МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Воронежский государственный технический университет»

Кафедра радиоэлектронных устройств и систем

СЕТЕВЫЕ ТЕХНОЛОГИИ В РАДИОЭЛЕКТРОННЫХ СИСТЕМАХ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к выполнению лабораторных работ №1-4
для студентов специальности 11.05.01
«Радиоэлектронные системы и комплексы»
очной формы обучения

Воронеж 2024

УДК 621.396.6(07)
ББК 32.844я7

Составитель А. И. Сукачев

Сетевые технологии в радиоэлектронных системах: методические указания к выполнению лабораторных работ №1-4 для студентов специальности 11.05.01 «Радиоэлектронные системы и комплексы» очной формы обучения / ФГБОУ ВО «Воронежский государственный технический университет»; сост.: А. И. Сукачев. – Воронеж: Изд-во ВГТУ, 2024. – 41 с.

В соответствии с рабочими учебными программами дисциплин приведены описания методов измерений и методик выполнения лабораторных работ, изложены теоретические сведения, содержащие основы использования микропроцессорной техники. По каждой лабораторной работе в описание включены: теоретические материалы, используемое оборудование и приборы, порядок подготовки и проведения работы.

Предназначены для студентов специальности 11.05.01 «Радиоэлектронные системы и комплексы» очной формы обучения.

Методические указания подготовлены в электронном виде и содержатся в файле МУ_СТвРЭС_ЛР1-4.pdf.

Ил. 47. Табл. 1. Библиогр.: 6 назв.

УДК 621.396.6(07)
ББК 32.844я7

Рецензент – А. В. Останков, д-р техн. наук, профессор кафедры радиотехники ВГТУ

Издается по решению редакционно-издательского совета Воронежского государственного технического университета

1. ЛАБОРАТОРНАЯ РАБОТА №1
УСТАНОВКА И НАСТРОЙКА ВИРТУАЛЬНОЙ МАШИНЫ. УСТАНОВКА LINUX

1.1. ЦЕЛЬ РАБОТЫ

Установить Linux Ubuntu на виртуальную машину VM VirtualBox.

1.2. ЗАДАЧИ РАБОТЫ

Задачами работы являются:

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

1.3. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

1.3.1. 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).

Интерфейс Oracle VM VirtualBox Manager представлен на рис. 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.

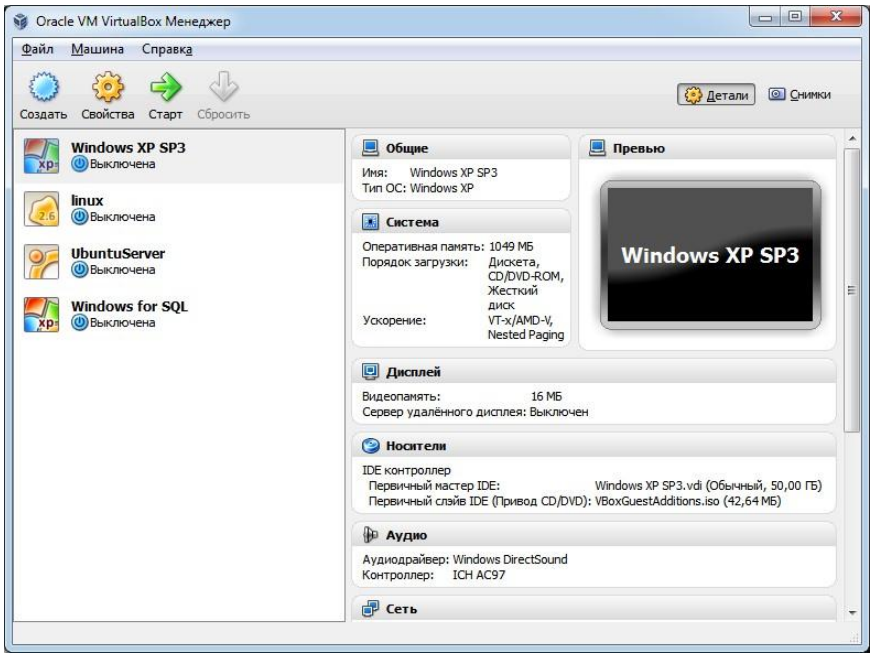

Рис. 1. Интерфейс Oracle VM VirtualBox

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 также 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 также 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.

