

ГОСУДАРСТВЕННЫЙ КОМИТЕТ СВЯЗИ, ИНФОРМАТИЗАЦИИ И
ТЕЛЕКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ РЕСПУБЛИКИ
УЗБЕКИСТАН
ТАШКЕНТСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

К защите.
Зав. кафедрой

2013 г.

Выпускная квалификационная работа бакалавра

на тему «Разработка способа регулирования охлаждения компьютерных систем»

Выпускник	_____	<u>Туляганов Ж.А.</u>
	(подпись)	(Фамилия)
Руководитель	_____	<u>Ташпулатов Х.З.</u>
	(подпись)	(Фамилия)
Рецензент	_____	<u>Миршахаджаев Б.М.</u>
	(подпись)	(Фамилия)
Консультант по БЖД	_____	<u>Кодиров Ф.М.</u>
	(подпись)	(Фамилия)

Ташкент-2013

ГОСУДАРСТВЕННЫЙ КОМИТЕТ СВЯЗИ, ИНФОРМАТИЗАЦИИ И
ТЕЛЕКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ РЕСПУБЛИКИ
УЗБЕКИСТАН
ТАШКЕНТСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Факультет: Информационные технологии Кафедра: Компьютерные системы
Направления (специальность): 5811300 - Сервис (электроники и
компьютерной техники)

У Т В Е Р Ж Д А Ю

Зав кафедрой _____
« ____ » _____ 2013 г.

ЗАДАНИЕ

на выпускную квалификационную работу Туляганова Жамолиддина
Абдусаматовича
(фамилия, имя, отчество)

1. Тема работы: Разработка способа регулировка охлаждения компьютерных систем.
2. Утверждена приказом по университету от «11»февраль 2013 г. № 145-07.
3. Срок сдачи законченной работы: 30.05.13 г.
4. Исходные данные к работе: учебные пособие, информационно-справочные материалы фирм изготовителей вентиляторов, выбор программы для тестирования температурного режима.
5. Содержание расчётно-пояснительной записи (перечень подлежащих разработке вопросов): Введение. Обзор существующих типов и видов охлаждения. Воздушное охлаждение компьютерных систем. Пассивное охлаждение. Водяное охлаждение компьютерных систем. Разработка способа регулировка охлаждения компьютерных систем. Тепловая защита процессора. Минимизация потребления энергии. Утилита RMClock. Автаразгон видеокарты. Перспективы развития систем охлаждения. Заключение. Литература.
6. Перечень графического материала: рисунок видов охлаждения, рисунок вентилятора, диаграмма анализа стоимости различных видов охлаждения, рисунок программы Throttle Watch, рисунок утилиты RMClock, диаграмма эффективности систем охлаждения ПК.
7. Дата выдачи задания: 26.03.13.

Руководитель _____ Задание принял _____
(подпись) (подпись)

8. Консультант по отдельным разделам выпускной работы

Раздел	Ф.И.О Руководителя	Подпись дата	
		Задание выдал	Задание получил
Основная часть	Ташпулатов Х.З.	26.03.13	30.05.13
Безопасность жизнедеятельности	Кддиров Ф.М.	20.05.13	31.05.13

9. График выполнения работы

№	Наименование раздела работы	Срок выполнения	Отметка руководителя выполнении
1.	Обзор существующих типов и видов охлаждения.	26.03.13-13.04.13	
2.	Охлаждения компьютерных систем.	15.04.13-30.04.13	
3.	Разработка способа регулирования охлаждения компьютерных систем; Заключение	11.05.13-31.05.13	
4.	Безопасность жизнедеятельности	22.05.13-31.05.13	
5.	Предварительная защита ВКР	21.05.13	

Выпускник

« »

2013 г.

(подпись)

« »

2013г.

Руководитель

(подпись)

Битирув малакавий ишда компьютер тизимини совутиш, турли хдл совутиш тизимларида хдво окимлари хдракати, актив ва пасив совутиш тизимини ишлатиш киёсий характеристикаси назарий саволлари куриб чикилди, компьютер тизимларида хдрорат режимини тадқиқ қилиш ва тестлаш учун Idle ва 3Dmark Vantage дастурлари ишлатилди. Компьютер тизими хдроратини тестлаш натижаларига кура куш и мча вентиляторлар ва термореле урнатиш тақлиф этилди.

БМИ кириш, 3 та булим ва хдёт фаолияти хдвфсизлиги, хулоса, 65 варок матндан, 23 та расмдан ва фойдаланилган адабиётлар руйхатидан иборат. В данной выпускной квалификационной работе рассмотрены теоретические вопросы охлаждения компьютерных систем, движение воздушных потоков при различных системах охлаждения, сравнительная характеристика использования активной и пассивной системы охлаждения, использованы программы Idle и 3Dmark Vantage для исследования и тестирования температурного режима в компьютерных системах. По результатам тестирования температуры компьютерных систем предложена установить дополнительные вентиляторы и термореле. ВКР состоит из введения, 3 разделов и БЖД, заключение, 65 страниц текста, 23 рисунков и списка использованной литературы. In this final qualification work considered theoretical issues of computer cooling systems, air traffic flows at different cooling systems, comparative analysis of the use of active and passive cooling system, used program Idle and 3Dmark Vantage for research and testing temperature in the computer systems. Based on the results of testing the temperature of computer systems proposed to install additional fans and thermostats. Final qualification work consists from introduction, 3 sections and health and safety, the inference, the 65 pages of the text, the 23 figures and the list of the used literature.

Содержание

Введение.....	7
1. ОБЗОР СУЩЕСТВУЮЩИХ ТИПОВ И ВИДОВ ОХЛАЖДЕНИЯ	
1.1 .Принципы охлаждения (типы и виды).....	9
1.2.Охлаждение процессоров и видеокарт	18
1.3.Охлаждение жесткого диска	19
1.4.Охлаждение системного блока	21
1.5.Анализ различных видов охлаждения	23
2. ОХЛАЖДЕНИЯ КОМПЬЮТЕРНЫХ СИСТЕМ	
2.1. Воздушное охлаждение компьютерных систем.....	26
2.2. Пассивное охлаждение	34
2.3. Водяное охлаждение компьютерных систем.....	35
3. РАЗРАБОТКА СПОСОБА РЕГУЛИРОВКА ОХЛАЖДЕНИЯ КОМПЬЮТЕРНЫХ СИСТЕМ	
3.1. Тепловая защита процессора	38
3.2. Минимизация потребления энергии.....	40
3.3. Утилита RMClock.....	42
3.4. Авторазгон видеокарты.....	44
3.5.Перспективы развития систем охлаждения.....	47
4. БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ	
4.1 .Защита человека от поражения электрическим током.....	51
4.2. Ликвидация последствий чрезвычайных ситуаций.....	55
Заключение	62
Список литературы	63
Приложение	65

Введение

В Постановлении Президента Республики Узбекистан "О мерах по дальнейшему внедрению и развитию современных информационно-коммуникационных технологий. (Собрание законодательства Республики Узбекистан, 2012 г., № 13, ст. 139) одной из основных задачи дальнейшего внедрения и развития информационно-коммуникационных технологий, в частности, является программа мер по коренному и качественному улучшению функционирования национальной информационно-поисковой системы, увеличению количества ее пользователей [1].

Компьютеры уже прочно вошли в современный мир, во все сферы человеческой деятельности и науки, тем самым, создавая необходимость в обеспечении их различным программным обеспечением. Конечно, в первую очередь это связано с развитием электронной вычислительной техники и с её быстрым совершенствованием и внедрением в различные сферы человеческой деятельности.

Объединение компьютеров в сети позволило значительно повысить производительность труда. Компьютерные сети используются как для производственных (или офисных) нужд, так и для обучения, общения и т.д.

Практически все операционные системы, занимающие заметное место на рынке, стали сетевыми. Сетевые функции сегодня встраиваются в ядро ОС, являясь ее неотъемлемой частью. Операционные системы получили средства для работы со всеми основными технологиями локальных (Ethernet, Fast Ethernet, Gigabit Ethernet, Token Ring, FDDI, ATM) и глобальных (X.25, frame relay, ISDN, ATM) сетей, а также средства для создания составных сетей (IP, IPX, AppleTalk, RIP, OSPF, NLSP).

Целью данной работы является создание программного продукта, который предназначен для создания небольшой сети с оптимизацией ее работы на платформе операционной системы Linux, обеспечивая полное управление сетью в качестве администрирования. И описание идей и

принципов работы данной программной системы.

Linux - общее название Unix-подобных операционных систем на основе одноимённого ядра и собранных для него библиотек и системных программ, разработанных в рамках проекта GNU. GNU - другая Unix - подобная ОС, разрабатываемая Ричардом Столлманом, ПО для которой и было во многом заимствовано Linux. Само же ядро Linux было разработано программистом и хакером Линусом Торвальдсом, поэтому правильнее говорить GNU/Linux - то есть совокупность ядра и ПО.

По данным Linux Foundation, отчасти этот рост связан с той ролью, которую играет Linux в решении крупнейших на сегодня задач ИТ (информационные технологии) - поддержки растущего объема сверхбольших данных, повышении производительности и безопасности при использовании виртуализации и облачных вычислений. Корпоративные пользователи Linux демонстрируют непрерывный прогресс на всех этих направлениях и отдают явное предпочтение Linux при решении названных задач.

Операционные системы семейства Linux - наиболее динамично развивающееся семейство ОС. Незаконченность установления полной упорядоченности ресурсов, некоторая сложность в освоении обычными пользователями и неполное признание производителями ПО - вот основные недостатки Linux. Преимуществ гораздо больше, это: массовость - то есть общественная доступность проекта, участие в нём программистов со всего мира. Благодаря массовости быстрее выявляются изъяны, создаются новые дистрибутивы; открытость ОС Linux сделала её потрясающей сферой для работы программиста; бесплатность ОС говорит сама за себя; развитая оболочка делает эту систему более профессиональной; наконец система безопасности в этой ОС продумана с особой тщательностью. Linux - это особая идеология, которая возможно вскоре покорит мир, придя на смену гегемонии Windows.

1. ОБЗОР И СИСТЕМОТЕХНИЧЕСКИЙ АНАЛИЗ

1.1. Основные задачи оптимизации сетей

В данной выпускной квалификационной работе предлагается разработать модуль программной системы оптимизации работы сети, ориентированной на использование в небольших локальных сетях и в университетских лабораториях компьютерных сетей для проведения практических занятий. Система должна решать следующий круг задач:

- управление пользователями и группами пользователей (добавление, удаление, изменение параметров учетных записей, назначение пользователям маски режима создания файлов). Управление файлами и каталогами (изменение прав доступа, изменение владельца и группы). Обеспечение удаленного доступа к компьютерам сети;

- установка, конфигурирование и запуск в сети служб DHCP, DNS, FTP, Proxu и электронной почты;

- установка, конфигурирование и запуск службы, реализующей протокол маршрутизации;

- предоставление администратору сети сведений о функционировании служб, установленных и запущенных с помощью системы.

Анализ литературных источников по теме проектирования был проведен с целью определения потребности в декомпозиции основной задачи на составляющие. В частности, были выделены следующие базовые подзадачи:

- а) реализация интерактивного взаимодействия с пользователем;
- б) обработка данных, введенных пользователем, написание скриптов, через которые программа будет взаимодействовать с операционной системой и сетью;
- г) анализ данных и выдача результатов пользователю.

Так как все сложные системы разрабатываются группами разработчиков, то было принято решение сформировать группу разработчиков из 3-х человек в соответствии с приведенной иерархией задач. Следует уточнить, что в виду своего широкого круга задач система может быть разделена на составляющие. При этом

каждая составляющая будет независима друг от друга, так как указанные типы задач для системы не требуют взаимодействия и обмена промежуточными результатами. В итоге, система в целом разрабатывается группой из трех человек, ориентированных на разработку следующих подсистем:

- модуль протоколов управления;
- модуль протоколов высокого уровня;
- модуль маршрутизации.

Каждый модуль будет иметь графический интерфейс.

В рамках разработки модуля протоколов управления следует спроектировать и реализовать:

- предлагаемую функциональность системы для конечных пользователей на основе указанного выше круга задач.

- механизмы взаимодействий программы с операционной системой и локальной сетью. В том числе: методы передачи управления от интерфейсного модуля к терминалу операционной системы Linux, через который непосредственно выполнение системных команд с параметрами и управление конфигурационными файлами, методы скриптов проверки установки и запуска служб, возвращения данных после их выполнения в интерфейсную часть.

Программное обеспечение должно быть протестировано.

Системотехнический анализ программного модуля следует проводить с точки зрения обеспечения оптимизации в работе сети по протоколам управления. При выборе вариантов реализации программного модуля основными критериями должны быть быстродействие, используемые ресурсы и стоимость.

Следует учитывать сроки разработки, которые рассчитываются на основании задания к разработке и технико-экономического обоснования. А также разработать мероприятия по охране труда, а именно, произвести анализ условий труда и расчет естественного освещения. Разработать мероприятия по гражданской обороне, а именно, произвести оценку радиационной обстановки, определить мероприятия по защите работников лаборатории в зоне радиоактивного загрязнения при аварии на АЭС.

Для того чтобы оптимизировать сеть, то есть сделать ее работу более эффективной, необходимо решить следующие задачи [10]:

1) Сформулировать критерии эффективности работы сети. Чаще всего такими критериями служат производительность и надежность, для которых в свою очередь требуется выбрать конкретные показатели оценки, например, время реакции и коэффициент готовности, соответственно.

2) Определить множество варьируемых параметров сети, прямо или косвенно влияющих на критерии эффективности. Эти параметры действительно должны быть варьируемыми, то есть нужно убедиться в том, что их можно изменять в некоторых пределах по вашему желанию. Так, если размер пакета какого-либо протокола в конкретной операционной системе устанавливается автоматически и не может быть изменен путем настройки, то этот параметр в данном случае не является варьируемым, хотя в другой операционной системе он может относиться к изменяемым по желанию администратора, а значит и варьируемым. Другим примером может служить пропускная способность внутренней шины маршрутизатора - она может рассматриваться как параметр оптимизации только в том случае, если вы допускаете возможность замены маршрутизаторов в сети.

Все варьируемые параметры могут быть сгруппированы различным образом. Например, параметры отдельных конкретных протоколов (максимальный размер кадра протокола Ethernet или размер окна неподтвержденных пакетов протокола TCP) или параметры устройств (размер адресной таблицы или скорость фильтрации моста, пропускная

способность внутренней шины маршрутизатора). Параметрами настройки могут быть и устройства, и протоколы в целом. Так, например, улучшить работу сети с медленными и зашумленными глобальными каналами связи можно, перейдя со стека протоколов IPX/SPX на протоколы TCP/IP. Также можно добиться значительных улучшений с помощью замены сетевых адаптеров неизвестного производителя на адаптеры BrandName.

3) Определить порог чувствительности для значений критерия эффективности. Так, производительность сети можно оценивать логическими значениями "Работает"/ "Не работает", и тогда оптимизация сводится к диагностике неисправностей и приведению сети в любое работоспособное состояние. Другим крайним случаем является тонкая настройка сети, при которой параметры работающей сети (например, размер кадра или величина окна неподтвержденных пакетов) могут варьироваться с целью повышения производительности (например, среднего значения времени реакции) хотя бы на несколько процентов. Как правило, под оптимизацией сети понимают некоторый промежуточный вариант, при котором требуется выбрать такие значения параметров сети, чтобы показатели ее эффективности существенно улучшились, например, пользователи получали ответы на свои запросы к серверу баз данных не за 10 секунд, а за 3 секунды, а передача файла на удаленный компьютер выполнялась не за 2 минуты, а за 30 секунд.

