/default /etc/apache2/sites-available/site1
```

Рис. 4.7. Копирование файла конфигурации

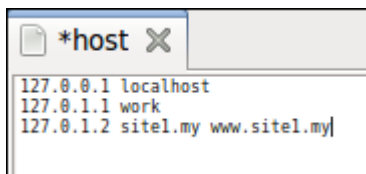
Теперь необходимо отредактировать файл, который находится по директории /etc/apache2/sites-available/site1. Необходимо настроить имя сервера, URL сервера и директорию, по которой находятся файлы сайта. После настроек файл конфигурации должен выглядеть, например, так, как на рис. 4.8.



```
*site1 X
<VirtualHost *:80>
    ServerName site1
    ServerAlias site1.my www.site1.my
    DocumentRoot /home/work/www/site1
</VirtualHost>
```

Рис. 4.8. Файл конфигурации сайта site1

Теперь необходимо как-то научить операционную систему распознавать домен .my. Для этого достаточно прописать необходимые строки в файле /etc/hosts, например так, как на рис. 4.9.



```
*host X
127.0.0.1 localhost
127.0.1.1 work
127.0.1.2 site1.my www.site1.my
```

Рис. 4.9. Редактирование файла hosts

Для начала необходимо разместить ссылку на VirtualHost в папку sites-enabled, и перечитать конфигурацию сервера Apache. Для

создания ссылки можно выполнить такую команду и перечитать параметры (рис. 4.10). После этого ваш сайт, файлы которого размещаются в директории /home/user/www/site1 будет отображаться в браузере по адресу: site1.my или www.site1.my.

```
work@work:~$ sudo a2ensite site1
Site site1 already enabled
work@work:~$ sudo /etc/init.d/apache2 reload
* Reloading web server config apache2
apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1 for ServerName
[ OK ]
```

Рис. 4.10. Активация сайта

Практическая работа

1. На виртуально машине разверните ftp-сервер;
2. Разрешите анонимный доступ для всех пользователей на данный ftp-сервер. Проверьте работу ftp-сервера с данной конфигурацией с вашей основной операционной системы;
3. Настройте разграничение прав доступа к определенным каталогам пользователей на ftp-сервере. Проверьте работу ftp-сервера с данной конфигурацией с вашей основной операционной системы;
4. Настройте смешанный режим доступа анонимных и зарегистрированных пользователей. Проверьте работу ftp-сервера с данной конфигурацией с вашей основной операционной системы;
5. Установить ssh-сервер на вашу операционную систему Linux. Настройте ssh с точки зрения безопасности;
6. На вашей основной операционной системе установить ssh-клиент (если основная операционная система Linux) или putty (если основная операционная система Windows). Проверьте работу ssh, настроив клиент соответствующим образом;
7. Установить веб сервер на Linux Ubuntu;
8. Создайте простую html-страничку. Разместите её на веб-сервере по веб-адресам: work.my и www.work.my.

Контрольные вопросы

1. Каково назначение ftp-сервера?
2. Каким образом производится настройка vsftpd?
3. Каково назначение сетевого протокола SSH?
4. Какие основные параметры рекомендуется менять при настройке SSH с точки зрения его безопасности и почему?
5. Каково назначение Telnet? Почему Telnet не рекомендуется использовать?
6. Каково назначение Apache?
7. Какие основные конфигурационные файлы Apache существуют?

ЛАБОРАТОРНАЯ РАБОТА №5

Прокси-сервер

Цель работы: Изучить работу прокси-сервера.

Задачи работы:

- Установить Squid;
- Настроить работу Squid;
- Проверить работу Squid.

Теоретическая часть

Прокси-сервер Squid

Squid - программный пакет, реализующий функцию кэширующего прокси-сервера для протоколов HTTP, FTP, Gopher и (в случае соответствующих настроек) HTTPS. Разработан сообществом как программа с открытым исходным кодом (распространяется в соответствии с GNU GPL). Все запросы выполняет как один неблокируемый процесс ввода/вывода. Используется в UNIX-системах и в операционных системах семейства Windows NT. Имеет возможность взаимодействия с Active Directory Windows Server путём аутентификации через LDAP, что позволяет использовать разграничения доступа к интернет-ресурсам пользователей, которые имеют учётные записи на Windows Server, также позволяет организовать «нарезку» интернет трафика для различных пользователей. Сервер Squid развивается в течение уже многих лет. Обеспечивает совместимость с большинством важнейших протоколов Интернета

Для установки Squid запустите в терминале следующую команду (рис. 5.1):

```
work@work:~$ sudo apt-get install squid
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
```

Рис. 5.1. Установка Squid

Конфигурация для Squid находится в файле /etc/squid/squid.conf. Оставим все настройки по умолчанию. Изменим лишь некоторые правила:

Для аутентификации пользователей добавим следующие строки:

```
auth_param basic program /usr/lib/squid/ncsa_auth
/etc/squid/internet_users
auth_param basic children 5
auth_param basic realm Enter Login/Password
auth_param basic credentialsttl 2 hours
auth_param basic casesensitive off
```

В директории `/usr/lib/squid/ncsa_auth` находится программа, для аутентификации; в директории `/etc/squid/internet_users` – список пользователей и их пароли в зашифрованном виде. Текст «Enter Login/Password» обозначает приглашение к аутентификации.

Добавим строку `http_access allow localnet`. Это имя правила. Теперь напишем и сами правила (рис. 5.2):

```
acl localnet src 10.0.0.0/8 #RFC1918 possible internal network
acl localnet src 172.16.0.0/12 #RFC1918 possible internal network
acl localnet src 192.168.0.0/24 #RFC1918 possible internal network
```

Рис. 5.2. Правила

Аналогичным образом добавим правило с именем «password» и его описание:

```
acl password proxy_auth REQUIRED
```

После этого сохраняем файл-конфигурации и перезапускаем squid следующей командой:

```
work@work:~$ sudo service squid restart
squid start/running, process 2010
```

Рис. 5.3. Перезапуск squid

Дальше проверим работу нашего прокси сервера. Например, создадим ещё одну виртуальную машину с любой операционной системой. К примеру, будем использовать Windows XP.

Меняем настройки виртуальной машины с Linux Ubuntu: в категории «Сеть» добавляем второй адаптер. При этом тип подключения «Внутренняя сеть», имя: «`intent`».

Меняем настройки виртуальной машины с Windows XP (или другой любой операционной системой): в категории «Сеть» для адаптера 1 тип подключения «Внутренняя сеть», имя: «`intent`».

В самих операционных системах приписываем статический IP-адрес, например, для Linux Ubuntu 192.168.0.1, для Windows XP – 192.168.0.2.

Далее, в Windows XP необходимо использовать подключение к LAN через прокси-сервер. В браузере Internet Explorer это можно сделать следующим образом:

Сервис > Свойство обозревателя > Подключение > Настройка LAN. Далее выделить пункт «Использовать прокси-сервер». В адресе прописать IP-адрес Ubuntu (192.168.0.1) и порт, который был прописан в файле-конфигурации squid (3128).

Практическая работа

1. Отключите межсетевой экран. Установите Squid на вашу операционную систему Linux;
2. Создайте новое правило работы Squid: разрешить доступ подсети 192.168.0.0/24;
3. Проверьте работу Squid следующим образом: создайте ещё две виртуальные машины с операционными системами Windows и Linux. Настройте второй адаптер вашей Linux Ubuntu и адаптеры у вспомогательных операционных систем. Не забудьте указать в браузерах вспомогательных операционных систем, что подключение будет осуществляться через прокси-сервер. Если у вас всё настроено правильно и в вашей Linux Ubuntu есть подключение к Internet, то и во вспомогательных операционных системах так же должен быть доступ к Internet;
4. Измените предыдущее правило работы Squid: доступ подсети 192.168.0.0/24 должен быть разрешен в период рабочего времени (с 8.00 до 17.00) и только в рабочие дни (понедельник, вторник, среда, четверг, пятница).

Контрольные вопросы

1. Каково назначение Squid?
2. Каким образом изменяются настройки работы Squid?
3. Каким образом прописываются правила работы Squid? Опишите синтаксис.

ЛАБОРАТОРНАЯ РАБОТА №6

Удаленное хранение данных.

Цель работы: Изучить технологии удаленного доступа к файлам.

Задачи работы

- Изучить работу Samba;
- Изучить работу Network File System.

Теоретическая часть

Samba

Samba - программа, которая позволяет обращаться к сетевым дискам на различных операционных системах по протоколу SMB/CIFS. Имеет клиентскую и серверную части. Является свободным программным обеспечением, выпущена под лицензией GPL.

Samba работает на большинстве Unix-подобных систем, таких как GNU/Linux, POSIX-совместимых Solaris и Mac OS X Server, на различных вариантах BSD, в OS/2. Samba включена практически во все дистрибутивы GNU/Linux, в том числе, конечно, и в Ubuntu.

Установка и настройка Samba

Для установки достаточно открыть терминал и ввести команду (рис. 6.1):

```
work@work:~$ sudo apt-get install samba
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
```

Рис. 6.1. Установка Samba

Приложение будет автоматически загружено и установлено.

Конфигурирование автономного файлового сервера

Файл конфигурации находится в директории: /etc/samba/smb.conf. Домен Active Directory есть далеко не у всех. Поэтому часто возникает необходимость организовать на Linux машине автономное файловое хранилище со своей собственной системой авторизации. Это сделать очень просто.

Основной особенностью такой организации файлового хранилища будет то, что вся информация о пользователях будет храниться в базе данных Samba, соответственно добавлять и удалять пользователей на Samba надо будет вручную.

Самое главное - определиться с используемым способом доступа к ресурсу. Для его изменения надо правильно установить значение параметра `security` в секции `[global]` файла `/etc/samba/smb.conf`. Обычно используется значение `share` или `user`.

Параметр `security` влияет на то, как будут себя вести клиенты Samba и это один из наиболее важных параметров файла `smb.conf`.

security = share

Когда клиенты присоединяются к ресурсу с `security = share` им не нужно регистрироваться с использованием действительного имени пользователя и пароля. Вместо этого, клиенты посылают информацию аутентификации (пароли) на конкретный ресурс, в тот момент, когда хотят получить доступ к этому ресурсу.

Заметьте, что при использовании уровня безопасности `share` может быть очень сложно определить какой же пользователь UNIX будет использован в конечном счете.

security = user

При этом режиме клиент должен сначала произвести вход (logon), с существующим именем пользователя и паролем (имя может быть транслировано с помощью параметра `username map`). Шифрованные пароли также могут быть использованы в этом режиме.

Заметьте, что имя запрошенного ресурса не будет послано серверу до тех пор, пока сервер не аутентифицирует клиента. Именно поэтому гостевые учетки не работают в режиме `user`, не позволяя серверу преобразовывать неопознанных пользователей в гостей.

security = domain

Такой режим будет работать только в том случае, если была использована команда `net` для добавления этого компьютера в домен Windows NT. Это требует установленного параметра `encrypted passwords = yes`.

В этом режиме Samba попытается опознать имя пользователя и пароль, передав их первичному или резервному домен контроллерам

Windows NT, т.е. сделает тоже самое, что сделал бы сервер Windows NT.

Заметьте, что реальный пользователь UNIX все-таки должен быть, потому что Samba придется проверять права доступа пользователя UNIX, к файлу в файловой системе UNIX. С точки зрения клиента, нет разницы между режимами domain и user. Это затрагивает лишь то, как сервер проводит аутентификацию, только и всего.

Имя запрошенного ресурса не будет послано серверу до тех пор, пока сервер не аутентифицирует клиента. Именно поэтому гостевые шары не работают в режиме user, не позволяя серверу преобразовывать неопознанных пользователей в гостей, чтобы сервер аутентифицировал пользователя используются параметр guest account.

security = server

В этом режиме Samba попытается определить правильность пары пользователь/пароль, передав ее другому серверу SMB, такому как NT. Если это не получится, будет работать security = user.

security = ADS

В этом режиме Samba работает как член домена AD.

Для работы в этом режиме, компьютеру, на котором запущена Samba, необходим будет установленный и настроенный Kerberos, и Samba должна быть присоединена к области AD с использованием утилиты net.

Делаем так, что для того чтобы залогинится на сервере samba обязательно нужно использовать учётную запись самой Ubuntu, то есть для того чтобы создать samba-юзера надо сделать следующее (рис. 6.2):