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

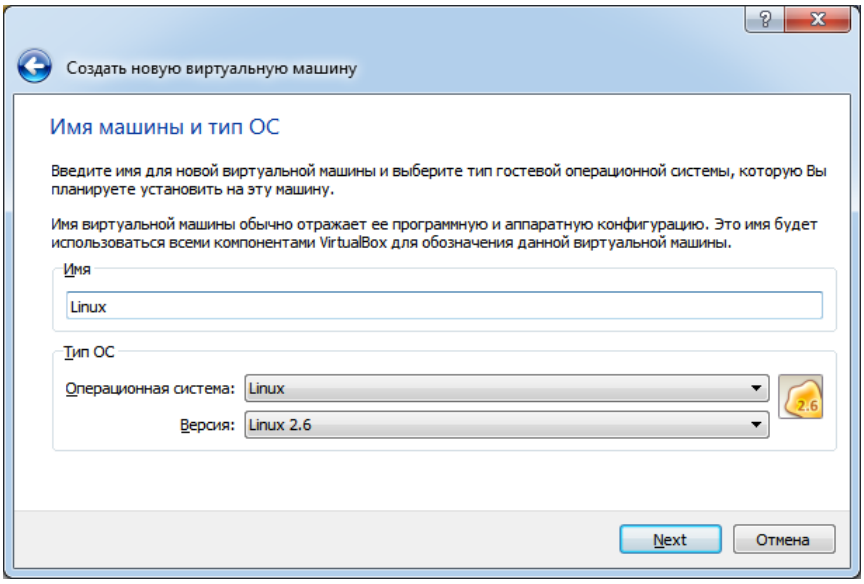

Рис. 2. Мастер создания виртуальной машины

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. 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. 3).

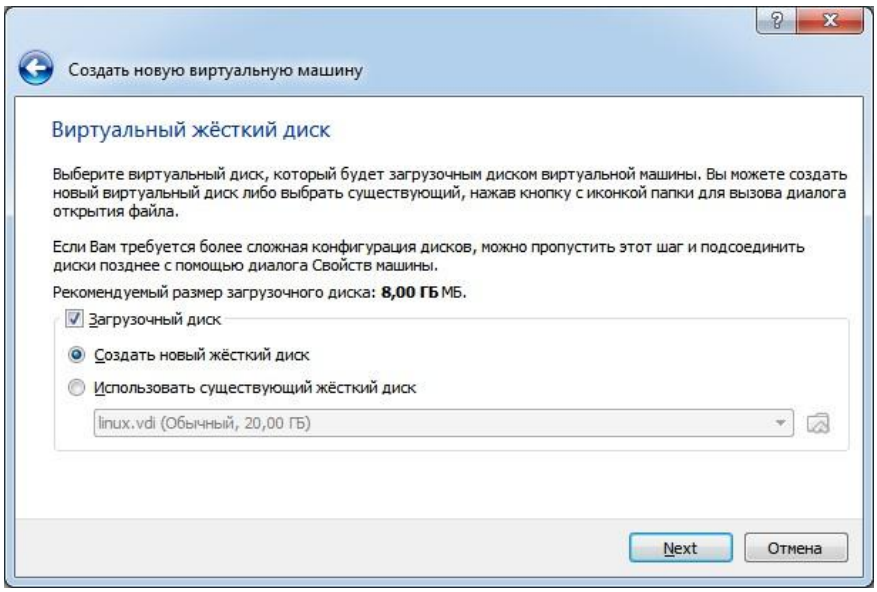

Рис. 3. Создание виртуального жесткого диска

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 будет 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. 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.

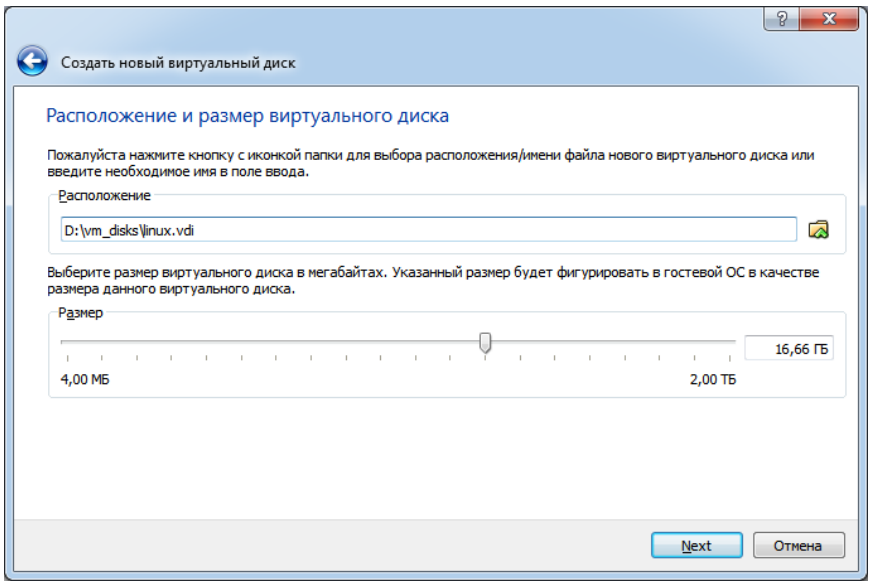

Рис. 4. Выбор директории хранения виртуального жесткого диска

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 отобразиться в списке слева.

1.3.2. 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.

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.

Linux при работе с различными устройствами и источниками данных использует иную идеологию, в отличии 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 четвёртого 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.

1.3.3 УСТАНОВКА UBUNTU

После загрузки с установочного диска появится следующее диалоговое окно (рис. 5).