Таким образом, можно предложить три различных трактовки задачи оптимизации:

1) Приведение сети в любое работоспособное состояние. Обычно эта задача решается первой, и включает:

- поиск неисправных элементов сети - кабелей, разъемов, адаптеров, компьютеров;
- проверку совместимости оборудования и программного обеспечения;
- выбор корректных значений ключевых параметров программ и устройств, обеспечивающих прохождение сообщений между всеми узлами сети - адресов сетей и узлов, используемых протоколов, типов кадров

Ethernet и т.п.

2) Грубая настройка - выбор параметров, резко влияющих на характеристики (надежность, производительность) сети. Если сеть работоспособна, но обмен данными происходит очень медленно (время ожидания составляет десятки секунд или минуты) или же сеанс связи часто разрывается без видимых причин, то работоспособной такую сеть можно назвать только условно, и она безусловно нуждается в грубой настройке. На этом этапе необходимо найти ключевые причины существенных задержек прохождения пакетов в сети. Обычно причина серьезного замедления или неустойчивой работы сети кроется в одном неверно работающем элементе или некорректно установленном параметре, но из-за большого количества возможных виновников поиск может потребовать длительного наблюдения за работой сети и громоздкого перебора вариантов. Грубая настройка во многом похожа на приведение сети в работоспособное состояние. Здесь также обычно задается некоторое пороговое значение показателя эффективности и требуется найти такой вариант сети, у которого это значение было бы не хуже порогового. Например, нужно настроить сеть так, чтобы время реакции сервера на запрос пользователя не превышало 5 секунд.

3) Тонкая настройка параметров сети (собственно оптимизация). Если сеть работает удовлетворительно, то дальнейшее повышение ее производительности или надежности вряд ли можно достичь изменением только какого-либо одного параметра, как это было в случае полностью неработоспособной сети или же в случае ее грубой настройки. В случае нормально работающей сети дальнейшее повышение ее качества обычно требует нахождения некоторого удачного сочетания значений большого количества параметров, поэтому этот процесс и получил название "тонкой настройки".

Даже при тонкой настройке сети оптимальное сочетание ее параметров (в строгом математическом понимании термина "оптимальность") получить невозможно, да и не нужно. Нет смысла затрачивать колоссальные усилия по

нахождению строгого оптимума, отличающегося от близких к нему режимов работы на величины такого же порядка, что и точность измерений трафика в сети.

Достаточно найти любое из близких к оптимальному решений, чтобы считать задачу оптимизации сети решенной. Такие близкие к оптимальному решения обычно называют рациональными вариантами, и именно их поиск интересует на практике администратора сети или сетевого интегратора.

1.2. Оптимизация работы сети на основе протоколов управления в ОС Linux

Эффективная профессиональная работа в Linux немыслима без использования командной строки. Пользователям, привыкшим работать в системах с графическим интерфейсом, работа с командной строкой может показаться неудобной: то, что можно сделать одним перетаскиванием мышью в командной строке потребует ввода с клавиатуры нескольких слов: команды с аргументами. Однако в Linux этот вид интерфейса всегда был основным, а поэтому и хорошо развитым. В командных оболочках, используемых в Linux, есть масса способов экономии усилий (нажатий на клавиши) при выполнении наиболее распространённых действий: автоматическое дополнение длинных названий команд или имён файлов, поиск и повторное выполнение команды, уже когда-то исполнявшейся раньше, подстановка списков имён файлов по некоторому шаблону и многое другое. Преимущества командной строки становятся особенно очевидны, когда требуется выполнять однотипные операции над множеством объектов. В системе с графическим интерфейсом потребуется столько перетаскиваний мышью, сколько есть объектов, в командной строке будет достаточно одной (пусть длинной и сложной) команды [5].

В первую очередь рассмотрим инструменты, предоставляющие возможность работы с командной строкой в дистрибутиве Linux (командные оболочки и эмуляторы терминала), а затем основные утилиты командной

строки, выполняющие важнейшие операции с файловой системой и данными.

Командная оболочка (или интерпретатор команд) - это программа, задача которой состоит в том, чтобы передавать ваши команды операционной системе и

прикладным программам, а их ответы — вам. По своим задачам ему соответствует `command.com` в MS-DOS или `cmd.exe` в Windows, но функционально оболочки в Linux несравненно богаче. На языке командной оболочки можно писать небольшие программы для выполнения ряда последовательных операций с файлами и содержащимися в них данными - сценарии (скрипты).

В дистрибутиве Linux доступны следующие командные оболочки:

- `bash` - самая распространённая оболочка под Linux. Она умеет дополнять имена команд и файлов, ведёт историю команд и предоставляет возможность их редактирования;
- `pdkdh` - клон `korn shell`, хорошо известной в системах UNIX командной оболочки;
- `sash` - особенность этой оболочки состоит в том, что она не зависит ни от каких разделяемых библиотек и включает в себя упрощённые реализации некоторых важнейших утилит, таких как `ls`, `dd` и `gzip`. Поэтому `sash` особенно полезна при восстановлении после некоторых системных сбоев или при обновлении версии важнейших разделяемых библиотек;
- `tcsh` - улучшенная версия `C shell`;
- `zsh` - новейшая из перечисленных здесь оболочек, в ней реализованы расширенные возможности автодополнения аргументов команд и множество других функций, делающих работу с оболочкой ещё более удобной и эффективной. Однако все расширения `zsh` по умолчанию отключены, поэтому прежде чем начать пользоваться этой командной оболочкой, необходимо прочесть документацию к ней и включить те функции, которые необходимы.

Оболочкой по умолчанию является `bash` (Bourne Again Shell). Чтобы проверить, какая оболочка используется, необходимо набрать команду: `echo`

14

`$ SHELL`.

Терминал в Linux - это программа, которая предоставляет пользователю возможность вести диалог с системой при помощи интерфейса командной строки. Терминалы позволяют передавать системе и получать от неё только текстовые

данные. Стандартный терминал к системе Linux можно получить на любой текстовой виртуальной консоли, а для того, чтобы получить доступ к командной строке из графической оболочки, требуются специальные программы - эмуляторы терминала.

Ниже перечислены некоторые из эмуляторов терминала вошедших в дистрибутив Linux:

- `xterm` - стандартный эмулятор терминала для X Window System. Этот эмулятор совместим с терминалами DEC VT102/VT220 и Tektronix 4014 и предназначен для программ, не использующих графическую среду напрямую. Если операционная система поддерживает изменение окна терминала (например, сигнал SIGWINCH в системах, произошедших от 4.3bsd), `xterm` может использовать сообщать программам, запущенным в нем, что размер окна изменился;

- `aterm` - это цветной эмулятор терминала `gxt` версии 2.4.8, дополненный полосами прокрутки в стиле NeXT от Альфредо Кодзима (Alfredo Kojima). Он предназначен для замены `xterm` в случае, если не нужна эмуляция терминала Tektronix 4014.

Оптимизация работы сети на платформе операционной системы Linux по протоколам управления выполняется на четырех этапах или уровнях [4]:

- 1) Управление пользователями. Подключение новых пользователей и удаление старых рутинная задача в большинстве систем. Выполнение этих задач не представляет особой сложности, но достаточно трудоемкая. Большинство администраторов создает различные средства автоматизации этого процесса, а затем поручают выполнение реальной работы своему заместителю или оператору.

Файл `/ETC/PASSWD`. Файл `/etc/passwd` содержит список пользователей,

15

которые известны системе. В процессе регистрации пользователя система обращается к этому файлу в поисках идентификатора пользователя и его домашнего каталога. Каждая строка файла описывает одного пользователя и содержит семь полей, разделенных двоеточиями:

- регистрационное имя;

- зашифрованный пароль или “заполнитель” пароля;
- идентификатор пользователя;
- идентификатор группы по умолчанию;
- поле GECOS (полное имя, номер офиса, рабочий и домашний телефоны);
- домашний каталог;
- регистрационная оболочка.

В наши дни совершенно недопустимо хранить зашифрованные пароли в текстовом виде. Современные быстрые компьютеры позволяют взламывать такие пароли за считанные минуты. Все версии UNIX позволяют спрятать их, поместив в отдельный файл, недоступный для чтения. Это называется механизмом скрытых паролей и (по вполне понятным причинам) применяется в большинстве дистрибутивов. Механизм скрытых паролей становится понятнее, когда его описывают в качестве расширения к традиционному файлу `/etc/passwd`.

Ниже рассматривается назначение отдельных полей файла `/etc/passwd`.

Регистрационное имя. Регистрационные имена (называемые также пользовательскими именами) должны быть уникальными и состоять не более чем из 32 символов. Они могут содержать любые символы, кроме двоеточия и символа новой строки. Если используется база данных, длина имени ограничена 8 символами независимо от операционной системы.

В некоторых старых версиях допустимый диапазон ограничивался алфавитно цифровыми символами, а длина имени не должна была превышать 8 символов. В смешанных системах рекомендуется придерживаться наиболее строгих ограничений. Это дает возможность избежать конфликтов со

16

старыми программами, а также использовать одно и то же регистрационное имя на любом компьютере.

Регистрационные имена могут включать как строчные, так и прописные буквы. Смешение регистров в именах не приводит к возникновению проблем, но имена, состоящие из строчных букв, считаются традиционными, к тому же их проще вводить. Следует выбирать такие регистрационные имена, которые легко запомнить, поэтому имя, состоящее из случайной последовательности букв,

является не самым удачным вариантом.

Поскольку регистрационные имена часто используются в адресах электронной почты, полезно определить стандартную процедуру их формирования. Пользователям должна быть предоставлена возможность делать обоснованные предположения о регистрационных именах друг друга. Имена, фамилии, инициалы и сочетания этих элементов - вот приемлемые варианты для схем формирования имен.

Любая жестко заданная схема в конечном счете приводит к появлению дубликатов или слишком длинных имен, поэтому иногда придется делать исключения. В случае появления длинного имени можно использовать предоставляемую почтовой системой функцию определения псевдонимов, чтобы задать две версии одного и того же имени, по крайней мере, для электронной почты.

Например, схема выбора имен может быть такой: первый инициал и фамилия каждого сотрудника. Пользователь Брент Браунинг (Brent Browning), таким образом, превратится в "bbrowning", но девять символов это слишком много для некоторых систем. Лучше присвоить пользователю регистрационное имя "brentb", а "bbrowning" сделать элементом файла aliases: bbrowning: brentb.

Если в организации имеется глобальный файл почтовых псевдонимов, то все новые регистрационные имена должны отличаться от псевдонимов, присутствующих в этом файле. В противном случае почта будет

17

доставляться не новому пользователю, а пользователю с соответствующим псевдонимом.

В крупных организациях в адресах электронной почты часто используются полные имена (например, John.Q.Public@mysite.com), благодаря чему регистрационные имена скрываются от внешнего мира. Это прекрасная идея, но, чтобы облегчить работу администраторам, необходимо установить четкое и понятное соответствие между регистрационными именами и реальными именами пользователей.

Регистрационные имена должны быть уникальными в двух отношениях.

Во-первых, необходимо, чтобы имя пользователя на всех компьютерах было одинаковым. Это удобно как самому пользователю, так и администратору. Во-вторых, регистрационное имя всегда должно относиться к одному и тому же лицу. Некоторые команды (например, `ssh`) можно настроить так, чтобы они аутентифицировали удаленных пользователей по регистрационным именам. Даже если адреса `scott@boulder` и `scott@refuge` относятся к разным пользователям, то в случае неправильной настройки учетных записей оба пользователя могут при определенных обстоятельствах получить доступ к файлам друг друга, не вводя пароль.

Зашифрованный пароль. Прежде чем переходить к деталям, напомним: в большинстве систем зашифрованные пароли хранятся в файле `/etc/shadow`, а не `/etc/passwd`.

Пароли хранятся в зашифрованном виде. Необходимо либо установить содержимое этого поля с помощью команды `passwd` либо скопировать строку, содержащую зашифрованный пароль, из другой учетной записи.

Вручную редактируя файл `/etc/passwd` для создания новой учетной записи, необходимо поставить звездочку (*) в поле зашифрованного пароля. Звездочка воспрепятствует несанкционированному использованию учетной записи до установки реального пароля. Нельзя оставлять это поле пустым, иначе безопасность системы подвергнется серьезному нарушению, поскольку для доступа к такой учетной записи пароль не требуется.

18

Основные дистрибутивы Linux поддерживают несколько методов шифрования паролей, и они могут определять какой из них был использован для шифрования каждого из паролей. Таким образом, не обязательно, чтобы все пароли были зашифрованы по одному алгоритму.

В большинстве дистрибутивов Linux по умолчанию используется алгоритм MD5. С криптографической точки зрения алгоритм MD5 лишь не многим более устойчив, чем DES, зато в нем допускаются пароли произвольной длины. Чем длиннее пароли, тем они надежнее. Поскольку алгоритм MD5 весьма универсален и безопасен, рекомендуется применять его во всех системах, которые его

поддерживают.

Зашифрованные пароли имеют постоянную длину (34 символа в случае MD5 и 13 символов в случае DES) независимо от длины исходного пароля. Пароли шифруются с добавлением случайной “примеси”, чтобы одному исходному паролю соответствовало много зашифрованных форм. Таким образом, факт выбора пользователями одинаковых паролей не может быть выявлен путем просмотра зашифрованных паролей. Пароли MD5 легко распознать, так как они всегда начинаются с последовательности “\$1\$”.

2. Идентификатор пользователя. Идентификатор пользователя (UID) - это 32-битное целое число. Но для обеспечения совместимости со старыми системами мы рекомендуем, чтобы значение самого старшего идентификатора по возможности не превышало 32767 (наибольшее 16- битное целое число со знаком).

По определению пользователь root имеет идентификатор 0. В большинстве систем есть также псевдопользователи bin, daemon и ряд других. Как правило, такие псевдоимена помещаются в начало файла /etc/passwd, и им назначаются маленькие идентификаторы. Чтобы зарезервировать побольше номеров для неперсонифицированных пользователей, рекомендуется присваивать реальным пользователям идентификаторы, начиная с номера 100 (или выше).

Нежелательно создавать, более одной учетной записи с

19

идентификатором 0. Может показаться удобным иметь несколько суперпользовательских записей с разными интерпретаторами команд и паролями, но в действительности это лишь создает дополнительные бреши в системе защиты и приводит к ненужным трудностям.

Необходимо избегать повторного использования идентификаторов, даже идентификаторов тех пользователей, которые уволились из организации и учетные записи которых были удалены. Такая мера предосторожности позволит избежать путаницы, если файлы впоследствии будут восстанавливаться из резервных копий, где пользователи могут быть идентифицированы по номерам, а не по регистрационным именам.

Идентификаторы должны быть уникальными в пределах всей организации, т.е. конкретный идентификатор должен соответствовать одному и тому же регистрационному имени и физическому лицу на каждом компьютере. Если уникальность идентификаторов нарушена, то в такой системе, как NFS, появятся проблемы безопасности; кроме того, это может привести в замешательство пользователей, переходящих из одной рабочей группы в другую.