```
work@work:~$ cd /etc/samba
work@work:/etc/samba$ sudo smbpasswd -a work
New SMB password:
Retype new SMB password:
```

Рис. 6.2. Запись пользователя в БД SMB

Вам будет предложено ввести пароль, пользователь будет добавлен в базу, теперь необходимо включить этого пользователя:

```
work@work:/etc/samba$ sudo smbpasswd -e work
Enabled user work.
```


Рис. 6.3. Включение пользователя

Открыть доступ на какую-нибудь папку очень просто. Допустим, мы хотим сделать три доступных каталога: usershare – доступный только определённым пользователям, download – откуда можно только скачивать файлы и upload – папка для загрузки файлов на сервер. Для начала создадим три директории /home/share/usershare, /home/share/download и /home/share/upload, сделать каталог upload доступным для записи

```
sudo chmod 777 /home/share/upload
```

И запишем в конец конфигурационного файла /etc/samba/smb.conf следующий текст:

```
[usershare]
comment = User share
path = /home/shares/usershare
valid users = username
create mask = 0660
directory mask = 0771
writable = yes
[download]
comment = All users download
path = /home/share/download
create mode = 0700
directory mask = 0700
available = yes
browsable = yes
public = yes
writable = no
[upload]
comment = All users upload
path = /home/share/upload
create mode = 0777
directory mask = 0777
available = yes
browsable = yes
public = yes
writable = yes
```

Не забудьте перезапустить samba после изменений (рис. 6.4):

```
work@work:/etc/samba$ sudo service smbd restart
smbd start/running, process 2048
```

Рис. 6.4. Перезапуск Samba

Приложения для настройки

Так же существуют приложения, позволяющие производить настройку Samba через графический интерфейс (см. GUI приложения для работы с Samba).

Установить самый простой GUI для Samba можно командой (рис. 6.5):

```
work@work:~$ sudo apt-get install system-config-samba
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
```

Рис. 6.5. Установка приложения для работы с Samba

Запускается он командой (рис. 6.6):

```
work@work:~$ sudo system-config-samba
```

Рис. 6.6. Запуск приложения для работы с Samba

Все изменения он записывает в конфигурационный файл samba. Интерфейс приложения приведен на рис. 6.7.

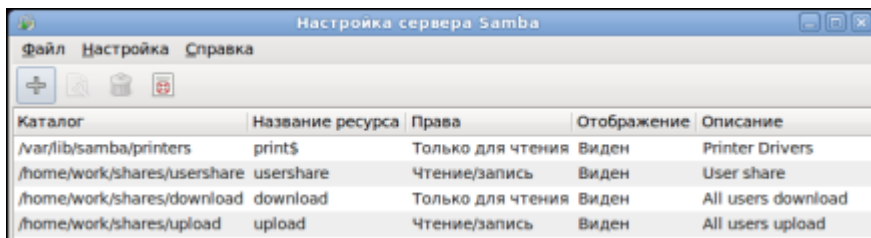


Рис. 6.7. Приложение для работы с Samba

Для удалённого администрирования Samba в качестве web-интерфейса для Samba отлично подойдёт webmin.

Network File System

Network File System (NFS) позволяет системе предоставлять в общий сетевой доступ каталоги и файлы. Посредством NFS,

пользователи и программы могут получать доступ к файлам на удаленных машинах так же легко, как будто это файлы на их локальном компьютере.

Установка и настройка NFS-сервера

Для работы NFS-сервера должны быть установлены пакеты (рис. 6.8):

- nfs-kernel-server;
- nfs-common;
- portmap.

```
work@work:~$ sudo apt-get install nfs-kernel-server nfs-common portmap
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
```

Рис. 6.8. Установка пакетов для работы с NFS

Настраиваем, какие именно директории мы хотим открыть для совместного пользования и кому. Все это делается в файле /etc/exports. Записи в этом файле имеют следующий формат:

каталог (или файловая система)
client1 (option1, option2) client2 (option1, option2)

Выделим директорию /homeshare (директория с данными на сервере) в совместное пользование всем компьютерам с IP - 192.168.0.0/24 с правами чтения и записи:

```
/homeshare
192.168.0.0/24 |( rw, sync, no_subtree_check, all_squash)
```

Рис. 6.9. Настройка NFS

Опции:

-ro - права только на чтение. Можно и не указывать, так как она установлена по умолчанию;

-rw - дает клиентам право на запись;

-async - этот параметр может повысить производительность, но он может также вызвать потерю данных при перезапуске NFS-сервера без предварительной процедуры остановки NFS-демона;

-sync - более безопасный режим. Выставлен по умолчанию;

-no_wdelay - актуален только в режиме async. Отключает задержку при записи;

-no_subtree_check - этот параметр отключает контроль подкаталогов, выполняющийся для некоторых проверок системой защиты. Например, могут проверяться права на все подкаталоги монтируемой директории. Параметр по умолчанию - разрешить контроль подкаталогов;

-noaccess - запрещает доступ к указанной директории. Может быть полезной, если перед этим задан доступ всем пользователям сети к определенной директории, и нужно ограничить доступ в поддиректории лишь некоторым пользователям.

Опции отображения пользователей:

-root_squash - этот параметр не позволяет пользователю root обращаться к смонтированному NFS-тому;

-no_root_squash - этот параметр позволяет пользователю root обращаться к смонтированному NFS-тому;

-all_squash - этот параметр, полезный для NFS-томов с открытым доступом, подавляет все UID и GID и использует только учетную запись анонимного пользователя. Установка по умолчанию - no_all_squash;

-anonuid и anongid - эти параметры меняют UID и GID анонимного пользователя на указанную учетную запись.

Установка и настройка NFS-клиента

На клиентской машине необходимо установить следующие пакеты:

- nfs-common;
- portmap.

Далее, создать точку монтирования. Для того, что бы не создавать точку монтирования каждый раз, необходимо добавить в файл /etc/fstab следующие строки:

Создаем точку монтирования (если её нет):