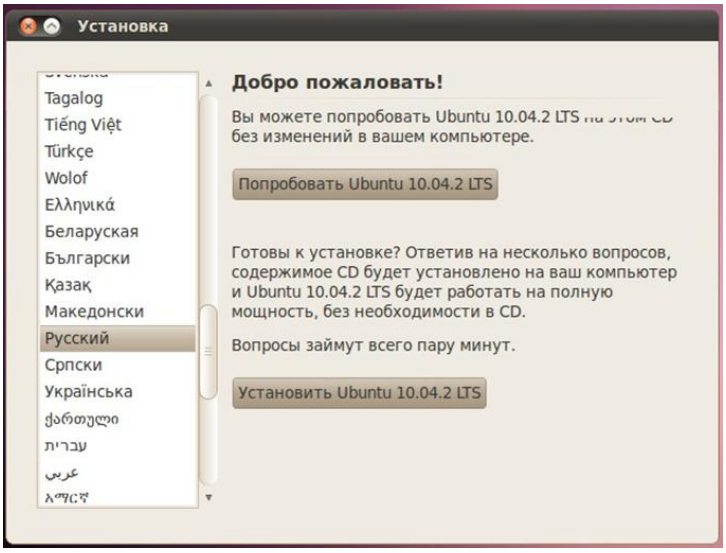

Рис. 5. Установка Linux, выбор языка

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. 6):

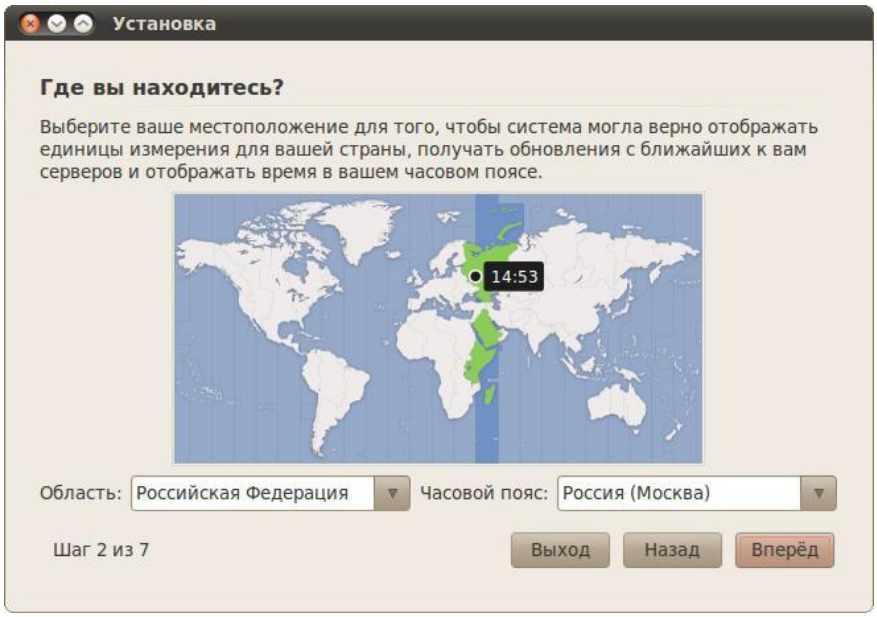

Рис. 6. Установка Linux, выбор часового пояса

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. 7).

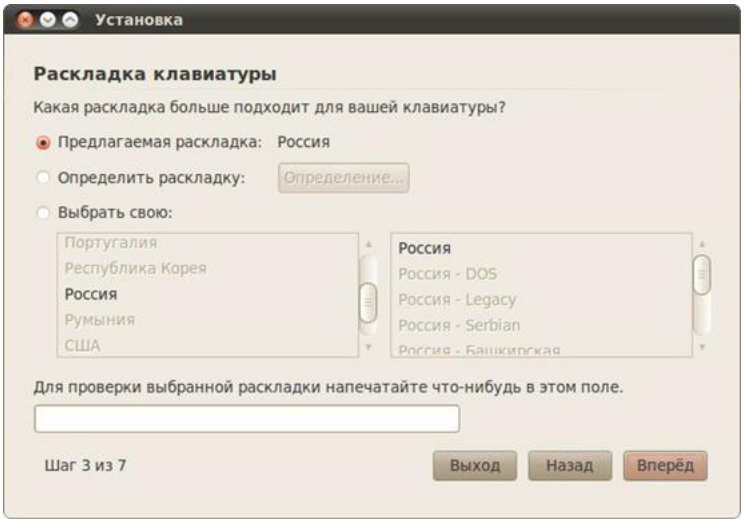

Рис. 7. Установка Linux, раскладка клавиатуры

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. 8.

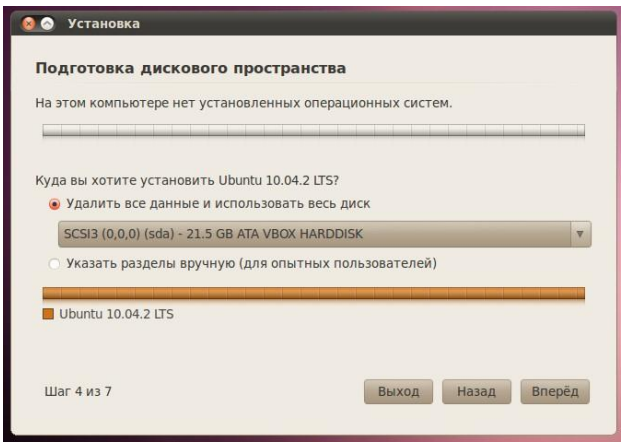

Рис. 8. Установка Linux, разметка жесткого диска

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 подходят, 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. 9).