Трудно соблюдать уникальность идентификаторов, когда группы компьютеров находятся в административном подчинении разных лиц и даже организаций. Это проблема как техническая, так и концептуальная. Лучшим ее решением является создание центральной базы данных, содержащей для каждого пользователя уникальную запись. Проще всего назначить каждой группе в пределах организации отдельный диапазон идентификаторов и позволить распоряжаться им по своему усмотрению. В результате решится вопрос с уникальностью пользовательских идентификаторов (необходимое условие в случае совместного использования файловых систем посредством NFS), но останется проблема уникальности регистрационных имен.

Идентификатор группы по умолчанию. Как и идентификатор пользователя, идентификатор группы (GID) является 32-битным целым числом. Идентификатор 0 зарезервирован для группы с именем `root`, 20 идентификатор 1 - для группы `bin`, а идентификатор 2 для группы `daemon`.

Группы определяются в файле `/etc/group`, а поле идентификатора группы в файле `/etc/passwd` задает стандартный (“эффективный”) идентификатор на момент регистрации пользователя в системе. Этот идентификатор не играет особой роли при определении прав доступа; он используется лишь при создании новых файлов и каталогов. Новые файлы обычно включаются в эффективную группу своего владельца, но если у каталога установлен бит SGID (2000) или файловая система смонтирована с опцией `gripd`, новые файлы включаются в группу своего каталога.

Поле GECOS. Поле GECOS в основном используется для хранения персональной информации о каждом пользователе. Оно не имеет четко определенного синтаксиса. Первоначально в BellLabs его использовали для хранения регистрационной информации, необходимой при передаче пакетных

заданий из UNIX-систем мэйнфрейму, работающему под управлением GECOS (General Electric Comprehensive OperatingSystem - комплексная операционная система компании General Electric).

В принципе структура поля GECOS может быть произвольной, но команда `finger` интерпретирует разделенные запятыми элементы поля в следующем порядке:

- полное имя (часто используется только это поле);
- номер офиса и здания;
- рабочий телефон;
- домашний телефон.

С помощью команды `chfn` можно изменять информацию, содержащуюся в поле GECOS. Эта команда используется для ведения и обновления списка телефонных номеров, но ею часто злоупотребляют: пользователь может изменить информацию так, что она станет нецензурной или некорректной.

3. Домашний каталог. Войдя в систему, пользователь попадает в свой домашний каталог. Если на момент регистрации этот каталог

21

отсутствует, выводится сообщение вроде “no home directory” (домашний каталог отсутствует). Если в качестве значения поля домашнего каталога по умолчанию `DEFAULTHOME` в файле `/etc/login.defs` установлено `no`, продолжение регистрации пользователя будет невозможно в противном случае пользователь попадает в корневой каталог.

Если домашние каталоги смонтированы через NFS, то в случае проблем с сервером или с самой сетью они могут оказаться недоступными.

4. Регистрационная оболочка. В качестве регистрационной оболочки, как правило, задается интерпретатор команд, например `/bin!sh` или `/bin/csh`, но в принципе это может быть любая программа. По умолчанию используется интерпретатор `bash`, который запускается, если соответствующее поле в файле `/etc/passwd` не задано. В Linux имена `csh` и `sh` являются всего лишь ссылками на `tcsh` (надмножество команд `csh`) и `bash` (GNU-версия `sh`), соответственно. Во многих дистрибутивах есть также открытая версия интерпретатора `ksh`.

Пользователи могут менять интерпретатор с помощью команды `chsh`. Файл `/etc/shells` содержит список тех интерпретаторов, которые доступны для выбора. В SuSE этот список является обязательным для соблюдения, а в RedHat будет лишь выдано предупреждение в случае выбора незарегистрированного интерпретатора.

Файл `/ETC/SHADOW`. Файл `/etc/shadow` доступен для чтения только суперпользователю и предназначен для хранения зашифрованных паролей подальше от любопытных глаз. В нем также содержится учетная информация, которая отсутствует в файле `/etc/passwd`. В одних дистрибутивах механизм скрытых паролей используется по умолчанию, в других - конфигурируется в виде дополнительного пакета. При использовании скрытых паролей соответствующие поля в файле `/etc/passwd` всегда содержат символ `x`.

Файл `shadow` не включает в себя файл `passwd`, и последний не генерируется автоматически при изменении файла `shadow`. Оба файла

22

необходимо сопровождать независимо друг от друга (или использовать команду наподобие `useradd` для автоматического управления файлами). Как и `/etc/passwd`, файл `/etc/shadow` содержит одну строку для каждого пользователя. Каждая строка состоит из 9 полей, разделенных двоеточиями;

- регистрационное имя;
- зашифрованный пароль;
- дата последнего изменения пароля;
- минимальное число дней между изменениями пароля;
- максимальное число дней между изменениями пароля;
- количество дней до истечения срока действия пароля, когда выдается

предупреждение;

- количество дней по истечении срока действия пароля, когда учетная запись аннулируется;

- срок действия учетной записи;

- зарезервированное поле, которое в настоящее время всегда пустое. Лишь первые два поля обязательно должны быть заполнены. Поля дат

задаются в виде числа дней (а не секунд), прошедших с 1-го января 1970 г. Это

отличается от стандартного способа вычисления времени в UNIX. Задавать поля можно с помощью команды `usermod`.

Типичная запись выглядит так: `millert:inNO.VRscIWn.: 11508:0:180:14::12417:`

Более подробное описание каждого поля.

- Регистрационное имя берется из файла `/etc/passwd`. Оно связывает записи файлов `passwd` и `shadow`.
- Зашифрованный пароль идентичен тому, который ранее хранился в файле `/etc/passwd`.
- Третье поле содержит дату последнего изменения пользователем своего пароля. Это поле обычно заполняется программой `/usr/bin/passwd`.
- В четвертом поле задано количество дней, спустя которые пользователь сможет снова изменить пароль. Его назначение принудительное

изменение регистрационной информации путем препятствования немедленному возврату пользователей к привычным паролям после их обязательного изменения.

- В пятом поле задано максимальное число дней между двумя изменениями пароля. С помощью этой установки администраторы заставляют пользователей менять свои пароли. Максимальное время жизни пароля определяется суммой значений данного и седьмого полей.

- В шестом поле задано количество дней, оставшихся до момента устаревания пароля, когда программа login должна предупреждать пользователя о необходимости сменить пароль.

- Седьмое поле определяет, спустя какое число дней после устаревания пароля отменяется учетная запись. Его назначение не совсем понятно.

- В восьмом поле задана дата отмены учетной записи. По прошествии этой даты пользователь не сможет зарегистрироваться в системе, пока администратор не сбросит значение поля. Если поле оставлено пустым, учетная запись всегда будет активной.

- Девятое поле зарезервировано на будущее.

Теперь, когда назначение полей стало понятным, вернемся к рассмотренному выше примеру:

```
millert:inNO.VRscIWn.: 11508:0:180:14:: 12417:
```

В этой записи говорится о том, что пользователь millert последний раз менял свой пароль 18 июля 2006 г. Следующий раз пароль должен быть изменен через 180 дней. За две недели до этого пользователь начнет получать предупреждения о необходимости сменить пароль. Учетная запись действительна до 31 декабря 2010г.

С помощью утилиты `rwconv` можно согласовать содержимое файлов `/etc/shadow` и `/etc/passwd`. Большинство параметров файла `shadow` берется из стандартных установок файла `/etc/login.defs`.

Базовая модель управления учетными записями реализуется автоматизированными утилитами с помощью стандартных команд:

1) `adduser` - регистрирует нового пользователя или изменяет информацию по умолчанию о новых пользователях.

Синтаксис:

- `adduser` [options] [`—home` DIR] [`—shell` SHELL] [`—no-create-home`] [`—uid` ID] [`—firstuid` ID] [`—lastuid` ID] [`—ingroup` GROUP | `—gid` ID] [`—disabled-password`] [`—disabled-login`] [`—gecos` GECOS] [`--addextragroups`] user;

- `adduser` `—system` [параметры] [`—home` КАТ] [`—shell` ОБОЛОЧКА] [`—no-create-home`] [`—uid` ID] [`—group|—ingroup` ГРУППА] `—gid` ID] [`—disabled-password`] [`—disabled-login`] [`—gecos` GECOS] пользователь.

Добавление обычного пользователя. При запуске с двумя аргументами и без параметров `—system` или `—group`, `adduser` добавляет обычного пользователя, `adduser` выберет первый доступный UID из диапазона обычных пользователей, заданного в файле настройки. UID может быть изменён с помощью параметра `—uid`. Диапазон, задаваемый файлом настройки, может быть изменён с помощью параметров `—firstuid` и `—lastuid`.

По умолчанию, каждый пользователь в Debian GNU/Linux входит в соответствующую группу с тем же именем. Объединение пользователей в группы позволяет групповую запись в каталог, что легко достигается посредством помещения соответствующих пользователей в новую группу, установкой `set-group-ID` бита на каталог и проверкой того, что все пользователи используют значение `umask` со значением `002`. Если параметру `USERGROUPS` присвоить значение `по`, то все `GID` пользователей будут устанавливаться в `USERS GID`. Первичные группы пользователей также могут быть изменены из командной строки с помощью параметров `—gid` или `- -ingroup`, которые задают группу по `id` или по имени соответственно.

Также, пользователей можно добавить в одну или более групп, указанных в `adduser.conf`, установкой `ADD EXTRA GROUPS` равной `1` в `adduser.conf`, или указав `—add extra groups` в командной строке.

`adduser` создаёт домашний каталог в соответствии со значениями

`DHOME`, `GROUPHOMES` и `LETTERHOMES`. Домашний каталог может быть

изменён из командной строки с помощью параметра `—home`, а оболочка с помощью `—shell`. На домашнем каталоге устанавливается Set-group-ID бит, если `USERGROUPS` равно `yes`, и поэтому любые файлы, создаваемые в домашнем каталоге пользователя, будут иметь правильную группу.

`adduser` копирует файлы из `SKEL` в домашний каталог, запрашивает информацию для `finger` (`gecos`) и пароль. `Gecos` также может быть задан с помощью параметра `—gecos`. При задании параметра `—disabled-login`, учётная запись будет создана, но пока не будет определён пароль, доступ через неё останется заблокированным. При задании параметра `—disabled-password` пароль не нужен, но вход в систему всё равно возможен (например с помощью SSH RSA ключей).

Если файл `/usr/local/sbin/adduser.local` существует, то он будет выполнен после создания учётной записи пользователя, чтобы выполнить все локальные настройки. Аргументы передаваемые `adduser.local`: имя пользователя `uid` `gid` домашний каталог.

Значение переменной среды `VERBOSE` устанавливается по следующему правилу:

- 0, если указан `-quiet`;
- 1, если не указан `—quiet` или `-debug`;
- 2, если указан `-debug`.

Добавление системного пользователя.

При запуске с одним параметром без аргументов и параметром `—system`, `adduser` добавляет системного пользователя. Если пользователь с тем же именем уже существует, то `adduser` завершает работу с выдачей предупреждения.

`adduser` выберет первый доступный `UID` из диапазона системных пользователей, заданного в файле настройки (`FIRST SYSTEM UID` и `LAST SYSTEM UID`). Если необходим какой-то определённый `UID`, то можно указать его с помощью параметра `—uid`.

По умолчанию, системные пользователи помещаются в группу `no group`.

Чтобы поместить нового системного пользователя в уже существующую группу,

используйте параметры `—gid` или `—ingroup`. Чтобы поместить нового системного пользователя в новую группу с тем же ID, используйте параметр `—group`.

Домашний каталог создаётся по тем же правилам что и для обычных пользователей. Новый системный пользователь имеет оболочку `/bin/false` (если не указан параметр `—shell`) и заблокированный пароль. Начальные файлы настроек не копируются.

Параметры:

- `—conf` файл - использовать файл вместо `/etc/adduser.conf`;
 - `—disabled-login` - не запускать `passwd` для задания пароля. Пользователь не сможет использовать свою учётную запись, пока его пароль не установлен;
 - `—disabled-password` - то же что и `—disabled-login`, но вход в систему возможен (например, с помощью ключей SSH RSA), но не использующих парольную аутентификацию;
 - `—force-badname` - по умолчанию, имена пользователя проверяются согласно заданному регулярному выражению `NAME REGEX`, указанному в конфигурационном файле. Этот параметр заставляет `adduser` выполнять только слабую проверку правильности имени;
 - `—gecos GECOS` - задаёт поле `gecos` для новой создаваемой записи, `adduser` не будет запрашивать информацию для `finger`, если указан этот параметр;
 - `—gid ID` - при создании пользователя этот параметр помещает пользователя в группу с этим номером;
 - `—group` - при совместном использовании с `—system`, создаётся группа с тем же именем и ID как и у системного пользователя. Если же `-system` не задана, то создаётся группа с указанным именем;
 - `—help` - показать краткую инструкцию;
 - `—home KAT` - использовать `KAT` в качестве домашнего каталога
- пользователя, а не как задаётся по умолчанию в файле настроек. Если каталог не существует, то он будет создан и будут скопированы начальные файлы настроек;
- `—shell` оболочка - использовать оболочку в качестве регистрационной оболочки, а не как задано по умолчанию в файле настроек;

- `—ingroup ГРУППА` - добавить нового пользователя в ГРУППУ, а не в группу с тем же именем или группу по умолчанию, определяемую `USERSGID` в конфигурационном файле. Это влияет на первичную группу пользователей;
- `—no-create-home` - не создавать домашний каталог, даже если он не существует;
- `—quiet` - не выводить информационные сообщения, показывать только предупреждения и ошибки;
- `—debug` - показывать отладочную информацию, очень полезно, если вы хотите понять в чём проблема у `adduser`;
- `—system` - создать системного пользователя;
- `—uid ID` - принудительно назначить новому `userid` определённый номер, `adduser` завершится неудачно, если такой `userid` уже используется;
- `—firstuid ID` - изменить первый `uid` диапазона, из которого выбран данный `uid` (заменяет значение `FIRST UID`, указанное в конфигурационном файле);
- `—lastuid ID` - изменить последний `uid` диапазона, из которого выбран данный `uid` (`LAST UID`);
- `—add extra groups` - добавляет нового пользователя в дополнительные группы, указанные в конфигурационном файле;
- `—version` - показать номер версии и информацию об авторском праве.

1.3. Системотехнический анализ

При системном анализе объекта проектирования необходимо

рассматривать модуль программной системы оптимизации сети на основе протоколов управления ОС Linux как самостоятельную систему со своими особенностями, при этом необходимо учитывать принципы системного анализа

Принцип конечной цели. Для выполнения проекции данного принципа на проектируемый программный продукт необходимо представить его в виде «черного ящика» (рисунок 1.1). Тогда входными данными, вектор X , будут являться: информация администратора (запрашиваемая при входе в систему), информация о пользователе (запрашиваемая при выполнении процедуры регистрации), команды управления, поступающие от администратора через командную строку. Выходными данными, вектор Y , в этом случае будут являться: предоставление или запрещение доступа к системе и файлам для пользователя, предоставление или запрещение удаленного доступа, информация о состоянии пользователей для администратора. Управляющие параметры, вектор Z - информация о зарегистрированных пользователях, хранящаяся в файле с перечнем пользовательских учётных записей.