```
$mkdir data (мы же используем как точку  
монтирования (/home))
```

Монтировать можно двумя способами - каждый раз вручную или прописав опции монтирования в файл /etc/fstab (рис. 6.10):

```
Device mountpoint fs-type options dump fsckorder 192.168.0.1|:/homeshare /home  
nfs rw,hard,intr,sync 0 0
```

Рис. 6.10. Монтирование директории на клиентской ОС

NFS клиент может обрабатывать сбои сервера в работе. Есть две опции монтирования: `hard` и `soft`.

`soft`: Если запрос на получение файла не выполнен, NFS клиент сообщит об ошибке процессу, который пытается получить доступ к файлу. Некоторые программы умеют это обрабатывать, большая же часть - нет. Разработчики `nfs` не рекомендуют использовать эту опцию - это прямой путь к повреждённым данным и потере информации.

`hard`: Программа, осуществляющая доступ к файлу повиснет при смерти сервера. Процесс не может быть прерван или убит (только «`sure kill`»), пока вы не укажете опцию `intr`. Когда NFS сервер вернётся к работе, программа продолжит работу с того места, где остановилась. Разработчики NFS рекомендуют использование опций `hard`, `intr` со всеми монтируемые NFS файловые системы.

Компоненты Windows

Для работы Windows с NFS необходимо добавить службу Майкрософт для Network File System. Для этого сделаем следующее:

1. Установка и удаления программ > Установка компонентов Windows;
2. Другие службы доступа к файлам и принтерам сети -> Служба Microsoft для NFS.

И выбираем там:

- Администрирование служб Microsoft для NFS;
- Внешнее представление данных RPC;
- Клиент для NFS.

После установки и перезагрузки появляется возможность примонтировать NFS.

Практическая работа

1. Установите Samba на вашу Linux Ubuntu;
2. Измените настройки Samba так, что бы вход осуществлялся с существующим именем пользователя и паролем;
3. Создайте Samba-юзера, его имя и пароль должны совпадать с именем и паролем учетной записи. Включите этого пользователя.

4. Создайте каталог /home/share и в нем три каталога: usershare, download и update. Настройте их так, что бы первый был для доступа только определенных пользователей, второй - для скачанных файлов, третий - для загрузки файлов на сервер. Проверьте работоспособность в дополнительных операционных системах – Linux и Windows;
5. Установите NFS-сервер на вашу Linux Ubuntu, а NFS-клиент на вспомогательную виртуальную машину с Linux. Создайте в основной операционной системе общий каталог – homeshare;
6. Проверьте работоспособность NFS в дополнительных операционных системах – Linux и Windows.

Контрольные вопросы

1. Каково назначение Samba?
2. Каким образом производится настройка Samba?
3. Какие приложения для настройки Samba существуют?
4. Каково назначение NFS?
5. Каким образом производится настройка NFS?
6. Что такое монтирование, каким образом оно осуществляется и какие способы монтирования бывают?

ЛАБОРАТОРНАЯ РАБОТА №7

Удаленное администрирование. Нормальное название

Цель работы: Изучить основы удаленного администрирования.

Теоретическая часть

Webmin

Webmin - это графический web интерфейс для управления сервером на базе Unix подобных операционных систем. То есть установив Webmin вы можете удалённо конфигурировать и управлять Linux сервер в удобном графическом интерфейсе на русском языке через браузер.

Возможности.

- Полное управление ОС (загрузка, процессы, состояние, log-файлы);
- Управлять пользователями и группами и их правами;
- Управлять всеми серверами (Apache, FTP, SSH, Samba);
- Настройка сети в том числе и Firewall;
- Проводить резервное копирование, и многое другое;

Установка Webmin

Установить webmin можно двумя путями. Первый путь: Скачайте установочный пакет с webmin сайта www.webmin.com.

Для установки выполните (рис. 7.1).

```
work@work:~$ sudo dpkg -i webmin 1.570 all.deb
```

Рис. 7.1. Установка Webmin

В ходе установки появится сообщение об ошибке в из за отсутствия необходимых пакетов в системе.

Для устранения зависимостей выполните команду, представленную на рис. 7.2.

```
work@work:~$ sudo apt-get install -f
```

Рис. 7.2. Устранение зависимостей

Все зависимости будут разрешены. Настройка пакета webmin выполнится автоматически после разрешения зависимостей.

Второй путь устранения ошибки: необходимо добавить в список репозитория сайт `wibmin`'а. Для этого необходимо открыть для редактирования файл: `/etc/apt/sources.list`. В данный список необходимо в конец добавить следующую строку:

```
deb http://download.webmin.com/download/repository
      sarge contrib
```

Сохраните файл. Теперь можно установить Webmin привычным способом (рис. 7.3):