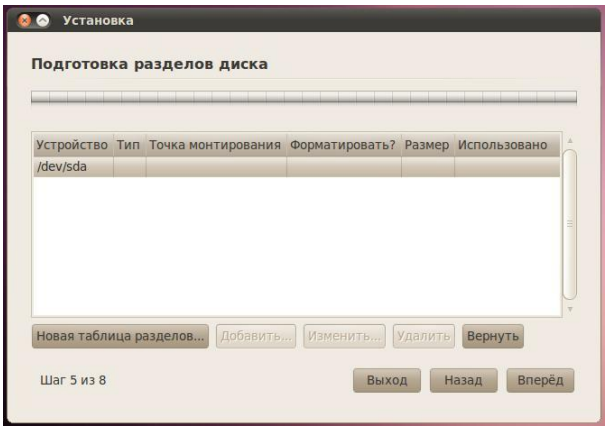

Рис. 9. Установка Linux, создание таблицы разделов

Выделите строку /dev/sda, нажмите кнопку «Новая таблица разделов…», появится следующее диалоговое окно (рис. 10).

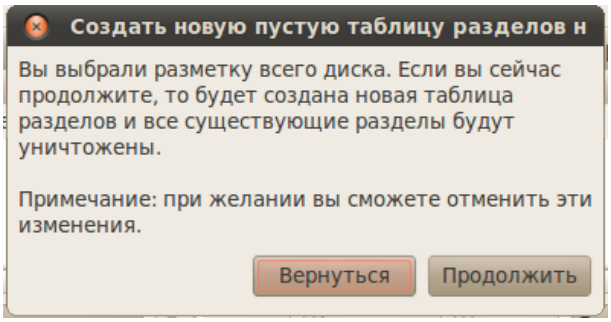

Рис. 10. Установка Linux, подтверждение создания новой разметки

Нажмите кнопку «Продолжить». После этого в таблице появится строка «Свободное место». Выделите её и нажмите кнопку «Добавить», после чего появится диалоговое окно (рис. 11).

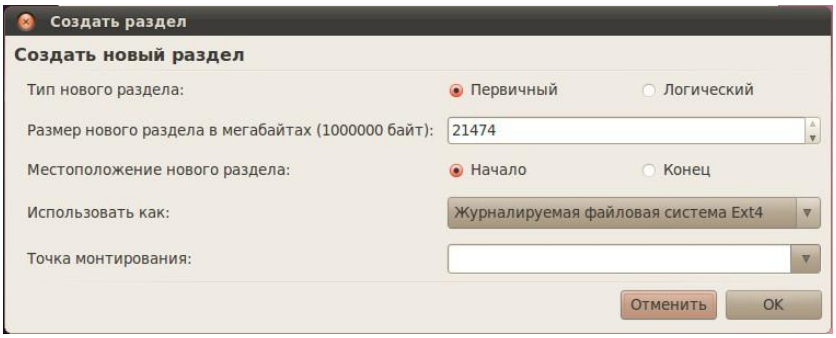

Рис. 11. Установка Linux, создание нового раздела

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. 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.

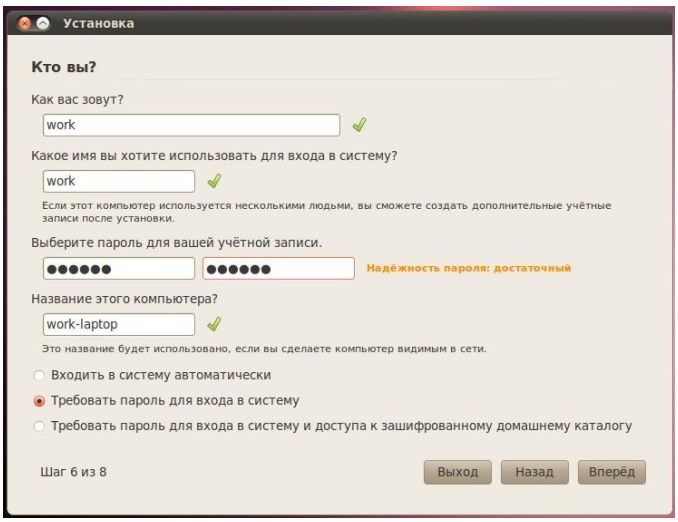

Рис. 12. Установка Linux, создание пользователя

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 также 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. 13):