Рисунок 1.1 Проектируемая система в виде “черного ящика”

Тогда для выполнения равенства $Y=F(X,Z)$ проектируемая система должна выполнять следующие функции (в совокупности представляющие собой функцию F):

- управление пользователями, т.е. регистрация пользователей в системе, путем внесения личной информации пользователя в файл с перечнем пользовательских учётных записей, а также отключение, удаление учетных записей и изменение пользовательской информации;

- управление группами, т.е. добавление новой группы в систему, путем внесения необходимой информации в файл с перечнем групп, а также удаление и

изменение групп;

- непосредственное предоставление работы удаленного доступа;
- управление ресурсами (изменение прав доступа к файлам, запрет доступа в систему).

Принцип единства. На основании выделенных функций проектируемой системы, можно выделить следующие подсистемы:

- подсистема управления пользователями;
- подсистема управления группами;
- подсистема удаленного доступа;
- подсистема управления ресурсами.

Принцип связности. Совокупность подсистем проектируемой программной системы и их связей - данными, которыми эти подсистемы обмениваются друг с другом и с внешней средой, - образует ее структуру.

Принцип модульности. В проектируемой системе целесообразно выделить следующие модули:

- модуль управления пользователями;
- модуль управления группами;
- модуль управления удаленным доступом;
- модуль управления ресурсами.

Принцип иерархии. При представлении системы целесообразно выделять в ней иерархические уровни:

- уровень системы в целом, как совокупности подсистем;
- уровень подсистем системы, которые состоят из модулей;
- уровень модулей, которые состоят из программ-служб (демонов) и их конфигурационных файлов.

Принцип функциональности. Функции системы в целом рассмотрены в связи с принципом конечной цели. Рассмотрим функции, входные и

выходные данные выделенных подсистем.

Основной функцией подсистемы управления пользователями является добавление нового пользователя в файл с перечнем пользовательских учётных записей, а также отключение, удаление учетных записей и изменение пользовательской информации. Очевидно, что входными данными для данной подсистемы будет являться пользовательская информация (имя, фамилия, группа, адрес электронной почты, и т.д.). Выходные данные - новая учетная запись пользователя, готовая для работы.

Основной функцией подсистемы управления группами является добавление новой группы в файл с перечнем групп, а также удаление и изменение групп. Очевидно, что входными данными для данной подсистемы будет являться информация о группе (имя группы, пароль, идентификатор группы, список членов). Выходные данные - новая группа, добавленная в системные файлы.

Под система управления удаленным доступом должна предоставлять удаленный доступ, начиная от простейшего консольного режима и заканчивая работой системы X Window, от простого редактирования текста до полного администрирования удаленной системы. Входные данные подсистемы - информация о пользователе или администраторе (имя и пароль). Выходные данные подсистемы - удаленный доступ к системе.

Подсистема управления ресурсами должна предоставлять администратору возможность просматривать списки всех зарегистрированных пользователей, запрещать или разрешать доступ к файлам. Входными данными для данной подсистемы являются информация о пользователях, файлы необходимые для доступа и список прав для выбора, а выходными - файлы, предоставленные пользователям с определенными правами.

Принцип развития. Проектируемая система поддерживает добавление новых уровней в протоколы управления.

За время жизни системы она может изменяться, развиваться. В системе

могут быть предусмотрены механизмы адаптации и реконфигурации, позволяющие приспособиться к меняющимся условиям внешней среды.

Возможные пути развития данной системы:

- добавление диалогов работы с новыми командами;
- улучшение интерфейса;
- расширение поддерживаемых протоколов.

При разработке системы следует учитывать возможные пути развития, чтобы в будущем систему было бы не так сложно реконфигурировать, а в лучшем случае просто добавлять соответствующие модули без изменения старых (кроме интерфейса и подсистемы управления).

Принцип сочетания централизации и децентрализации. Во множестве выделенных подсистем можно выделить несколько подмножеств (возможно пересекающихся), которые будут обладать достаточно высокой степенью автономности от других подмножеств. Например, можно выполнить декомпозицию таким образом:

- {подсистема управления пользователями и управления группами};
- {подсистема управления ресурсами};
- {подсистема удаленного доступа}.

Такое разбиение позволит реализовать полученные подмножества в виде отдельных исполняемых модулей и физически разделить процессы управления пользователями, управления ресурсами и работы удаленного доступа.

Принцип учета неопределенности и случайностей. В проектируемой системе следует предусмотреть возможность реакции на некорректные с точки зрения системы действия администратора, например:

-неправильный ввод информации о пользователе при регистрации (отсутствие заполнения какого-либо поля, слишком короткий пароль, неверный формат электронного адреса и т.д.);

-неправильный ввод информации при задании определенных прав на файлы;

- неправильный ввод информации при добавлении или изменении группы;
- неправильный ввод информации при настройке удаленного доступа.

1.4. Вариантный анализ реализации программной системы

Так как протоколы управления в ОС Linux выполняются стандартно, без каких либо вариантов реализации, то вариантному анализу подлежит только четвертый уровень оптимизации, т.е. необходимо решить задачу выбора одного из протоколов, предоставляющий удаленный доступ по сети. Альтернативными являются следующие протоколы: Telnet, SSH, Rlogin.

Для выбора протокола будем использовать метод анализа иерархии (МАИ) [2]. МАИ является систематической процедурой для иерархического представления элементов, определяющих суть любой проблемы. Метод состоит в декомпозиции проблемы на всё более простые составные части и дальнейшей обработки последовательных суждений лица принимающего решение по парным сравнениям. В результате может быть выражена относительная степень (интенсивность) взаимодействия элементов в иерархии. В результате получаются численные выражения этих суждений. МАИ включает в себя процедуры синтеза множественных суждений, получение приоритетных критериев и нахождение альтернативных решений. Полученные знания являются оценками в шкале отношений и соответствуют жёстким оценкам.

Точность вычислений - до четвертого знака после запятой. При расчёте принять отношение несогласованности матриц не более 10%.

Для начала выделим критерии, по которым можно выбирать тот или иной протокол:

- быстродействие (A1);
- стоимость (A2);
- используемые ресурсы (A3);
- возможность аутентификации сервера (A4);
- возможность шифрования паролей и данных (A5).

Определив составляющие части каждого уровня, построим иерархию, описывающую поставленную задачу (Рисунок 1.2).

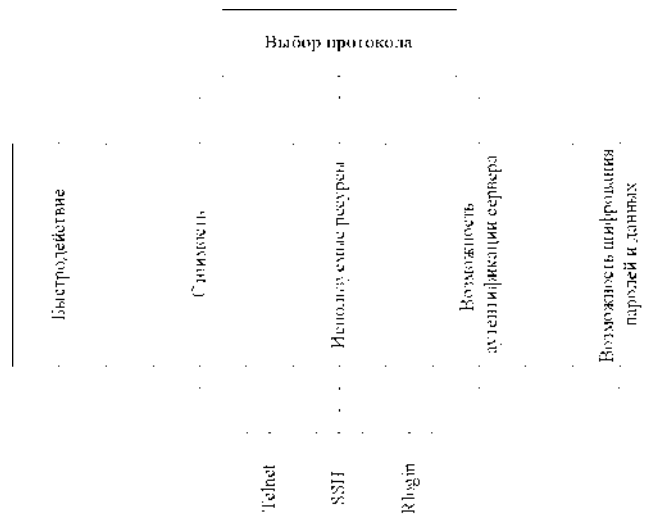


Рисунок 1.2. Иерархия для решаемой задачи

Построение матрицы парных сравнений 2-го уровня. На основе этих критериев построим матрицу парных сравнений второго уровня, где строки и столбцы составляют выбранные критерии. Сравнение критериев проведём по шкале относительной важности согласно с таблицей 5.1:

Таким образом, получим матрицу суждений, приведённую в Таблице 1.2- Матрица парных сравнений 2-го уровня.

Таблица 1.1.Шкала относительной важности

Интенсивность относительной важности	Определение
1	Равная важность
3	Умеренное превосходство одного над другим
5	Существенное или сильное превосходство
7	Значительное превосходство
9	Очень сильное превосходство
2,4,6,8	Промежуточные значения между двумя соседними суждениями

Таблица 1.2. Матрица парных сравнений 2-го уровня

Критерий	Быстродействие	Стоимость	Используемые ресурсы	Возможность аутентификации сервера	Возможность шифрования паролей и данных
Быстродействие	1	3	1/5	2	1/5
Стоимость	1/3	1	1/3	1/2	1/5
Используемые ресурсы	5	3	1	4	1/3
Возможность аутентификации сервера	1/2	2	1/4	1	1/3
Возможность шифрования паролей и данных	5	5	3	3	1

Оценка критериев (второй уровень иерархии). Синтез локальных приоритетов для матрицы парных сравнений 2-го уровня.

Вычислим вектор локальных приоритетов для составленной матрицы, используя формулу:

где $i = 1, m$ и $J = 1, 1)$

$$b_1 = \sqrt[5]{1 \times 3 \times 1 \times 5 \times 5} = \sqrt[5]{75} = 0.7517;$$

$$b_2 = \sqrt[5]{1 \times 1 \times 3 \times 3 \times 5} = \sqrt[5]{45} = 0.4066;$$

$$b_3 = \sqrt[5]{5 \times 3 \times 1 \times 4 \times 3} = \sqrt[5]{180} = 1.8206;$$

$$b_4 = \sqrt[5]{5 \times 2 \times 3 \times 1 \times 3} = \sqrt[5]{90} = 0.6084;$$

$$b_5 = \sqrt[5]{5 \times 5 \times 3 \times 3 \times 1} = \sqrt[5]{225} = 2.9542$$

Найдем сумму всех значений по формуле:

$$b_i = \frac{1}{n}, \text{ где } n = 5, (1.2)$$

$$B = 0.7517 + 0.4066 + 1.8206 + 0.6084 + 2.9542 = 6.5415.$$

Рассчитаем значения компонент вектора локальных переменных по формуле:

$$x_i = \frac{b_i}{\sum_{i=1}^n b_i}, (1.3)$$

$$x_1 = \frac{0.7517}{6.5415} = 0.1149; \quad x_2 = \frac{0.4066}{6.5415} = 0.0622;$$

$$x_3 = \frac{1.8206}{6.5415} = 0.2783; \quad x_4 = \frac{0.6084}{6.5415} = 0.0930;$$

$$x_5 = \frac{2.9542}{6.5415} = 0.4516.$$

Проверим нормализацию полученных значений по формуле:

$$\sum_{i=1}^n x_i = 1, (1.4)$$

$$x' = 0.1149 + 0.0622 + 0.2783 + 0.0930 + 0.4516 = 1.$$

Рассчитаем относительную погрешность полученного значения по формуле:

$$A = J \frac{|x - x'|}{x} \times 100\%, (1.5)$$

$$A = J \frac{|1 - 1|}{1} \times 100\% = 0\%.$$

При заданной точности вычисление вектора локальных приоритетов произведено без погрешности.

Исследование на согласованность матрицы парных сравнений 2-го уровня.

Необходимо провести анализ согласованности матрицы второго уровня. Для этого вычислим сумму элементов каждого столбца матрицы суждений второго порядка по формуле:

$$y_i = \sum_{j=1}^n a_{ij} x_j$$

$$\text{где } y_i = \sum_{j=1}^n a_{ij} x_j, (1.6)$$

$$y_1 = 1 + 1 + 5 + 1 + 5 = 11.8333;$$

$$y_2 = 3 + 1 + 3 + 2 + 5 = 14;$$

$$y_3 = 1 + 1 + 1 + 1 + 3 = 4.7833;$$

$$y_4 = 2 + 1 + 4 + 1 + 3 = 10.5000;$$

$$y_5 = 1 + 1 + 1 + 1 + 1 = 2.0667.$$

Далее определим наибольшее собственное значение матрицы суждений по формуле:

$$\lambda_{\max} = \frac{1}{n} \sum_{i=1}^n \left(\sum_{j=1}^n a_{ij} x_j \right) / x_i, (1-7)$$

$$\lambda_{\max} = 0.1149 \times 11.8333 + 0.0622 \times 14 + 0.2783 \times 4.7833 + 0.0930 \times 10.5000 + 0.4516 \times 2.0667 = 1.3596 + 0.8708 + 1.3312 + 0.9765 + 0.9333 = 5.2715.$$

Используя полученные данные, определим индекс согласованности (ИС) по формуле:

$$IS = \frac{\lambda_{\max} - n}{n(n-1)}, (1-8)$$

где n - размерность матрицы.

$$IS = \frac{5.2715 - 5}{5 \times (5-1)} = 0.0679.$$

Используя индекс согласованности (ИС) вычислим отношение согласованности (ОС), по формуле:

$$OS = \frac{IS}{IS_{\text{сл}}} \times 100\%, (1.9)$$

где ОС - случайная согласованность.

ОС - случайная согласованность, является табличным значением для матрицы конкретного размера, и берётся из Таблицы 1.3 - Таблица случайной

согласованности. Таблица 1.3. Таблица случайной согласованности

Размерность квадратной матрицы	1	2	3	4	5	6	7	8	9	10
СС	0	0	0.58	0.9	1.12	1.24	1.32	1.41	1.45	1.49

Для матрицы размерностью 5x5 СС=1.12.

$$OC = \frac{0.0679}{1.12} \times 100\% = 6.0625\%.$$

Так как критерием хорошей согласованности является отношение по величине составляющее менее 10%, то можно сделать вывод, что матрица, представленная в Таблице 1.2, является согласованной. Нарушение транзитивной и кардинальной согласованности не наблюдается.

Анализ результатов оценки критериев. По полученным значениям вектора локальных приоритетов можно сделать выводы о важности критериев и отсортировать критерии по их значимости.

Ниже приведены критерии в порядке убывания их приоритетов:

- 1 .Возможность шифрования паролей и данных (0.4516);
2. Используемые ресурсы (0.2783);
3. Быстродействие (0.1149);
4. Возможность аутентификации сервера (0.0930);
5. Стоимость (0.0622).

Исходя из этих данных, можно сделать вывод, что критерий «Возможность шифрования паролей и данных» является наиболее важным критерием. Критерии «Используемые ресурсы» и «Быстродействие» также имеют большой приоритет. Остальные критерии являются малозначимыми.

Оценка альтернатив (третий уровень иерархии). Проведем сравнительную оценку каждой из альтернатив относительно каждого из критериев. Обозначение протоколов: А - Telnet, Б - SSH, В - Rlogin. Получим матрицы третьего уровня иерархии размерностью $n=3$. Для каждой из матриц, вычислим векторы локальных приоритетов X_i и отношения согласованности OC .

Критерий «Быстродействие».

для критерия

Таблица 1.4. Матрица парных сравнений

«Быстродействие»

Быстродействие	A	Б	В
A	1	1/4	5
Б	4	1	8
В	1/5	1/8	1

ssh: не установлен В настоящий момент в системе работают следующие пользователи: alim	Вычислим собственный вектор локальных приоритетов матрицы, используя формулу (1.1):
groupmod [1 /etc/group] Управление пользователями	Статус

$$b = \sqrt[3]{1/4 \times 1 \times 8} = \sqrt[3]{1/32} = 3.1748$$

$$L = \sqrt[3]{1 - x - x} = \sqrt[3]{1 - 0.0250} = 0.2924$$

b

Найдем сумму всех значений ' по формуле (1.2):

$$B = 1.0772 + 3.1748 + 0.2924 = 4.5444$$

Рассчитаем значения компонент вектора локальных переменных по формуле

(1.3):

$$r_1 = \frac{1.0772}{4.5444} = 0.2370, \quad z_1 = \frac{3.1748}{4.5444} = 0.6986, \quad z_2 = \frac{0.2924}{4.5444} = 0.0643$$

Проверим нормализацию полученных значений по формуле (1.4):

$$Z^1 = 0.2370 + 0.6986 + 0.0643 = 0.9999$$

Рассчитаем относительную погрешность полученного значения по формуле

(1.5):

$$\Delta = \frac{|1 - 0.9999|}{1} \times 100\% = 0.01\%$$

При вычислении вектора локальных приоритетов имеем относительную погрешность 0.01%. Полученное значение погрешности удовлетворяет возможному отклонению от истинного значения.