```
work@work:~$ sudo apt-get install webmin
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
```

Рис. 7.3. Установка Webmin

Запустить Webmin можно через браузер для этого введите следующий адрес в строку url: `https://localhost:10000`

Примите сертификат (рис. 7.4).

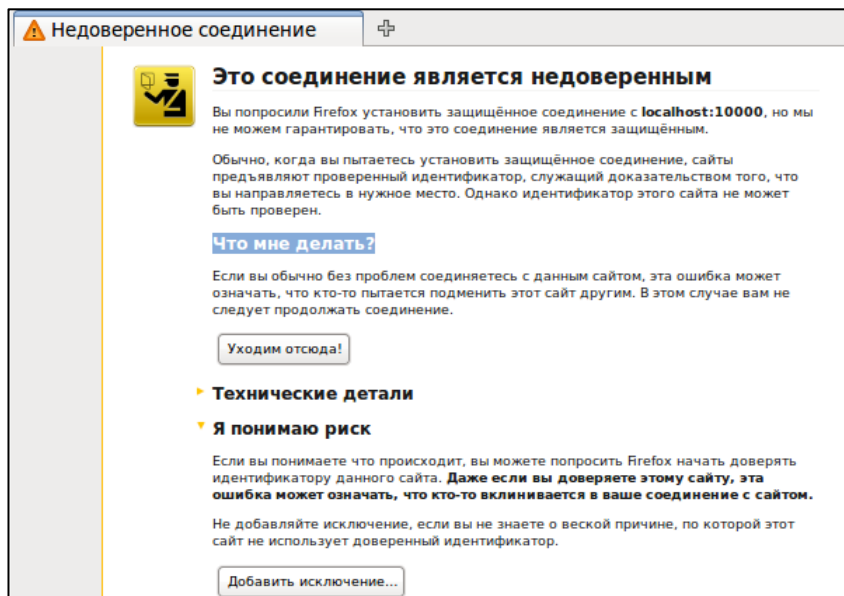
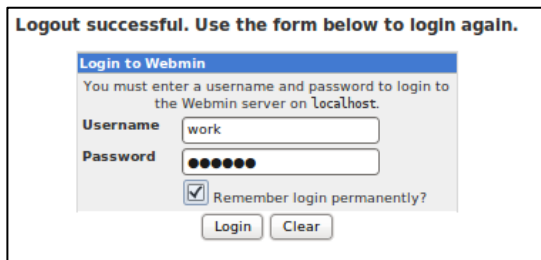


Рис. 7.4. Добавление в исключение

Интерфейс

При входе в Webmin вам будет предложено авторизоваться (рис. 7.5).



Logout successful. Use the form below to login again.

Login to Webmin

You must enter a username and password to login to the Webmin server on localhost.

Username work

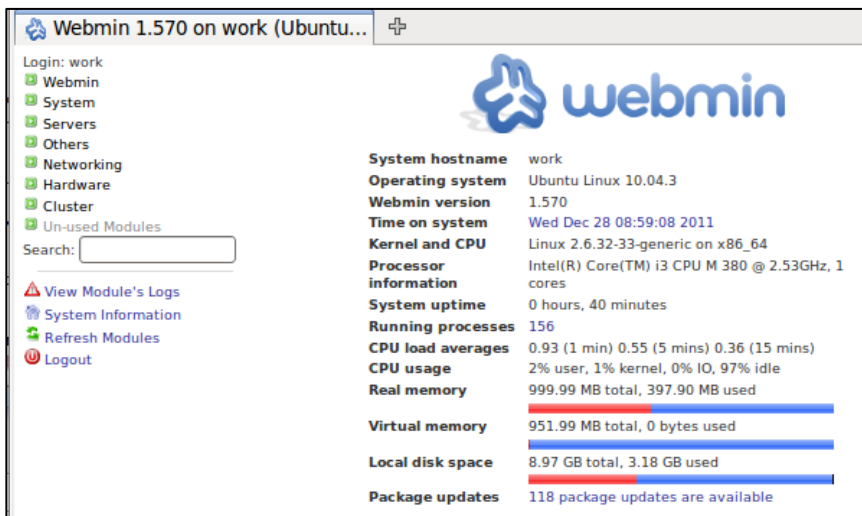
Password ●●●●●●

Remember login permanently?

Login Clear

Рис. 7.5. Авторизация

Введите логин и пароль от вашей учетной записи. Далее вы попадете на страницу управления сервером (рис. 7.6).



Webmin 1.570 on work (Ubuntu...) +

Login: work

- Webmin
- System
- Servers
- Others
- Networking
- Hardware
- Cluster
- Un-used Modules

Search:

[View Module's Logs](#)

[System Information](#)

[Refresh Modules](#)

[Logout](#)

System hostname work

Operating system Ubuntu Linux 10.04.3

Webmin version 1.570

Time on system Wed Dec 28 08:59:08 2011

Kernel and CPU Linux 2.6.32-33-generic on x86_64

Processor information Intel(R) Core(TM) i3 CPU M 380 @ 2.53GHz, 1 cores

System uptime 0 hours, 40 minutes

Running processes 156

CPU load averages 0.93 (1 min) 0.55 (5 mins) 0.36 (15 mins)

CPU usage 2% user, 1% kernel, 0% IO, 97% idle

Real memory 999.99 MB total, 397.90 MB used

Virtual memory 951.99 MB total, 0 bytes used

Local disk space 8.97 GB total, 3.18 GB used

Package updates 118 package updates are available

Рис. 7.6. Главная страница Webmin

С левой стороны находится список возможных конфигураций. Для смена языка нажмите Webmin > Выбрать язык.

Все сервисы находятся в пункте Service. Рассмотрим, например, конфигурацию Squid и просмотр журнала событий. Для начала,

установим sarg – сервис, позволяющий вести статистику по работе Squid (рис. 7.7).