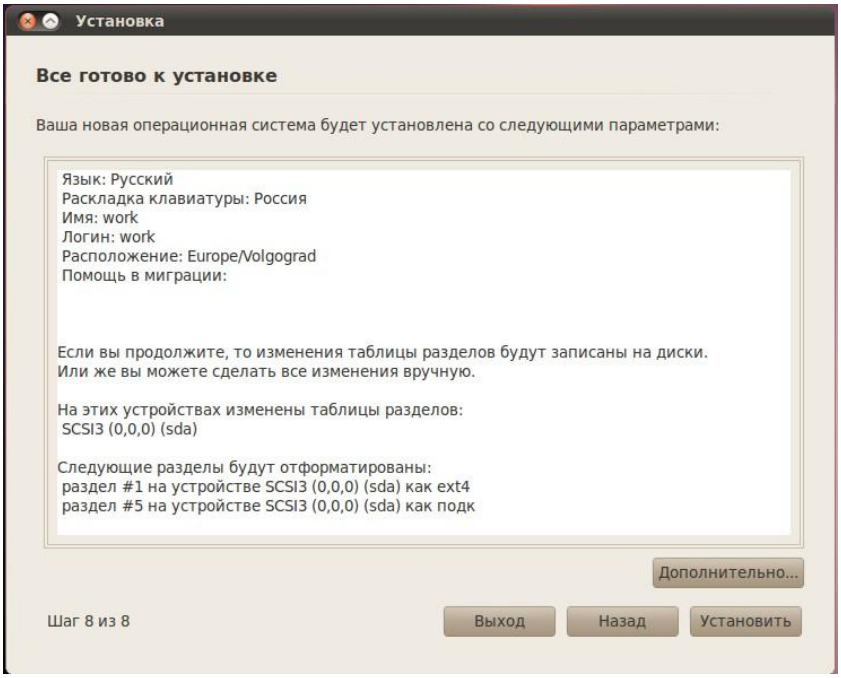

Рис. 13. Установка Linux, проверка сведений

Ну а теперь, пора нажать на кнопку «Установить». После того, как вы это сделаете, появится окно, показывающее ход установки (рис. 14).

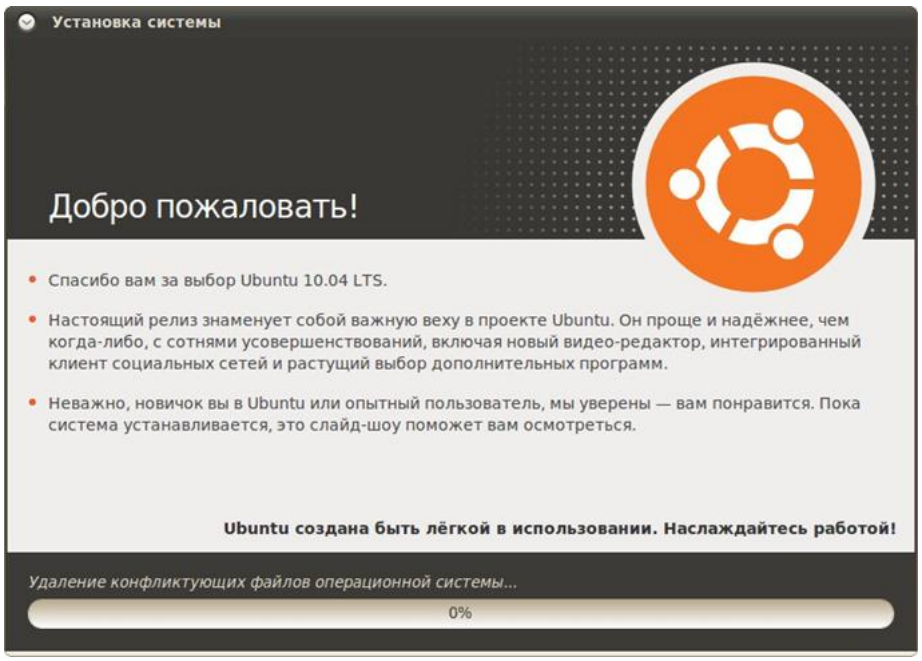

Рис. 14. Установка Linux, ход установки

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.

1.4. ЗАДАНИЕ К ЛАБОРАТОРНОЙ РАБОТЕ

Создайте виртуальную машину со следующими характеристиками: объем ОЗУ от 512Мб, объем жесткого диска: от 10Гб;

Скачайте дистрибутив Linux Ubuntu 10.04 и выше с официального сайта;

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;

При запросе создать пользователя, введите имя пользователя «work» и пароль.

1.5. КОНТРОЛЬНЫЕ ВОПРОСЫ

Перечислите основные характеристики VM VirtualBox.

Перечислите преимущества и недостатки виртуализации.

Какие типы разделов жесткого диска существуют?

Опишите принцип работы Linux с различными устройствами.

Назначение раздела подкачки.

Какие файловые системы существуют в Linux?

2. ЛАБОРАТОРНАЯ РАБОТА №2
РАБОТА С ПОЛЬЗОВАТЕЛЯМИ. УПРАВЛЕНИЕ ПРАВАМИ ДОСТУПА. УПРАВЛЕНИЕ ФАЙЛАМИ И КАТАЛОГАМИ. ССЫЛКИ

2.1. ЦЕЛЬ РАБОТЫ

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

2.2. ЗАДАЧИ РАБОТЫ

Рассмотреть концепцию Linux при работе с пользователями;

Изучить управление базами данных пользователей;

Рассмотреть возможности манипулирования доступом к данным.

2.3. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

2.3.1. 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 также 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.

2.3.2. 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. 15).