Для проверки согласованности вычислим сумму элементов каждого из столбцов матрицы по формуле (1.6):

$$y_l = 1 + 4 + 1 = 5.2000;$$

$$v_7 = 1 + 1 + 1 = 1.3750;$$

$$y_3 = 5 + 8 + 1 = 14.$$

Определим наибольшее собственное значение матрицы суждений по формуле (1.7):

$$L_{max} = 0.2370 \times 5.2000 + 0.6986 \times 1.3750 + 0.0643 \times 14 = 1.2324 + 0.9606 + 0.9002 = 3.0932.$$

Используя полученные данные, определим индекс согласованности (ИС) по формуле (1.8):

$$IS = \frac{3.0932 - 3}{3 - 1} = 0.0466.$$

Вычислим отношение согласованности (ОС) по формуле (1.9):

Для матрицы размерностью 3x3 $CS=0.58$.

$$OS = 0.0466 \times 100 \% = 8.0344\%.$$

$$0.58$$

Так как критерием хорошей согласованности является отношение по величине составляющее менее 10%, то можно сделать вывод, что матрица, представленная в Таблице 1.4, является согласованной. Нарушение транзитивной и кардинальной согласованности не наблюдается.

Критерий «Используемые ресурсы». Вычислим собственный вектор локальных приоритетов для матрицы, используя формулу (1.1):

Таблица 1.6. Матрица парных сравнений для критерия «Используемые ресурсы»

Используемые ресурсы	А	Б	В
А	1	1/5	1/4
Б	5	1	3
В	4	1/3	1

$$A = 3 \times 1 \times 1 \times 1 = 3 \text{ л/005} = 0.3684;$$

$$V = 5 \times 4$$

$$b_3 = 3 \times 4 \times 1 \times 1 = 12 \text{ л/005} = 1.1006.$$

b

Найдем сумму всех значений ' по формуле (1.2):

$$b_2 = \sqrt[3]{1/5 \times 1 \times 3} = \sqrt[3]{0.6} = 0.8434$$

$$B = 0.3684 + 2.4662 + 1.1006 = 3.9352.$$

Рассчитаем значения компонент вектора локальных переменных по формуле (5.3):

$$z_1 = \frac{0.0936}{3.9352} = 0.0238$$

$$z_2 = \frac{2.4662}{3.9352} = 0.6267$$

$$z_3 = \frac{1.1006}{3.9352} = 0.2797$$

Проверим нормализацию полученных значений по формуле (1.4):

$$Z' = 0.0238 + 0.6267 + 0.2797 = 1.$$

Рассчитаем погрешность полученного значения по формуле (1.5):

$$A = J \frac{|1-1|}{1} \times 100\% = 0\%.$$

При заданной точности вычисление вектора локальных приоритетов произведено без погрешности.

Для проверки согласованности вычислим сумму элементов каждого столбца матрицы по формуле (1.6):

$$v_1 = 1 + 5 + 4 = 10;$$

$$v_2 = \frac{1}{5} + 1 + \frac{1}{3} = 1.5333;$$

$$v_3 = \frac{1}{4} + 3 + 1 = 4.2500.$$

Далее определим наибольшее собственное значение матрицы суждений по формуле (1.7):

$$\lambda_{m!lx} = 0.0936 \times 10 + 0.6267 \times 1.5333 + 0.2797 \times 4.2500 = 0.9360 + 0.9609 + 1.1887 = 3.0856.$$

Используя полученные данные, определим индекс согласованности (ИС) по формуле (1.8):

Вычислим отношение согласованности (ОС) по формуле (1.9):

Для матрицы размерностью 3x3 $CC=0.58$.

$$OC = \frac{0.0498}{0.58} \times 100\% = 7.3793\%.$$

Так как критерием хорошей согласованности является отношение по величине составляющее менее 10%, то можно сделать вывод, что матрица, представленная в Таблице 1.6 является согласованной. Нарушение транзитивной и кардинальной согласованности не наблюдается.

Критерий «Возможность аутентификации сервера».

Таблица 1.7. Матрица парных сравнений для критерия «Возможность аутентификации сервера»

Возможность аутентификации сервера	А	Б	В
А	1	1/5	1
Б	5	1	5
В	1	1/5	1

Вычислим собственный вектор локальных приоритетов матрицы, используя формулу (1.1):

$$B_1 = \sqrt[3]{1 \times 1 \times 1} = \sqrt[3]{1} = 0.5848;$$

$$B_2 = \sqrt[3]{1/5 \times 1 \times 5} = \sqrt[3]{1} = 0.9240$$

$$B_3 = \sqrt[3]{1 \times 1 \times 1} = \sqrt[3]{1} = 0.5848.$$

Б

Найдем сумму всех значений λ по формуле (1.2):

$$B = 0.5848 + 0.9240 + 0.5848 = 2.0936.$$

Рассчитаем значения компонент вектора локальных переменных по формуле (5.3):

Проверим нормализацию полученных значений по формуле (1.4):

$$Z^7 = 0.1429 + 0.7143 + 0.1429 = 1.0001$$

Рассчитаем относительную погрешность полученного значения по формуле

(1.5):

$$A = J \frac{|1 - 1.0001|}{1} \times 100\% = 0.01\%.$$

При вычислении вектора локальных приоритетов имеем относительную погрешность 0.01%. Полученное значение погрешности удовлетворяет возможному отклонению от истинного значения.

Для проверки согласованности вычислим сумму элементов каждого столбца матрицы по формуле (1.6):

$$y_2 = 1 + 5 + 1 = 7;$$

$$v_7 = 1 + 1 + 1 = 1.4000;$$

$$y_3 = 1 + 5 + 1 = 7.$$

Далее определим наибольшее собственное значение матрицы суждений по формуле (1.7):

$$D_{\max} = 0.1429 \times 7 + 0.7143 \times 1.4000 + 0.1429 \times 7 = 1.0003 \text{ и } 1 + 1.0003 = 3.0006$$

Используя полученные данные, определим индекс согласованности (ИС) по формуле (1.8):

$$IS = \frac{3.0006 - 3}{3 - 1} = 0.0003.$$

Используя полученные данные, определим индекс согласованности (ИС) по формуле (1.9):

Для матрицы размерностью 3x3 $CC=0.58$.

$$OC = \frac{0.0003}{0.58} \times 100\% = 0.0517\%.$$

Так как критерием хорошей согласованности является отношение по величине составляющее менее 10%, то можно сделать вывод, что матрица, представленная в Таблице 1.7 является согласованной. Нарушение транзитивной и кардинальной согласованности не наблюдается.

Критерий «Возможность шифрования паролей и данных».

Таблица 1.8. Матрица парных сравнений для критерия «Возможность шифрования паролей и данных»

Возможность шифрования паролей и данных	A	Б	В
A	1	1/7	3
Б	7	1	9
В	1/3	1/9	1

Вычислим собственный вектор локальных приоритетов для матрицы, используя формулу (1.1):

$$B_1 = \sqrt[3]{1 \times 1 \times 3} / 0.4286 = 0.7539, B_2 = \sqrt[3]{7 \times 1 \times 9} / 63 = 3.9791$$

$$K = \frac{3}{3} \times \frac{1}{9} = 0.3333.$$

5

Найдем сумму всех значений ' по формуле (1.2):

$$B = 0.7539 + 3.9791 + 0.3333 = 5.0663.$$

Рассчитаем значения компонент вектора локальных переменных по формуле (1.3):

$$z_1 = \frac{0.7539}{5.0663} = 0.1488;$$

$$z_2 = \frac{3.9791}{5.0663} = 0.7854;$$

$$z_3 = \frac{0.3333}{5.0663} = 0.0658.$$

Проверим нормализацию полученных значений по формуле (1.4):

$$Z^I = 0.1488 + 0.7854 + 0.0658 = 1.$$

Рассчитаем относительную погрешность полученного значения по формуле (1.5):

$$\Delta = \frac{|1 - 1|}{1} \times 100\% = 0\%.$$

При заданной точности вычисление вектора локальных приоритетов произведено без погрешности.

Для проверки согласованности вычислим сумму элементов каждого столбца

матрицы по формуле (1.6):

$$V_1 = 1 + 7 + \frac{1}{3} = 8.3333;$$

$$V_7 = 1 + 1 + \frac{1}{9} = 1.2540;$$

$$= 3 + 9 + 1 = 13.$$

Далее определим наибольшее собственное значение матрицы суждений по формуле (5.7):

$$P_{max} = 0.1488 \times 8.3333 + 0.7854 \times 1.2540 + 0.0658 \times 13 = 1.2400 + 0.9849 + 0.8554 = 3.0803.$$

Используя полученные данные, определим индекс согласованности (ИС) по формуле (1.8):

$$ИС = \frac{3.0803 - 3}{3 - 1} = 0.0401.$$

Вычислим отношение согласованности (ОС) по формуле (1.9):

Для матрицы размерностью 3x3 $СС=0.58$.

$$ОС = \frac{0.0401}{0.58} \times 100 \% = 6.9224\%.$$

Так как критерием хорошей согласованности является отношение по величине составляющее менее 10%, то можно сделать вывод, что матрица, представленная в Таблице 1.8 является согласованной. Нарушение транзитивной и кардинальной согласованности не наблюдается.

Анализ результатов оценки альтернатив. По полученным значениям векторов локальных приоритетов можно сделать выводы о важности альтернатив для каждого из критериев.

Критерий «Быстродействие».

1. Альтернатива Б (0.6986);

2. Альтернатива А (0.2370);

3. Альтернатива В (0.0643).

Альтернатива Б имеет значительное преимущество в данном критерии над остальными альтернативами.

Критерий «Используемые ресурсы».

1. Альтернатива Б (0.6267);

2. Альтернатива В (0.2797);

3. Альтернатива А (0.0936).

Альтернатива Б имеет значительное преимущество в данном критерии над остальными альтернативами.

Критерий «Возможность аутентификации сервера».

1. Альтернатива Б (0.7143);

2. Альтернатива А (0.1429);

3. Альтернатива В (0.1429).

По данному критерию уверенно лидирует Альтернатива Б, другие альтернативы равнозначны.

Критерий «Возможность шифрования паролей и данных».

1. Альтернатива Б (0.7854);

2. Альтернатива А (0.1488);

3. Альтернатива В (0.0658).

Альтернатива Б имеет преимущество в данном критерии над остальными альтернативами.

Из полученных данных можно сделать вывод, что Альтернатива Б лидирует во всех пяти критериях («Быстродействие», «Стоимость», «Используемые ресурсы», «Возможность аутентификации сервера», «Возможность шифрования паролей и данных»).

Количественные оценки вкладов критериев в конечный результат. Определим количественные оценки вкладов критериев в конечный результат.

Альтернатива А:

Рассмотрим вклад каждого из критериев подробнее и сделаем выводы.

Критерий «Быстродействие» внес следующий вклад в процентном отношении:

$$A = \frac{0.2783 \times 0.0936}{0.1282} \times 100\% = 21.2413\%.$$

Критерий «Используемые ресурсы» внес следующий вклад в процентном отношении:

$$A = \frac{0.2783 \times 0.0936}{0.1282} \times 100\% = 21.2413\%.$$

Критерий «Возможность аутентификации сервера» внес следующий вклад в процентном отношении:

$$A = \frac{0.0930 \times 0.1429}{0.1282} \times 100\% = 10.3664\%.$$

Критерий «Возможность шифрования паролей и данных» внес следующий вклад в процентном отношении:

$$A = \frac{0.4516 \times 0.1488}{0.1282} \times 100\% = 52.4166\%.$$

Исходя из этих данных, можно сделать вывод, что критерии «Быстродействие», «Используемые ресурсы» и «Возможность шифрования паролей и данных» внесли наибольший вклад в конечный результат данной альтернативы по причине того, что имели наивысший приоритет. Остальные критерии не оказали сильного влияния на результат.

В результате полезность данной альтернативы оказалась низкой (0.1282), так как имела не высокие показатели по критериям.

Альтернатива Б:

Критерий «Быстродействие» внес следующий вклад в процентном отношении:

$$A = \frac{0.1149 \times 0.6986}{0.7297} \times 100\% = 11.0003\%.$$

Критерий «Используемые ресурсы» внес следующий вклад в процентном отношении:

$$A = \frac{0.2783 \times 0.6267}{0.7297} \times 100\% = 23.9019\%.$$

Критерий «Возможность аутентификации сервера» внес следующий вклад в процентном отношении:

Критерий «Возможность шифрования паролей и данных» внес следующий вклад в процентном отношении:

$$A = \frac{X_{\text{к}} \cdot X_{\text{д}}}{V} \times 100\% = \frac{0.4516 \times 0.7854}{0.7297} \times 100\% = 48.6072\% \text{ «}_{\text{о}}$$

Исходя из этих данных, можно сделать вывод, что критерии «Быстродействие», «Используемые ресурсы» и «Возможность шифрования паролей и данных» внесли наибольший вклад в конечный результат данной альтернативы по причине того, что имели наивысший приоритет. Остальные критерии не оказали сильного влияния на результат.

В результате полезность данной альтернативы оказалась самой высокой(0.7297) из-за того, что критерии, по которым данная альтернатива выигрывает у других, имеют самый высокий приоритет.

Альтернатива В:

Рассмотрим вклад каждого из критериев подробнее и сделаем выводы.

Критерий «Быстродействие» внес следующий вклад в процентном отношении:

$$A = \frac{X_{\text{т}} \cdot X_{\text{с}}}{V} \times 100\% = \frac{0.1149 \times 0.0643}{0.1422} \times 100\% = 5.1955\% \text{ «}_{\text{о}}$$

Критерий «Используемые ресурсы» внес следующий вклад в процентном отношении:

$$A = \frac{X_{\text{л}} \cdot X_{\text{г}}}{V} \times 100\% = \frac{0.2783 \times 0.2797}{0.1422} \times 100\% = 54.7402\% \text{ «}_{\text{о}}$$

Критерий «Возможность аутентификации сервера» внес следующий вклад в процентном отношении:

$$A = \frac{X_{\text{л}} \cdot X_{\text{г}}}{V} \times 100\% = \frac{0.0930 \times 0.1429}{0.1422} \times 100\% = 9.3453\% \text{ «}_{\text{о}}$$

Критерий «Возможность шифрования паролей и данных» внес следующий вклад в процентном отношении:

$$A = \frac{X_{\text{л}} \cdot X_{\text{г}}}{V} \times 100\% = \frac{0.4516 \times 0.0658}{0.1422} \times 100\% = 20.8\% \text{ «}_{\text{о}}$$

Исходя из этих данных, можно сделать вывод, что критерии

«Используемые ресурсы» и «Возможность шифрования паролей и данных» внесли наибольший вклад в конечный результат данной альтернативы. В результате полезность данной альтернативы оказалась низкой (0.1422), так как имела не высокие показатели по критериям.