```
work@work:~$ sudo apt-get install sarg
[sudo] password for work:
Чтение списков пакетов... Готово
Построение дерева зависимостей
```

Рис. 7.7. Установка sarg

Для настройки sarg зайдите в пункт Неиспользуемые модули (Un-user Modules). Там выберите Squid Report Generator (рис. 7.8). Для настройки нажмите «Настройки модуля» и введите путь для файла конфигурации: /etc/sarg/sarg.conf.

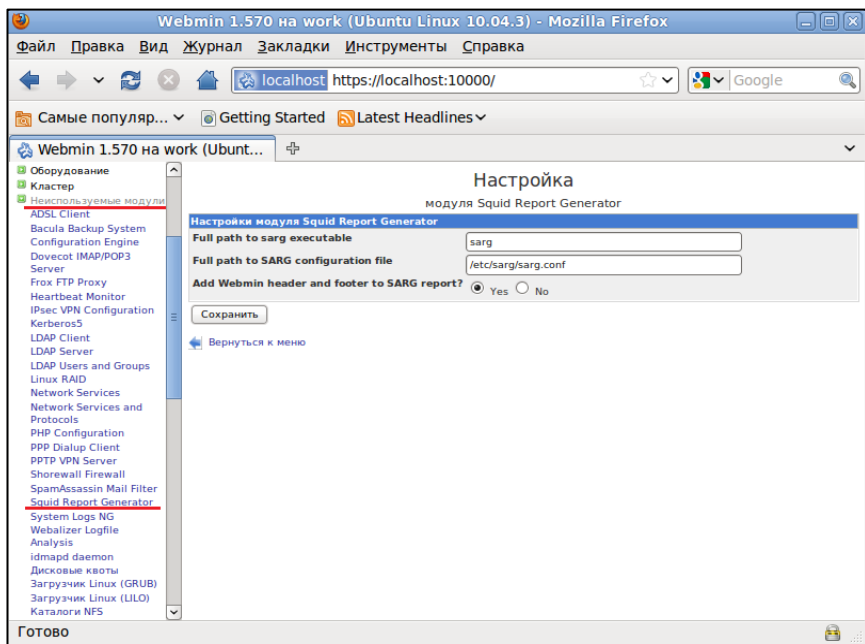


Рис. 7.8. Настройка sarg

Сохраните настройки и перезагрузить страницу. После этого Squid Report Generator появится в пункте Сервисы (Servers).

PHPmyAdmin

phpMyAdmin - это программа написанная на PHP и предназначенная для управления сервером MySQL через всемирную сеть. phpMyAdmin поддерживает широкий набор операций над MySQL. Наиболее часто используемые операции поддерживаются с помощью пользовательского интерфейса (управление базами данных, таблицами, полями, связями, индексами, пользователями, правами, и т. д.), одновременно вы можете напрямую выполнить любой SQL запрос.

phpMyAdmin обладает большим разделом документации и все пользователи приглашаются для обновления наших вики страниц для обмена идеями и способами применений различных операций. Команда phpMyAdmin постарается помочь вам при возникновении проблем, вы можете использовать различные каналы связи для получения поддержки.

Возможности phpMyAdmin

- интуитивно понятный веб-интерфейс;
- поддержка большинства функций MySQL:
 - o просмотр и удаление баз данных, таблиц, вьюшек, полей и индексов;
 - o создание, копирование, удаление, переименование и изменение баз данных, таблиц, полей и индексов;
 - o управление сервером, базами данных и таблицами, с советами по настройке сервера;
 - o выполнение, редакция и сохранение любого SQL-выражения, включая пакетные запросы;
 - o управление пользователями MySQL и их привилегиями;
 - o работа с хранимыми процедурами и триггерами.
- поддержка импорта данных из CSV и SQL;
- поддержка экспорта в различные форматы CSV, SQL, XML, PDF, ISO/IEC 26300 - OpenDocument текст и таблицы, Word, Excel, LATEX и другие;
- администрирование нескольких серверов;
- генерирование наглядных схем баз данных в виде PDF;
- создание комплексных запросов с помощью функции Запрос по шаблону;
- глобальный или частичный поиск в базе данных;

- трансформация данных в любой формат, используя набор предназначенных функций вроде отображения BLOB-данных в виде картинки или ссылки для скачивания;
- это не все, лишь часть возможностей phpMyAdmin которых, впрочем, достаточно чтобы объяснить его международную популярность.

Практическая работа

1. Установите Webmin на вашу Linux Ubuntu;
2. С помощью Webmin'a добавьте новую группу пользователей и добавьте в неё нового пользователя;
3. С помощью Webmin'a измените конфигурацию Squid;
4. С помощью Webmin'a просмотрите файловый журнал Squid.

Контрольные вопросы

1. Что такое Webmin?
2. Какие основные возможности Webmin?
3. Каково назначение PHPmyAdmin?
4. Каковы возможности PHPmyAdmin?

ПРИЛОЖЕНИЕ

Основные команды Linux

Общие команды Linux

# uname -a	Отобразить версию ядра Linux;
# lsb_release -a	Отобразить на экране информации о версии операционной системы;
# man hier	Отобразить описания иерархии файловой системы;
# clear	Осуществить очищение экрана терминала;
# wall Hello	Осуществить отправление на терминалы других пользователей сообщение «Hello»;
# date	Отобразить текущую дату и время;
# cal -3	Отобразить в удобной форме предыдущий, текущий и последующий месяц;
# uptime	Отобразить текущее время и работу системы без перезагрузки и выключения;
# hostname	Отобразить сетевое имя компьютера;
# whois linux.org	Отобразить информацию о домене имени linux.org;
# pppconfig	Осуществить создание и настройка Dial-Up соединения для выхода в Интернет по модему;
# pppoeconf	Осуществить создание и настройка выхода в Интернет через ADSL-модем;
# !!	Осуществить повторение последней выполненной команды;
# history tail - 50	Отобразить последние 50 набранных команд;
# exit	Осуществить завершение сеанса текущего пользователя;
# passwd	Осуществить изменение пароля текущего пользователя;
# shutdown -h now	Осуществить выход из Linux;
# poweroff	Осуществить выход из Linux;
# reboot	Осуществить перезагрузку системы;
# last reboot	Отобразить статистику перезагрузок;
# host site.ru	Отобразить IP-адрес введенного сайта;

Работы с файлами и директориями

# pwd	Отобразить текущий путь;
# ls	Отобразить список файлов и каталогов по порядку;
# ls -laX	Отобразить форматированный список всех файлов и директорий, включая скрытые;
# cd	Осуществить переход в домашнюю директорию;
# cd /home	Осуществить переход в директорию /home;
# touch /home/file	Осуществить создание пустого файла /home/file;
# cat /home/file	Отобразить содержимое файла /home/file;
# tail /var/log/file	Отобразить окончание файла (последние строки данного файла, которые поместятся на экран). Удобно при работе с логами и большими файлами;
# nano /home/file	Осуществить редактирование файла /home/file;
# gedit /home/file	Осуществить редактирование файла /home/file;
# echo «the End» sudo tee -a /home/file	Осуществить добавление к концу файла «the End» в файл /home/file;
# cp /home/work/primer.txt /home/primer.txt	Осуществить копирование /home/work/primer.txt в home/primer.txt;
# ln -s /home/work/primer.txt /home/primer	Осуществить создание символической ссылки /home/primer к файлу /home/work/primer.txt;
# mkdir /home/work/dir	Осуществить создание директории с именем dir;
# rmdir /home/work/dir	Осуществить удаление директории с именем dir;
# rm -rf /home/work/dir	Осуществить удаление директории с вложенными файлами;
# cp -la /dir1	Осуществить копирование директорий;

<code>/dir2</code>	