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

Рис. 15. Отобразить UID и GID

Пример добавления пользователя (рис. 16):

```
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. 16. 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. 16 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.

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

- /etc/passwd - файлпаролей, содержащий основную информацию о пользователях;

- /etc/shadow – файл теневых шифрованных паролей, содержащий зашифрованные пароли;
- /etc/group - файл групп, содержащий основную информацию о группах и принадлежащих этим группам пользователях;
- /etc/gshadow – файл теневых групп, содержащий шифрованные 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 указанный файлов, то при помощи команды vipw можно безопасно редактировать файл /etc/passwd, а при помощи команды vigr безопасно редактировать файл /etc/group. Эти команды заблокируют 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 vigr, вы получите подсказку, что необходимо обновить и файл /etc/gshadow. Если необходимо удалить администраторов группы, необходимо использовать команду vigr, поскольку команда gpasswd 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.

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. 17).

```
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. 17. 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--).

2.3.3. 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 находиться в директории /home/work/Загрузки всем пользователем необходимо выполнить команду (рис. 18):

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

Рис. 18. Команда, выдающая права на исполнение файла

Далее, чтобы оставить права записи только для владельца файла необходимо выполнить (рис. 19):

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

Рис. 19. Команда, повзоляющая 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 1:

Таблица 1

Представление прав доступа в виде битовой строки

rwx	rwx	rwx
421	421	421
user	group	others
владелец	группа	остальные

Таким образом, для команды chmod 666 ip имеем (рис. 20):

```
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. 20. Пример испоьлзования 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.

2.4. 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.

2.4.1. T-БИТ, SUID И SGID

Помимо стандартных «rwx» 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 ката-

лог 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.

2.5. 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. 21 доступ к файлу file2 из текущего каталога (laba) возможен по полному имени:

/home/myvar/file2 или по относительному имени: .../.../.../myvar/file2.

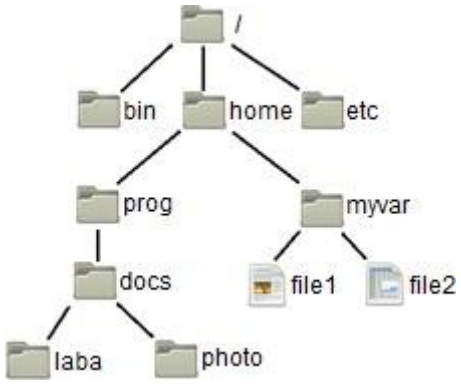

Рис. 21. Пример дерева каталогов

2.5.1. 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 «-» (черточки).

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 ло-

кальной 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.

2.5.2. 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. 22): 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 - уда-

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

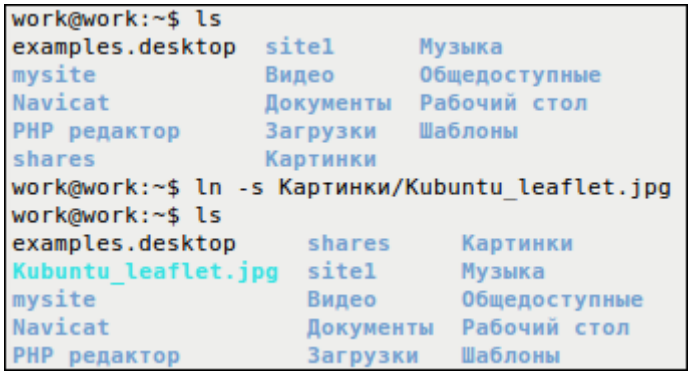

Рис. 22. Пример создания ссылки

2.6. 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. 23.

Рис. 23. 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. 24.

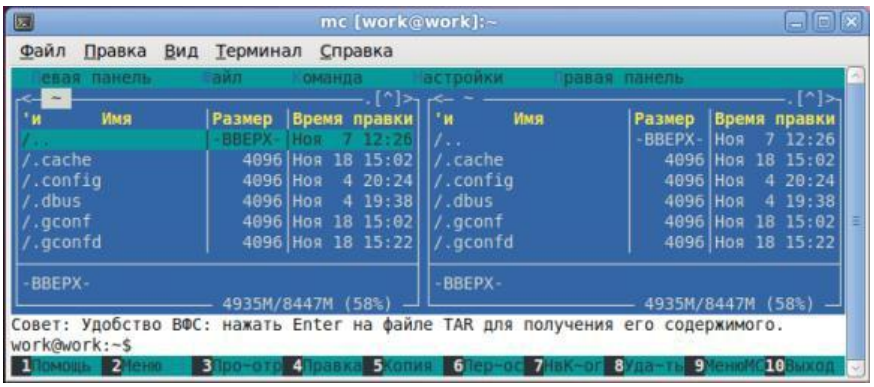

Рис. 24. Midnight Commander

2.7. 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.

2.8. КОНТРОЛЬНЫЕ ВОПРОСЫ

Расскажите про идентификационные номера пользователей и групп в Linux.

Расскажите о файлах Linux, содержащих информацию о пользователях и группах системы.

Как система Linux хранит пароли пользователей и групп?

Как организуется разграничение доступа к файлам в Linux?

Как изменить права доступа к файлу? Как сделать это с помощью битовой строки?

Расскажите о T-бит, SUID и SGID.

Расскажите про файловые системы Linux.

Какие типы файлов существуют в Linux?

Расскажите о мягких и жестких ссылках.

3. ЛАБОРАТОРНАЯ РАБОТА №3
УПРАВЛЕНИЕ СЕТЬЮ В LINUX. СЕТЕВЫЕ ИНТЕРФЕЙСЫ. МЕЖСЕТЕВОЙ ЭКРАН

3.1. ЦЕЛЬ РАБОТЫ

Научится управлять сетевыми подключениями в ОС Linux.

3.2. ЗАДАЧИ РАБОТЫ

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

3.3. 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 символьного 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. 25):