Из этих данных можно сделать вывод, что наивысшую полезность имеет Альтернатива Б, за ней следует Альтернатива В с большим отставанием и с небольшим отставанием идёт Альтернатива А.

В результате проведенной работы МАИ показал, что протокол SSH предпочтительнее остальных протоколов.

2. ОПТИМИЗАЦИЯ ПРОГРАММНОГО МОДУЛЯ И ТЕСТИРОВАНИЕ ПРОГРАММНОГО ПРОДУКТА

2.1. Функциональное назначение

Общие сведения. Цель ВКР - создание программного модуля оптимизации работы сети на платформе операционной системы Linux по протоколам управления.

Для функционирования программы требуется IBM-совместимый персональный компьютер на основе микропроцессора Intel Pentium 667 MHz или более производительная система (а также их аналоги), минимум 32 MB ОЗУ, минимум 10 MB свободного места на диске. Операционная система Linux Debian..

Программа разработана на языке программирования высокого уровня JAVA в среде визуального проектирования NetBeansIDE 6.7 [11]. Листинг исходного кода программы приведён в Приложении А.

Программный продукт в целом предназначен для управления пользователями и группами пользователей, для управления доступа к файлам и каталогам как на локальном компьютере, так и на удаленном компьютере внутри локальной сети. Также программный продукт обеспечивает средства для установки, запуска и настройки службы удаленного доступа ssh. Конечной целью

является оптимизация работы сети и сетевого администратора, связанная с улучшением работы по администрированию пользователей и файловых ресурсов внутри сети. В данной работе разрабатывается модуль, который является частью комплексной программной системы, и который в свою очередь не взаимодействует с другими модулями системы. Это объясняется тем, что по своей функциональности модуль является независимым от других разрабатываемых модулей в комплексной ВКР.

2.2. Руководство пользователя

Для начала работы с программой необходимо запустить файл Diploma.jar на рабочем столе. После запуска файла появится главное окно:

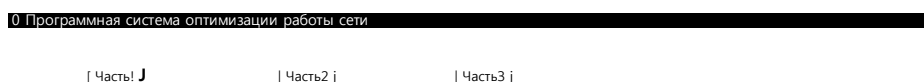


Рисунок 2.1. Главное окно программы

Выбрав «Часть 1» осуществляется переход на следующее окно, в котором пользователь (администратор) начинает работу с протоколами управления.

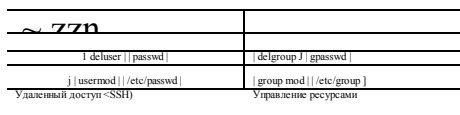


Рисунок 2.2. Окно «Протоколы управления»

В самом низу окна имеется функция включения подсказок, которая при наведении на какой либо объект, разъясняет суть этого объекта, чтобы пользователь без труда мог выполнить необходимые ему процедуры.

Рассмотрим каждую подсистему подробнее.

1. Управление пользователями

1.1 Кнопка «adduser» -добавление пользователя в систему. При нажатии на кнопку «adduser» произойдет переход в диалоговое окно, в котором имеются поля и параметры для выбора, т.е. необходимо задать имя нового пользователя, выбрать необходимый параметр и нажать кнопку «Ок».

Имя пользователя: `qwerty`

Параметры:

еще параметр `U`

Системный
`jdduser qwerty`

ОК | Сброс | Отмена

Рисунок 2.3. Окно добавления нового пользователя

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя для продолжения работы.

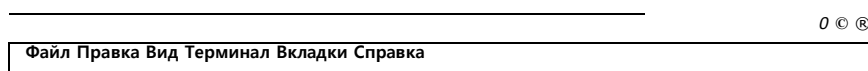


Рисунок 2.4. Окно терминала

После того как пароль принят необходимо заполнить такие поля как пароль, информацию о пользователе и сохранить.

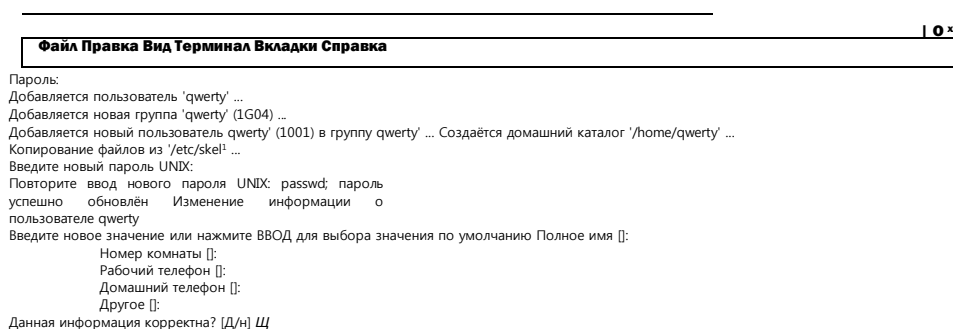


Рисунок 2.5. Окно редактирования информации о пользователе

В итоге мы получаем новую учетную запись с именем «qwerty».

1.2 Кнопка «usermod» - изменение параметров учетной записи

пользователя. При нажатии на кнопку «usermod» произойдет переход в диалоговое окно, в котором имеются поля и параметры для выбора, т.е. необходимо задать имя пользователя, у которого хотим произвести изменения, выбрать необходимый параметр и нажать кнопку «Ок». Например, приведем случай, когда мы хотим изменить имя этого пользователя.

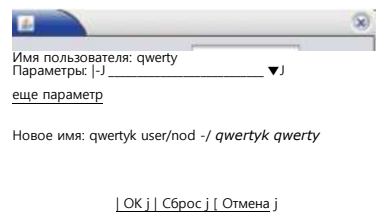


Рисунок 2.6. Окно изменения параметров учетной записи пользователя

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого произойдет изменение имени.

1.3 Кнопка «users» - показывает имена пользователей, которые в данный момент работают в системе. Результат выполнения этой команды отображается в окне «Протоколы управления» в области «Статус».

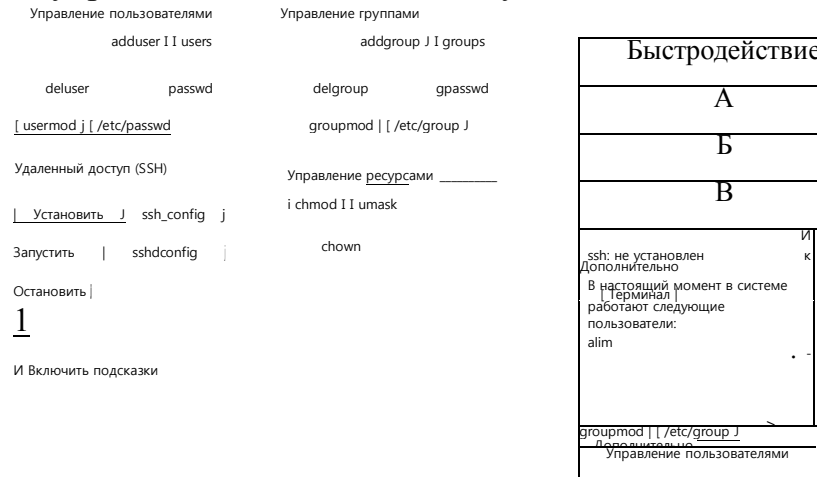


Рисунок 2.7. Окно «Протоколы управления»

1.4 Кнопка «passwd» - изменение пароля пользователя. При нажатии на кнопку «passwd» произойдет переход в диалоговое окно, в котором имеются поля и параметры для выбора, т.е. необходимо задать имя пользователя, у которого хотим произвести изменения, выбрать необходимый параметр и нажать кнопку «Ок». Например, приведем случай, когда мы хотим просто без каких либо параметров изменить пароль этого пользователя.

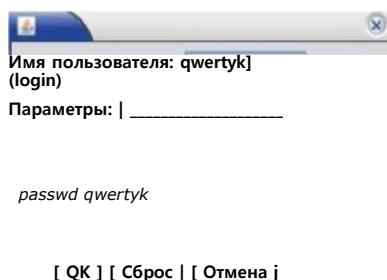


Рисунок 2.8. Окно изменения пароля пользователя

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого необходимо ввести новый пароль и пароль обновиться.

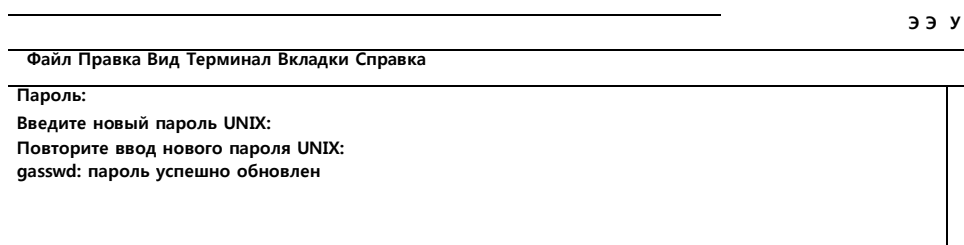


Рисунок 2.9. Окно терминала для изменения пароля

1.5 Кнопка «deluser» - удаление пользователя из системы. При нажатии на кнопку «deluser» произойдет переход в диалоговое окно, в котором имеются поля и параметры для выбора, т.е. необходимо задать имя пользователя, которого хотим удалить, выбрать необходимый параметр и нажать кнопку «Ок». Например, приведем случай, когда мы хотим удалить

пользователя «qwertyk», а также его домашний каталог и хранилище почты.

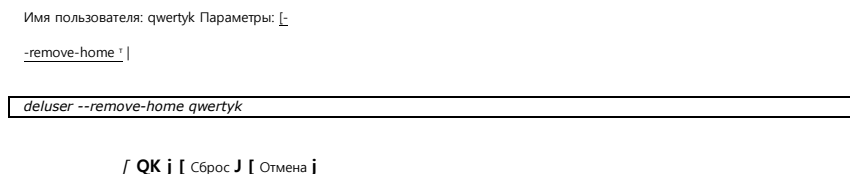


Рисунок 2.10. Окно удаления пользователя

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого этот пользователь будет удален из системы.

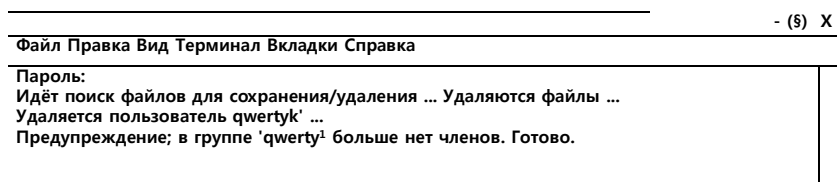


Рисунок 2.11. Окно терминала удаления пользователя

1.6 Кнопка «/etc/passwd» - предназначена для просмотра файла с перечнем пользовательских учетных записей, известных системе.

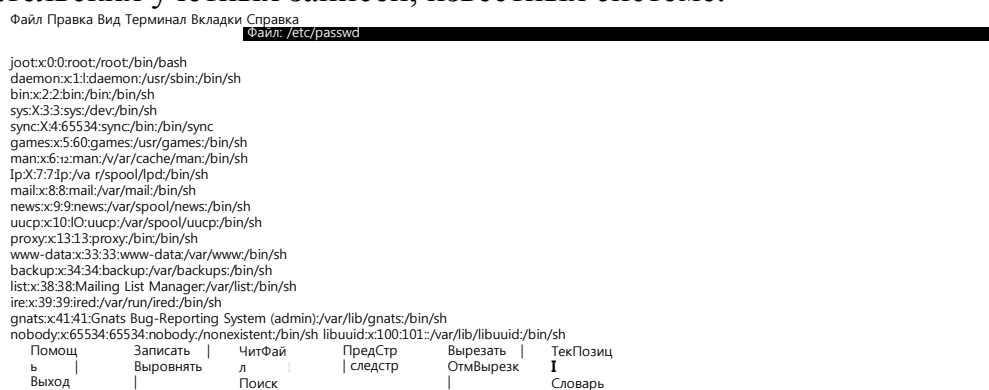


Рисунок 2.12. Фрагмент файла учетных записей

При нажатии на кнопку «/etc/passwd» произойдет переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После

этого откроется файл «passwd», содержащий все учетные записи, известные системе.

2. Управление группами.

2.1 Кнопка «addgroup» -добавление группы в систему. При нажатии на кнопку «addgroup» произойдет переход в диалоговое окно, в котором имеются поля и параметры для выбора, т.е. необходимо задать имя группы, выбрать необходимый параметр и нажать кнопку «Ок».

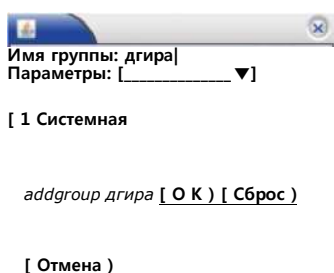


Рисунок 2.13. Окно добавления группы в систему

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого эта группа будет создана.

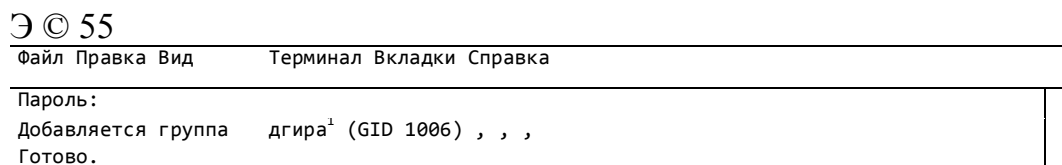


Рисунок 2.14. Окно терминала добавления группы

2.2 Кнопка «groupmod» -изменение определения группы в системе. При нажатии на кнопку «groupmod» произойдет переход в диалоговое окно, в котором имеются поля и параметры для выбора, т.е. необходимо задать имя группы, выбрать необходимый параметр и нажать кнопку «Ок». Например, приведем случай, когда мы хотим изменить имя группы.



Имя группы:	grupa
Параметры:	-n x
Новое имя:	grupas

groupmod -n grupas grupa

[OK] [Сброс] | Отмена

Рисунок 2.15. Окно изменения определения группы

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого произойдет изменение имени группы.

2.3 Кнопка «groups» - показывает имена групп, в которые входят пользователи. При нажатии на кнопку «groups» произойдет переход в диалоговое окно, в котором имеется один параметр, т.е. необходимо задать имя пользователя или имена пользователей через пробел и нажать кнопку «Ок».

Имя пользователя:	
	user
[OK] [Сброс] [Отмена]	

Рисунок 2.16. Окно для введения имени пользователя

Результат выполнения этой команды отображается в окне «Протоколы управления» в области «Статус».

2.4 Кнопка «grasswd» - управление группами и их паролями. При нажатии на кнопку «grasswd» произойдет переход в диалоговое окно, в котором имеются поля и параметры для выбора, т.е. необходимо задать имя группы, в которой хотим произвести изменения, выбрать необходимый параметр и нажать кнопку «Ок».