<code># mv /dir1 /dir2</code>	Осуществить переименование директории;
<code># du -sh /home/work/</code>	Отообразить на экран размер заданной директории. Можно использовать для определения размера файлов;
<code># locate primer</code>	Осуществить поиск всех файлов с именем primer;
<code># [sudo] chmod 0777 /home/</code>	Осуществить изменение прав доступа к директории только для /home. 0777 – разрешение на чтение/запись/исполнение для всех групп;
<code># [sudo] chmod -R 0777 /home/</code>	Осуществить рекурсивное изменение прав доступа к директории /home. 777 – разрешение на чтение/запись/исполнение для всех групп. Все вложенные директории и файлы будут иметь права 0777;
<code># [sudo] chown work:test /home/primer.tx t</code>	Осуществить изменение владельца и группы только для файла /home/primer.txt;
<code># [sudo] chown -R work /home/</code>	Осуществить изменение владельца для всего содержимого директории /home/;
<u>Работа с архивами</u>	
<code># tar cf primer.tar /home/primer.tx t</code>	Осуществить создание tar-архива с именем primer.tar содержащий /home/primer.txt;
<code># tar czf primer.tar.gz /home/primer.tx t</code>	Осуществить создание tar-архива с сжатием Gzip по имени primer.tar.gz;
<code># tar cjf primer.tar.bz2 /home/primer.tx t</code>	Осуществить создание tar-архива с сжатием Bzip2 по имени primer.tar.bz;
<code># tar xf primer.tar</code>	Осуществить распаковку архива primer.tar в текущую папку;
<code># tar xzf primer.tar.gz</code>	Осуществить распаковку tar-архива с Gzip;
<code># tar xjf</code>	Осуществить распаковку tar-архива с Bzip2;

primer.tar.bz

Установка программ (RPM-дистрибутивы)

rpm -qa Отобразить список установленных пакетов RPM в системе;

[sudo] rpm -i pkgname.rpm Осуществить установку RPM пакета pkgname.rpm;

[sudo] rpm -e pkgname Осуществить удаление RPM пакета pkgname;

[sudo] dpkg -i *.rpm Осуществить установку всех пакетов в директории;

Установка программ (DEB-дистрибутивы)

dpkg -l |more Отрбразить список установленных пакетов DEB в системе;

apt-cache search pack Осуществить поиск в индексах наличие доступного пакета и выводит на экран краткую информацию о пакете pack (очень полезная команда для поиска и установки программ из консоли);

apt-cache showpkg pack Отрбразить полную информация о пакете pack;

[sudo] apt-get update Осуществить обновление списка доступных пакетов из Internet;

[sudo] apt-get upgrade Осуществить обновление доступной версии установленных пакетов в системе;

[sudo] apt-get install pkgname Осуществить установку DEB пакета pkgname;

[sudo] apt-get remove pkgname Осуществить удаление DEB пакета pkgname;

[sudo] dpkg -i *.deb Осуществить установку всех пакетов в директории;

Мониторинг работы и просмотр логов

top Отобразить информацию в реальном времени о загруженных процессах, потребление ОЗУ;

dmesg Отобразить log-файл загрузки ОС и нахождения новых устройств;

mpstat 1 Отобразить расширенную статистику

	потребления ресурсов системы в процентах (для некоторых дистрибутивов необходима установка пакета sysstat);
# vmstat 2	Отобразить расширенную статистику по использованию виртуальной памяти;
# iostat 2	Отобразить расширенную статистику прерываний по устройствам;

Информация об устройствах

# lsdev	Отобразить информацию об уже установленных устройствах;
# cat /proc/cpuinfo	Отобразить полную информацию о модели процессора (частота, поддерживаемые инструкции и т.д.);
# cat /proc/meminfo	Отобразить расширенную информацию о занимаемой оперативной памяти (MemTotal, MemFree, Buffers, Cached, SwapCached, HighTotal, HighFree, LowTotal и т. д.);
# grep SwapTotal /proc/meminfo	Отобразить размер раздела выделенного под swap;
# watch -n1 'cat /proc/interrupts '	Отобразить информацию о прерываниях;
# free -m	Отобразить информацию о используемой и свободной ОЗУ и Swap-файле (-m указывает, что отображать нужно в Мб);
# lshal	Отобразить список всех устройств и их параметров;
# cat /proc/devices	Отобразить все устройства в системе (названия взяты из директории /proc/devices);
# lspci -tv	Отобразить обнаруженные PCI-устройства;
# lsusb -tv	Отобразить обнаруженные USB-устройства;
# [sudo] dmidecode	Отобразить информацию о версии BIOS компьютера;
# gtf 1024 768 75	Отобразить строку ModeLine для Вашего монитора на параметрах экрана 1024x768x75Hz;

Жесткие диски и файловая система

# fdisk -l	Отобразить информацию о всех подключенных жестких и сменных дисках;
# [sudo] hdparm -I /dev/sda	Отобразить полную информацию о IDE/ATA жестких дисках;
# smartctl -a /dev/sda1	Отобразить SMART-информацию о разделе жесткого диска /dev/sda1 (необходима установка пакета smartmontools);
# [sudo] blkid	Отобразить UUID всех доступных накопителей информации в системе;
# [sudo] hdparm -tT /dev/sda	Отобразить производительность жесткого диска;
# mount column -t	Отобразить полную информацию о примонтированных устройствах;
# cat /proc/partitions	Отобразить только примонтированные разделы жесткого диска;
# df	Отобразить свободное место на разделах;
# [sudo] mount /dev/sda1 /mnt	Осуществить монтирование раздел /dev/sda1 к точке монтирования /mnt;
# [sudo] mount -t auto /dev/cdrom /mnt/cdrom	Осуществить монтирование большинство CD-ROM'ов;
# [sudo] mount /dev/hdc -t iso9660 -r /cdrom	Осуществить монтирование IDE CD-ROM;
# [sudo] mount /dev/scd0 -t iso9660 -r /cdrom	Осуществить монтирование SCSI CD-ROM;
# [sudo] mount -t smbfs -o username=vasja ,password=pupk in //pup/Video	Осуществить монтирование сетевых ресурсов (SMB);
# [sudo] mount	Осуществить монтирование ISO-образов;