```
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 ph
ysical 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. 25. 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. 26.

```
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. 26. 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. 27):
3.

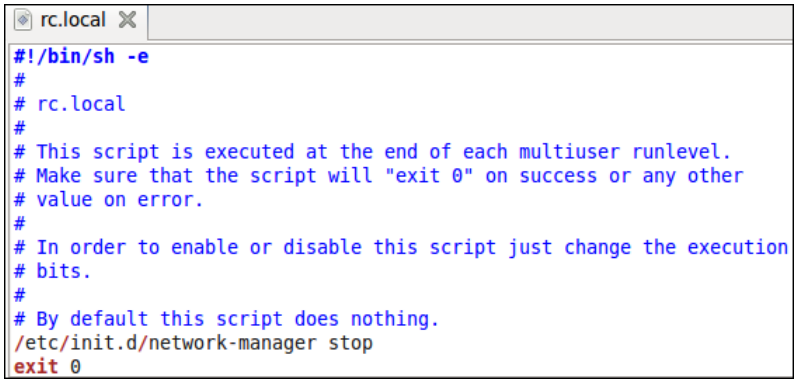

Рис. 27. Редактирование файла rc.local

3.4. 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. 28:

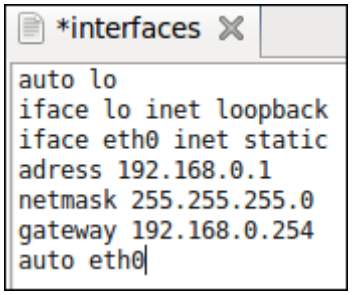

Рис. 28. Настройка сетевого интерфейса

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 - указывет системе что интерфейс eth0 необходимо включать автоматически при загрузке системы с вышеуказанными параметрами.

eth0 - имя подключаемого своего интерфейса. Список интерфейсов можно посмотреть, набрав (рис. 29):

$ 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. 29. Вывод списка интерфейсов

Пример конфигурации для динамического IP приведен на рис. 30:

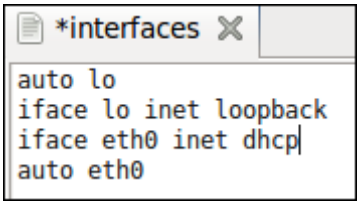

Рис. 30. Конфигурация для динамического IP

3.4.1. ВРЕМЕННАЯ НАСТРОЙКА IP АДРЕСА И МАСКИ ПОДСЕТИ

При необходимости задать пробные настройки, выполните (рис. 31).

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

Рис. 31. 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.

3.4.2. 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 за
пуске системы
```

Рис. 32. Команда включение ufw

После запуска межсетевого экрана необходимо открыть все необходимые порты. Это делается командой, приведенной на рис. 33.

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

Рис. 33. Открытие порта

В данном примере межсетевой экран открывает 22 порт, который используется ssh. Для того, что бы закрыть открытый порт, необходимо ввести команду, приведенную на рис. 34.

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

Рис. 34. 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. 35):