Управление пользователями	Управление группами	Статус	О ' -я*
---------------------------	---------------------	--------	---------

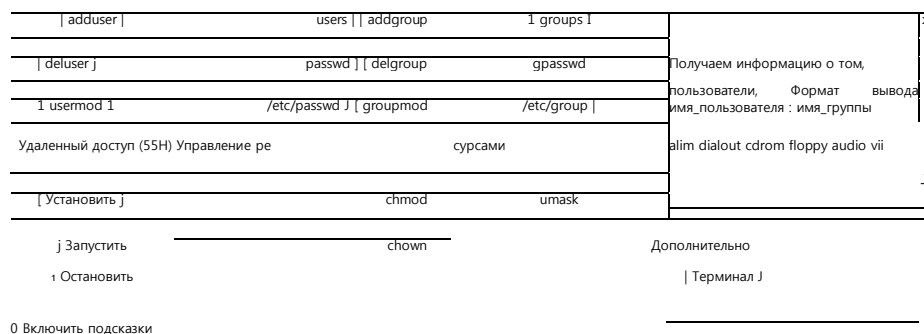


Рисунок 2.17. Окно «Протоколы управления»

Например, приведем случай, когда мы хотим просто удалить пароль группы.



Рисунок 2.18 - Окно управления группами и их паролями

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого пароль будет удален.

2.5 Кнопка «delgroup» - удаление группы из системы. При нажатии на кнопку «delgroup» произойдет переход в диалоговое окно, в котором имеются поля и параметры для выбора, т.е. необходимо задать имягруппы, которую хотим удалить, выбрать необходимый параметр и нажать кнопку «Ок». Например, приведем случай, когда мы хотим удалить группу «grupas» без каких либо параметров.

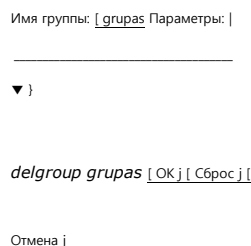


Рисунок 2.19. Окно удаления группы

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого эта группа будет удалена из системы.

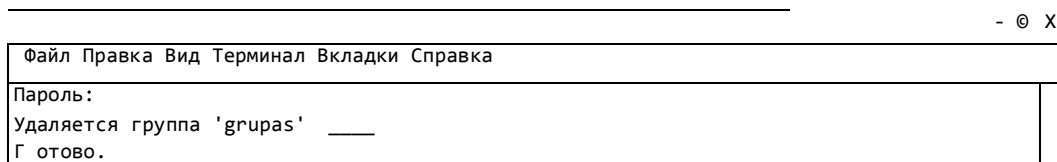


Рисунок 2.20. Окно терминала удаления группы

2.6 Кнопка «/etc/group» - предназначена для просмотра файла с перечнем групп, известных системе. При нажатии на кнопку «/etc/group» произойдет переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого откроется файл «group», содержащий все группы, известные системе.

3. Управление ресурсами.

3.1 Кнопка «chmod» -изменение прав доступа к файлам и директориям.

При нажатии на кнопку «chmod» произойдет переход в диалоговое окно, в котором имеются поля и параметры для выбора, т.е. необходимо задать файл, выбрать необходимые параметры и нажать кнопку «Ок». Например, приведем случай, когда мы хотим файлу «15.jpg» присвоить права чтения, записи и запуска для всех пользователей.

```
Файл Правка Вид Терминал Вкладки Справка
GNU nano 2.0.1 /etc/group
r]oot
:x:0:
daemon:x:
1:
bin:x:2:
sys:x:3:
adm:x:4:
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
:
man:x:12:
proxy:x:1
3:
kmem:x:15
:
dialout:x
:20:alim
fax:x:21:
voice:x:2
2:
cdrom:x:2
```

```

4:alim
floppy:x:
25:alim
] Помощь      Щ Записать В      2 ЧитФайл      ПредСтр      Щ Вырезать Ц      ТекПозиц
3 Выход      НИ Вывернуть 1      ц Поиск      Щ СледСтр      Я ОтмВырезк Ш      Словарь

```

Рисунок 2.21. Фрагмент файла «group»

Файл: /home/alim/15.jpg

Пользователь: [a ▼]

Права: [+rwx _____]

Параметры: [_____]

`chmod a+rwx /home/atim/15.jpg`

[QK] [Сброс] [Отмена]

Рисунок 2.22. Окно изменения прав доступа к файлу

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого права вступят в силу. Результат можно посмотреть в свойствах данного файла.

Основные Эмблемы	Правы	Открывать в программе	Заметки	Изображение	Документ
Владелец:	ий				
Доступ:	Чтение и запись Z				
Группа:	alim				
Доступ:	Чтение и запись Z				
Остальные					
Доступ:	Чтение и запись ^				
Выполнение:	P Позволять выполнение файла как программы				
Контекст SELinux:	Последние изменения:	неизвестно			
		Вто 24 Апр 2012 11:03:30			
^Справка					ЭС. Закрыть

3.2 Кнопка «chown» -изменение владельца и/или группы для указанных файлов. При нажатии на кнопку «chown» произойдет переход в диалоговое окно, в котором имеются поля и параметры для выбора, т.е. необходимо задать файл, выбрать необходимые параметры и нажать кнопку «Ок». Например приведем случай, когда мы хотим файлу «15.jpg» присвоить нового владельца и группу.

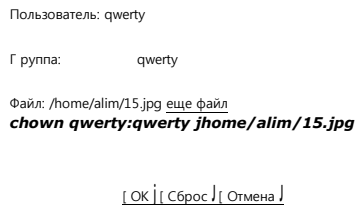
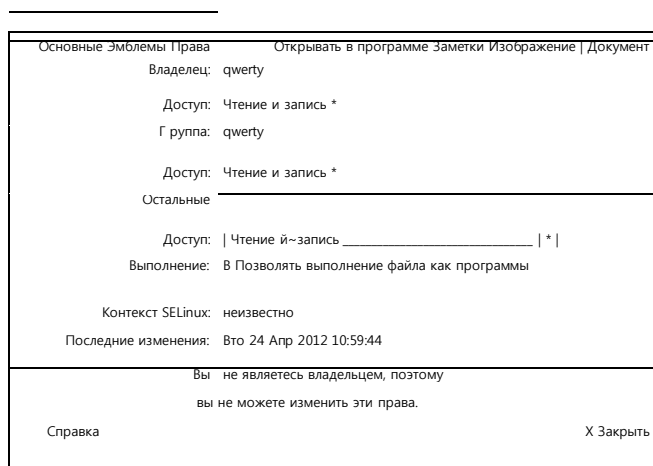


Рисунок 2.24. Окно изменения владельца и группы для файла

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого этому файлу будут присвоены новый владелец и группа. Результат можно посмотреть в свойствах данного файла.



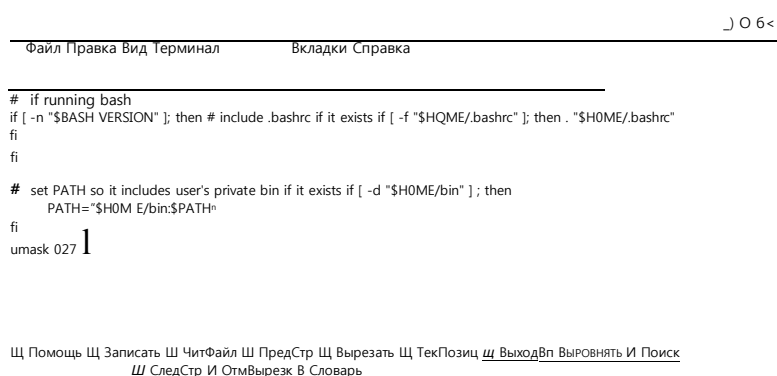
3.3 Кнопка «umask» -изменение прав доступа, которые присваиваются новым файлами директориям по умолчанию. При нажатии на кнопку «umask» произойдет переход в диалоговое окно, в котором имеются поля и параметры для выбора, т.е. необходимо выбрать пользователя, задать маску и нажать кнопку «Ок». Например приведем случай, когда мы хотим предоставить все права пользователю «alim», запретить читать файл членам группы и не давать никаких прав другим пользователям.



Рисунок 2.25. Свойства файла «15.jpg»

Рисунок 2.26. Окно изменения прав доступа по маске

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого этому пользователю предоставляться заданные права. Результат можно посмотреть, набрав в терминале `папо /home/alim/.profile`.



```
# if running bash
if [ -n "$BASH_VERSION" ]; then # include .bashrc if it exists if [ -f "$HOME/.bashrc" ]; then . "$HOME/.bashrc"
fi

# set PATH so it includes user's private bin if it exists if [ -d "$HOME/bin" ]; then
PATH="$HOME/bin:$PATH"
fi
umask 027
```

Рисунок 2.27. Фрагмент файла .profile для пользователя с именем user

4. Удаленный доступ (SSH). При нажатии на кнопку «Часть1», т.е. переход из главного окна в окно «Протоколы управления» происходит проверка службы ssh, установлена ли она в системе или нет. Результат показывается в статусе.

Быстродействие	А		Б
А	addgroup	1	1/4
Б passwd	delgroup	4 gpasswd	1
[usermod j [/etc/passwd] Удаленный	groupmod	1/5 /etc/group	1/8
<div> <div> <div>ssh: доступ (SSH) не установлен</div> <div>В настоящий момент в системе работают следующие пользователи: alim</div> <div>Остановить i</div> </div> <div>Управление ресурс.</div> <div>Дополнительно</div> <div>Терминал</div> </div>			
groupmod [/etc/group]	Управление группами		Статус

4.1 Кнопка «Установить» - предназначена для установки службы удаленного flOCTyna(ssh). При нажатии на кнопку «Установить» осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого начнется установка службы.

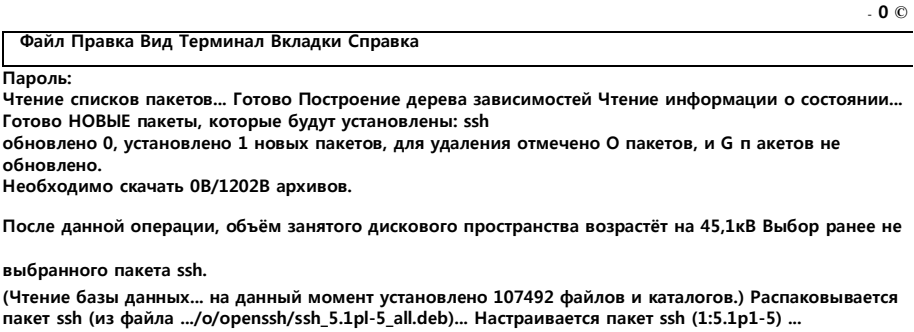


Рисунок 2.29. Фрагмент терминала установки ssh

4.2 Кнопки «ssh config» «sshd config» - предназначены для настроек конфигурационных файлов клиента и сервера.

```

Файл Правка Вид Терминал Вкладки Справка
GNU nano 2.0.7 Файл: /etc/ssh/ssh config
# This is the ssh client system-wide configuration file.

See
# ssh config(5) for more information. This file provides defaults for
# users, and the values can be changed in per-user configuration files
# or on the command line.

# Configuration data is parsed as follows:
# 1. command line options
# 2. user-specific file
# 3. system-wide file
# Any configuration value is only changed the first time it is set.
# Thus, host-specific definitions should be at the beginning of the
# configuration file, and defaults at the end.

# Site-wide defaults for some commonly used options. For a comprehensive
# list of available options, their meanings and defaults, please see the
# sshconfig(5) man page.

Host
Помощь
Выход
| ТекПозиц
Словарь щ|

```

Рисунок 2.30. Фрагмент файла sshconfig

```

Файл Правка Вид Терминал Вкладки Справка
GNU nano 2.0.7 Файл: /etc/ssh/sshd config
0 Package generated configuration file
# See the sshd(8) manpage for details

# What ports, IPs and protocols we listen for Port 22
# Use these options to restrict which interfaces/protocols sshd will bind to #ListenAddress ::
#ListenAddress 0.0.0.0 Protocol 2
# HostKeys for protocol version 2 HostKey /etc/ssh/ssh_host_rsa_key HostKey
/etc/ssh/ssh_host_dsa_key #Privilege Separation is turned on for security
UsePrivilegeSeparation yes

# Lifetime and size of ephemeral version 1 server key KeyRegenerationInterval 3600
ServerKeyBits 768

```

Прочитано 77 строк				
Щ Помощь	Щ Записать В	S ЧитФайл	щ предСтр	Щ Вырезать Щ
щ Выход	ай Вывернуть	" Поиск	Щ СледСтр	Щ ОтмВырезк Б

Рисунок 2.31. Фрагмент файла sshdconfig

4.3 Кнопка «Запустить» - предназначена для запуска службы удаленного доступа (ssh) на локальном компьютере для обеспечения возможности общаться с другими компьютерами в сети.

При нажатии на кнопку «Запустить» осуществится переход в системный терминал, в котором потребуется ввести пароль суперпользователя. После этого служба запустится.

Файл Правка Вид Терминал Вкладки Справка Пароль:
Starting OpenBSD Secure Shell server: sshd.

Рисунок 2.32. Окно терминала запуска службы

4.4 Кнопка «Остановить» - предназначена для остановки службы удаленного доступа (ssh) на локальном компьютере. При нажатии на кнопку «Остановить» осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого служба остановиться.

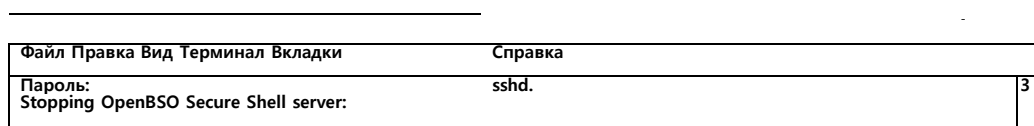


Рисунок 2.33. Окно терминала остановки службы

Также имеется дополнительно терминал, который предназначен для каких-либо иных целей.

Дополнительно

1 Терминал ,

Рисунок 2.34. Окно дополнительного терминала

При нажатии на кнопку «Терминал» осуществиться переход в системный терминал.

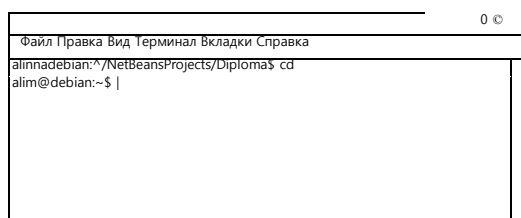


Рисунок 2.35. Окно системного терминала

В программе используются методы модульного программирования: выделение методов, реализующих отдельные подзадачи, стандартные команды

2.3. Описание логической структуры

операционной системы Linux, а также стандартные методы и функции языка Java [13].