```
-t iso9660 -o
loop
/home/file.iso
/home/iso
```

```
# [sudo] mount /dev/sdb1 -t vfat -o rw /mnt
```

Осуществить монтирование раздел с файловой системой FAT 16/32 (к примеру USB-накопитель) к точки монтирования /mnt с возможностью записи;

```
# [sudo] umount /mnt
```

Осуществить демонтирование раздел от точки монтирования /mnt;

Настройка сети

```
# ifconfig
```

Отобразить параметры всех сетевых интерфейсов;

```
# ifconfig eth0
```

Отобразить параметры сетевого интерфейса eth0;

```
# [sudo] ethtool eth0
```

Отобразить состояние сетевого интерфейса. Команда ethtool применяется только для проводных подключений, не работает с беспроводными интерфейсами;

```
# [sudo] ethtool -s eth0 speed 100 duplex full autoneg off
```

Осуществить принудительное задание скорости сетевому интерфейсу 100Mbit и режим Full duplex и отключить автоматическое определение;

```
# ifconfig eth0 192.168.50.254 netmask 255.255.255.0
```

Осуществить задание основного IP-адреса сетевому интерфейсу eth0;

```
# ip addr add 192.168.50.254 /24 dev eth0
```

Осуществить задание основного IP-адреса сетевому интерфейсу eth0;

```
# ifconfig eth0:0 192.168.51.254 netmask 255.255.255.0
```

Осуществить задание дополнительного IP-адреса сетевому интерфейсу eth0;

# ip addr add 192.168.51.254 /24 dev eth0 label eth0:1	Осуществить задание дополнительного IP адреса сетевому интерфейсу eth0;
# [sudo] ifconfig eth0 up	Осуществить запуск сетевого интерфейса eth0;
# [sudo] ifconfig eth0 down	Осуществить отключение сетевого интерфейса eth0;
# ifconfig eth0 hw ether 00:01:02:03:04: 05	Осуществить смену MAC адреса;
# [sudo] /etc/init.d/dhcp d restart	Осуществить перезагрузка DHCP клиента;
# ping 192.168.0.2	Осуществить проверку сетевого соединения. Проверяется доступность IP адрес 192.168.0.2;
# route -n	Отобразить на экране таблицу маршрутизации;
# netstat -rn	Отобразить на экране таблицу маршрутизации;
# netstat -an grep LISTEN	Отобразить список всех открытых портов;
# lsof -i	Отобразить список всех открытых портов в сеть Internet;
# [sudo] netstat -tup	Отобразить активные соединения с интернетом;
# socklist	Отобразить все открытые сокеты;
# [sudo] netstat -anp --udp --tcp grep LISTEN	Отобразить список приложений, которые открывают порты;
# [sudo] iptables -L -n - v	Отобразить статус firewall (статус iptables);
# [sudo] iptables -P INPUT ACCEPT	Осуществить открытие доступа ко всем портам;
# [sudo]	Осуществить открытие доступа ко всем портам;

```
iptables -P  
FORWARD  
ACCEPT
```

```
# [sudo]          Осуществить открытие доступа ко всем портам;
```

```
iptables -P  
OUTPUT  
ACCEPT
```

```
# [sudo]          Осуществить удаление всей цепочки;
```

```
iptables -X
```

```
# [sudo]          Осуществить включение NAT на интерфейсе  
iptables -t nat - eth0;
```

```
A  
POSTROUTING  
G -o eth0 -j  
MASQUERADE
```

```
# [sudo]          Осуществить перенаправление порта 20022,  
iptables -t nat - который используется для ssh;
```

```
A  
PREROUTING  
-p tcp -d  
78.31.70.238 --  
dport 20022 -j  
DNAT --to  
192.168.16.44:  
22
```

```
# [sudo]          Осуществить перенаправление диапазона портов  
iptables -t nat - 993-995;
```

```
A  
PREROUTING  
-p tcp -d  
78.31.70.238 --  
dport 993:995 -  
j DNAT --to  
192.168.16.254  
:993-995
```

```
# iptables -L -t Осуществить проверку статуса NAT;  
nat
```

Создание и запись ISO образов

# cdrecord - scanbus	Отобразить все доступные CD-ROM;
# dd if=/dev/hdc of=/tmp/mycd.i so bs=2048 conv=notrunc	Осуществить создание ISO образов с диска CD-ROM;

Пользователи и группы

# id	Отобразить сводную информацию по текущему пользователю (логин, UID, GID, группы);
# finger work	Отобразить информацию о пользователе work;
# last	Отобразить список последних зарегистрированных пользователей;
# who	Отобразить имя текущего пользователя и время входа;
# useradd work	Осуществить добавление нового пользователя work;
# groupadd test	Осуществить добавление группы test;
# usermod -a -G test work	Осуществить добавление пользователя work в группу test;
# groupmod -A work test	Осуществить добавление пользователя work в группу test;
# userdel work	Осуществить удаление пользователя work;
# groupdel test	Осуществить удаление группы test;

Печать на принтере

# export PRINTER=lbp2900	Осуществить выбор принтера по-умолчанию. В примере выбран принтер Canon LBP-2900;
# lpr #2 name.txt	Осуществить печать на принтере Canon LBP-2900 две копии файла name.txt;
# lprm -	Осуществить удаление всех задач с принтера по умолчанию.