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

Рис. 35. Удаление правила

Возможно разрешить доступ для определенных хостов или сетей. На рис. 36 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. 36. 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. 37.

```
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. 37. 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.

3.5. 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
4. Маска сети: 255.255.255.0
5. Включите межсетевой экран и добавьте в него правило: запретить входящий трафик по 80му порту;
6. Запретите любой исходящий трафик по 20му порту;
7. Разрешить доступ по 20му порту с ip-адреса 192.168.0.1.

3.6. КОНТРОЛЬНЫЕ ВОПРОСЫ

Какие файлы конфигурации сети существуют в Linux?
Каким образом присваиваются имена интерфейсам в Linux?
Какого назначение модуля Ufw?
Каким образом осуществляется фильтрация пакетов в Linux?

4. ЛАБОРАТОРНАЯ РАБОТА №4 НАСТРОЙКА FTP-СЕРВЕРА. УДАЛЕННОЕ УПРАВЛЕНИЕ ОПЕРАЦИОННОЙ СИСТЕМОЙ. ВЕБ-СЕРВЕР

4.1. ЦЕЛЬ РАБОТЫ

Изучить основы удаленного управления в Linux Ubuntu.

4.2. ЗАДАЧИ РАБОТЫ

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

4.3. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

4.3.1. 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.

4.3.2. УСТАНОВКА VSFTPD

Установка vsftpd приведена на рис. 38. 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. 38. Установка ftp

4.3.3. НАСТРОЙКА VSFTPD

Конфигурирование vsftpd осуществляется редактированием файла /etc /vsftpd.conf (Рис. 39). 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
#зарегестрированы 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

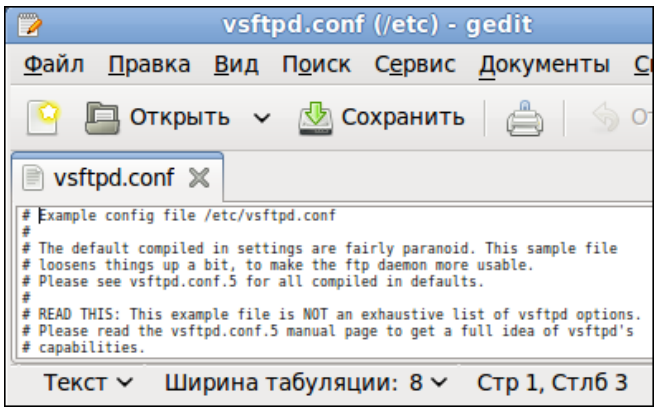

Рис. 39. Файл конфигурации ftp

4.4. TELNET

TELNET (англ. TErminaL 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 беззащитна, 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.

4.5. 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. 40. 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.

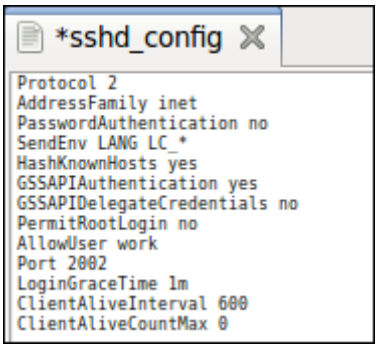

Рис. 41. Файл конфигурации ssh

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. 42.

Для удаленного доступа с операционной системы Windows необходимо установить на ней специальный клиент – putty (рис 43).

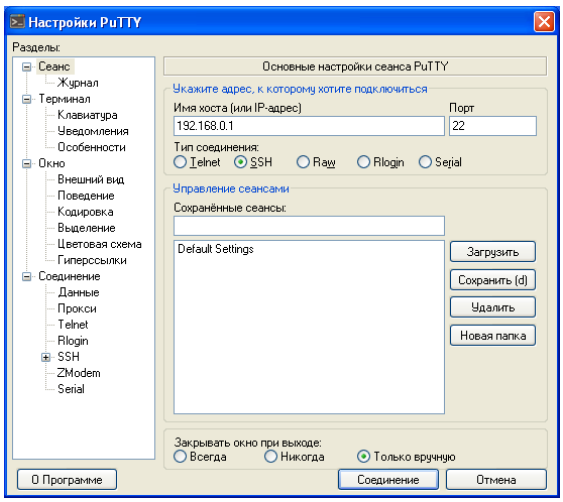

Рис. 42. PuTTy

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

4.6. ВЕБ-СЕРВЕР

Apache HTTP-сервер – свободный веб-сервер. Apache является кросс-сплатформенным 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. 43.

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

Рис. 43. 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. 44) создает запись виртуального хостинга копируя стандартную запись из файла конфигурирования Apache:

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

Рис. 44. 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. 45.

*site1

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

Рис. 45. 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. 46.

*host

```
127.0.0.1 localhost
127.0.1.1 work
127.0.1.2 site1.my www.site1.my
```

Рис. 46. 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. 47). 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 serve
r's fully qualified domain name, using 127.0.1.
1 for ServerName
                                          [ OK ]
```

Рис. 47. Активация сайта

4.7. 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 кактологам пользователей на ftp-сервере. Проверьте работу ftp-сервера с данной конфигурацией с вашей основной операционной системы;
4. Настройте смешанный режим доступа анонимных и зарегестрированных пользователей. Проверьте работу ftp- сервера с данной конфигурацией с вашей основной операционной системы;
5. Установить ssh-сервер на вашу операционную систему Linux. Настройет ssh с точки зрения безопасности;
6. На вашей основной операционной системе установить ssh- клиент (если основная операционная система Linux) или pytty (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.

4.8. 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 настройки 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?

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Блек Ю. Сети ЭВМ: Протоколы, стандарты, интерфейсы / Пер. с англ. – М.:Мир, 1990. – 506с.
2. Дж.Мартин. Вычислительные сети и распределенная обработка данных: программное обеспечение, методы и архитектура / Пер. с англ.; Предисл. В.С.Штаркмана. – Вып.2. – М.: Финансы и статистика, 1986. – 269с.
3. Семёнов Ю.А. Протоколы и ресурсы Internet. – М.: Радио и связь, 1996. – 320с.
4. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы. – СПб: Издательство «Питер», 2000. – 672с.
5. Зима В.М., Молдовян А.А., Молдовян Н.А. Компьютерные сети и защита передаваемой информации. – СПб.: СпбГУ, 1998. – 328с.
6. Кульгин М.В. Коммутация и маршрутизация IP/IPX–трафика. – М.: Компьютер-Пресс, 1998. – 320с.

ОГЛАВЛЕНИЕ

СЕТЕВЫЕ ТЕХНОЛОГИИ В РАДИОЭЛЕКТРОННЫХ СИСТЕМАХ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к выполнению лабораторных работ №1-4
для студентов специальности 11.05.01
«Радиоэлектронные системы и комплексы»
очной формы обучения

Составитель
Сукачев Александр Игоревич

Издается в авторской редакции

Подписано к изданию 19.03.2024.
Уч.-изд. л. 2,2.

ФГБОУ ВО «Воронежский государственный технический университет»
394006 Воронеж, ул. 20-летия Октября, 84