Основные функциональные составляющие:

- `classInterfacel` - данный класс содержит описание компонентов интерфейса пользователя, а также включает в себя основные методы, реализованные в обработчиках всевозможных событий (открытие окна, закрытие окна, нажатие на кнопку, выбор элемента выпадающего списка, изменение значения текстового поля, изменение состояния флажка);

- `String openFileDialog(JDialog dialog)` - отображает окно выбора файлов, указывающее на пользовательскую директорию по умолчанию. Параметры: `dialog` - ссылка на диалоговое окно, в котором необходимо выполнить операцию. Возвращает: `pathToFile` - полный путь к выбранному пользователем файлу в виде строки;

- `String openDirectoryDialog(JDialog dialog)` - отображает окно выбора папок, указывающее на пользовательскую директорию по умолчанию. Параметры: `dialog` - ссылка на диалоговое окно, в котором необходимо выполнить операцию. Возвращает: `pathToFile` - полный путь к выбранному пользователем каталогу в виде строки;

- `void ranCommand(String command)` - Выполняет указанную команду в отдельном процессе. При этом содержимое параметра `command` разбивается на отдельные лексемы, которые являются составляющими строки команды (сама команда и ее параметры). Параметры: `command` - указанная системная команда;

- `void makeAdduserCom(String [] param)` - формирует текст команды `adduser` - команды добавления пользователя в системе Linux. Метод

считывает введенную пользователем информацию из текстовых полей и

выпадающих списков, отображаемых в диалоге создания команды adduser. Параметр param содержит один или несколько введенных параметров. По полученным данным и формату команды составляется ее текст, который помещается в специальное текстовое поле для отображения пользователю;

- void makeDeluserCom() - составляет текст команды удаления пользователя с учетом введенных входных данных и формата команды;

- void makeUsermodCom(String [] param) - формирует текст команды модификации учетной записи пользователя. Входной параметр param содержит параметры команды, остальные данные считываются со специальных элементов интерфейса диалога работы с командой;

- void makePasswdCom() - выполняет компоновку заголовка команды изменения пароля пользователя, ее параметров и имени пользователя;

- void makeAddgroupCom() - строит текст команды добавления группы;

- void makeDelgroupCom() - данный метод формирует текст команды удаления группы;

- void makeGroupmodCom() - организывает текст команды изменения определения группы в системе. Исходные данные для команды задаются пользователем в окне диалога;

- void makeGpasswdCom() - выполняет компоновку заголовка команды управления файлом групп (/etc/group) в операционной системе Linux;

- void makeChmodCom() - формирует текст команды изменения прав доступа к файлу;

- void makeChownCom() - данный метод составляет текст команды изменения владельца и группы файлов;

- void makeUmaskCom() - строит текст команды получения или установки маски режима создания файлов и отображает его в специальном текстовом поле;

- void makeSh(String filename, String[] command) - данный метод создает и формирует содержимое sh-файлов (скрипт, специально разработанный для операционной системы Linux). Имя файла передается в параметре filename,

команды на выполнение, которые должны быть включены в данный файл скриптов, располагаются в параметре `command`;

- `void makeUmaskSh(String username, String mask)` - создает и формирует скрипт-файл для выполнения команды `umask` (команды получения или установки маски режима создания файлов). Параметры: `username` - имя пользователя, для которого необходимо применить данную маску; `mask` - цифровое значение маски, записанное в переменной строкового типа;

- `void usersToCombo()` - формирует и наполняет специальный выпадающий список именами пользователей, зарегистрированных в системе;

- `void addUserDialogClean()` - данный метод выполняет очистку содержимого всех текстовых полей для ввода данных, а также устанавливает выбранным первый элемент во всех выпадающих списках в диалоге команды `adduser`;

- `void deluserDialogClean()` - выполняет замену содержимого всех текстовых полей для ввода данных на пустое, а также устанавливает выбранным первый элемент во всех выпадающих списках в диалоге команды удаления пользователей;

- `void usermodDialogCleanQ` - данный метод выполняет очистку содержимого всех текстовых полей для ввода данных, а также устанавливает выбранным первый элемент во всех выпадающих списках в диалоге команды модификации учетной записи пользователя;

- `void passwdDialogClean()` - выполняет замену содержимого всех текстовых полей для ввода данных на пустое, а также устанавливает выбранным первый элемент во всех выпадающих списках в диалоге команды изменения пароля пользователей;

- `void addgroupDialogClean()` - данный метод выполняет очистку содержимого всех текстовых полей для ввода данных, а также устанавливает выбранным первый элемент во всех выпадающих списках в диалоге команды добавления группы пользователей;

- void gpasswdDialogClean() - выполняетзамену содержимого всех

68

текстовых полей для ввода данных на пустое, а также устанавливает выбранным первый элемент во всех выпадающих списках в диалоге команды изменения пароля пользователей;

- void chmodDialogClean() - данный метод выполняет очистку содержимого всех текстовых полей для ввода данных, а также устанавливает выбранным первый элемент во всех выпадающих списках в диалоге команды изменения прав доступа к файлу;

- void chownDialogClean() - выполняетзамену содержимого всех текстовых полей для ввода данных на пустое, а также устанавливает выбранным первый элемент во всех выпадающих списках в диалоге команды изменения владельца и группы файлов;

- void umaskDialogCleanQ - данный метод выполняет очистку содержимого всех текстовых полей для ввода данных, а также устанавливает выбранным первый элемент во всех выпадающих списках в диалоге команды получения или установки маски создания файлов;

- voidComboTooltip(JComboBox box, String[] tooltipsArray) - устанавливает для элементов заданного всплывающего списка - box - всплывающие подсказки из массива tooltipsArray. Индекс элемента в всплывающем списке соответствует индексу в массиве подсказок;

- ArrayList makeTooltipsArray(String[] StringArray) - создает специальную структуру данных - массив-список подсказок из массива строк.

- boolean mnCheck() - выполняет проверку установки службы удаленного доступа - ssh. Возвращает переменную логического типа passed, хранящую информацию о том, установлена ли в системе данная служба;

- void runCheckDaemonQ - выполняет проверку запуска службы ssh. Выводит в текстовое поле статуса соответствующее сообщение - запущена служба или нет;

- String[] getOutput(String command) - данный метод позволяет получить

выходные данные команды (параметр `command`), выполненной в отдельном процессе. Используется для получения информации по командам `users`

69

(получить текущих пользователей системы) и `groups` (получить информацию о том, в какие группы входят пользователи);

- Методы `voidadduserComponentsHide()`, `voiddeluserComponentsHide()`, `voidusermodComponentsHide()`, `voidpasswdComponentsHideQ`, `voidaddGroupComponentsHideQ`, `voidgroupmodComponentsHideQ` и `voidgpasswdComponentsHide()` выполняют одинаковую задачу, но для разных диалоговых окон: скрывают компоненты интерфейса, с которыми нет необходимости работать пользователю на начальном этапе формирования текста команды, либо на другом определенном этапе. После этого методы упаковывают свое диалоговое окно с целью уменьшения пустого рабочего места в окне.

- `voidprepUsermod(Stringtext)` - данный метод формирует содержимое окна диалога для работы с командой `usermod`. Параметр: `text` - текстовое значение, которое нужно присвоить метке (элемент интерфейса) с учетом выбранного параметра команды;

- `publicclassDoRuntime` - в данном классе расположены методы по проверки установки и запуска демонов (служб операционной системы Linux);

- `String install Verify(String daemonName)` - выполняет проверку установки службы по его имени (параметр `daemonName`). Для этого создается скрипт-файл в определенном формате, затем он выполняется в отдельном процессе. Метод получает результаты работы скрипта и по ним устанавливает - установлена ли в системе служба с данным именем. Метод возвращает строку `result` с соответствующим сообщением;

- `boolean isRunning()` - данный метод выполняет проверку запуска службы. В отдельном процессе выполняется уже заготовленный скрипт- файл, результаты работы которого помещаются в файл. По содержимому файла метод определяет, запущена ли данная служба;

•voidmakeProcSh(StringdaemonName) - создает и формирует sh-файл для проверки запуска службы операционной системы Linux по ее имени (параметр daemonName).

2.4.Средства и порядок испытаний

Объект испытаний. Объект испытаний - программный модуль оптимизации работы сети на платформе операционной системы Linux по протоколам управления.

Функции программного модуля:

- регистрация, управления пользователями и группами пользователей;
- управления доступом к файлам и каталогам как на локальном компьютере, так и на удаленном компьютере внутри локальной сети;
- предоставление работы удаленного доступа.

Цель испытаний. Целью проведения испытаний является подтверждение работоспособности программного модуля для заданных ситуаций взаимодействия с пользователями.

Конечной целью проектирования - создание программного модуля оптимизации работы сети и сетевого администратора, связанная с улучшением работы по администрированию пользователей и файловых ресурсов внутри сети.

Средства и порядок испытаний. Для испытаний использовался IBM PC-совместимый компьютер на базе процессора Pentium 4 2133 Mhz с 512MB SDRAM под управлением ОС LinuxDebian. В системе установлены: JAVA платформа и среда визуального проектирования NetBeansIDE 6.7 [12].

Запуск программы осуществляется файлом Diploma.jar.

Методы испытаний. Рассмотрим основные ситуации, при которых программа выдаст сообщение об ошибке.

1. Проведем попытку добавить пользователя, который уже существует, т.е. с таким же именем. Это некорректная ситуация. В файле с перечнем пользовательских учетных записей имеется пользователь с именем

«tester».

```
Файл Правка Вид Терминал Вкладки Справка
GNU nano 2.0.7 Файл: /etc/passwd

libuuid:x:100:101::/var/lib/libuuid:/bin/sh Debian-eximrx:101:105::/var/spool/exim4:/bin/false
statd:x:102:65534::/var/lib/nfs:/bin/false messagebus:x:103:108::/var/run/dbus:/bin/false
avahi:x:104:109:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false
gdm:x:105:112:Gnome Display Manager:/var/lib/gdm:/bin/false
haldaemon:x:106:114:Hardware abstraction layer,,,:/var/run/hal:/bin/false
hplip:x:107:7:HPLIP system user,,,:/var/run/hplip:/bin/false
alim:x:1000:1000:alim,,,:/home/alim:/bin/bash
peta:x:110:65534::/home/peta:/bin/false
qwerty:x:1001:1005::/home/qwerty:/bin/bash
sshd:x:35:35:sshd:./bin/sh
tester:x:1002:1007::/home/tester:/bin/bash
```

Рисунок 2.36. Фрагмент файла с перечнем пользовательских учетных записей

Добавление пользователя с именем «tester».

При нажатии на кнопку «adduser» переходим в диалоговое окно, в котором задаем имя нового пользователя и нажимаем кнопку «Ок».

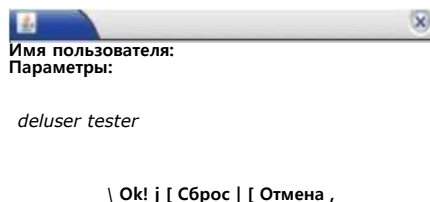


Рисунок 2.37. Окно добавления пользователя

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого программа сообщит следующее:

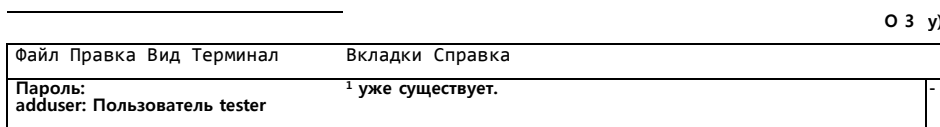


Рисунок 2.38. Окно терминала добавления пользователя

2. Проведем попытку удалить пользователя, который не существует.

При нажатии на кнопку «deluser» переходим в диалоговое окно, в

котором задаем имя пользователя, которого хотим удалить и нажимаем кнопку «Ок».

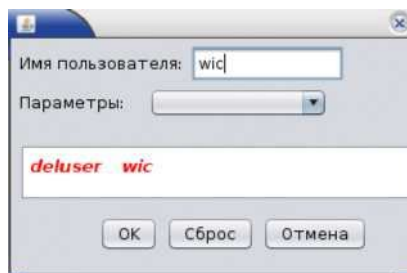


Рисунок 2.39. Окно удаления пользователя

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого программа сообщит следующее:

Файл	Правка	Вид	Терминал	Вкладки	Справка
Пароль:			Пользователь wie'		не существует.
/usr/sbin/deluser:					

Рисунок 2.40. Окно терминала удаления пользователя

3. Проведем попытку добавить группу, которая уже существует, т.е. с таким же именем. Это некорректная ситуация. В файле с перечнем групп имеется группа с именем «tester».

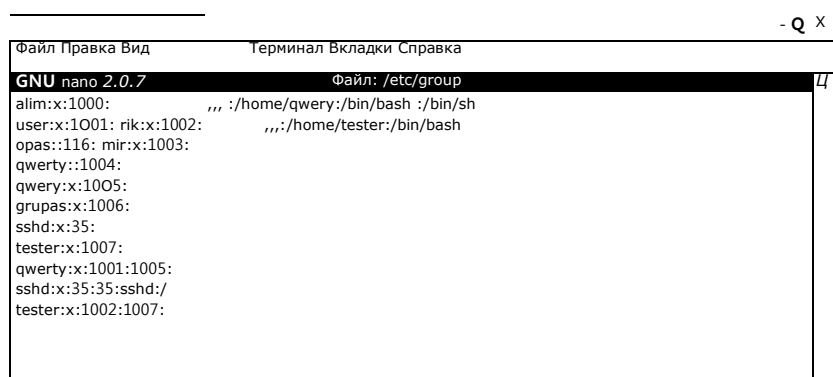


Рисунок 2.41. Фрагмент файла с перечнем групп

Добавление группы с именем «tester».

При нажатии на кнопку «addgroup» переходим в диалоговое окно, в

котором задаем имя новой группы и нажимаем кнопку «Ок».

Имя группы:

Параметры:

☐ Системная

`addgroup tester`

Рисунок 2.41. Окно добавления группы

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого программа сообщит следующее:

Файл	Правка	Вид	Терминал	Вкладки	Справка
Пароль:				существует.	
addgroup: Группа 'tester' уже				М	

Рисунок 2.42. Окно терминала добавления группы

4. Проведем попытку удалить группу, которой не существует.

При нажатии на кнопку «delgroup» переходим в диалоговое окно, в котором задаем имя группы, которую хотим удалить и нажимаем кнопку «Ок».

Имя группы:

Параметры:

`delgroup wic`

Рисунок 2.43.
Окно удаления группы

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого программа сообщит следующее:

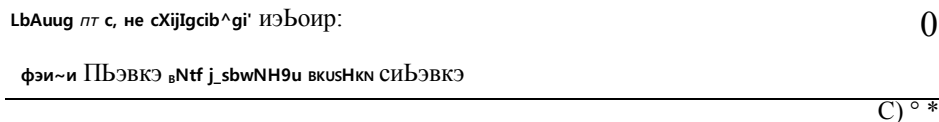


Рисунок 2.44. Окно терминала удаления группы

5. Проведем попытку изменить учетную запись пользователя, т.е. например для пользователя «tester» хотим изменить id (идентификатор пользователя) на 1001, но такой id уже занят другим пользователем.

При нажатии на кнопку «usermod»перехофlHivi в диалоговое окно, в котором задаем имя пользователя, указываем новый id и нажимаем кнопку «Ок».



Рисунок 2.45. Окно изменения информации о пользователе

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого программа сообщит следующее:

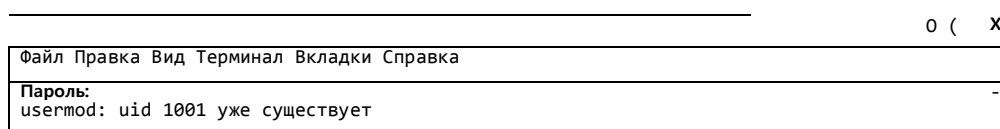
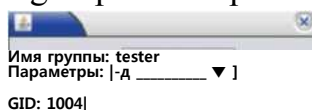


Рисунок 2.46. Окно терминала изменения информации о пользователе

6. Проведем попытку изменить группу, т.е. например для группы «tester» хотим изменить id (идентификатор группы) на 1004, но такой id уже занят другой группой.

При нажатии на кнопку «groupinod»перехофlHIM в диалоговое окно, в



котором задаем имя группы, указываем новый `id` и нажимаем кнопку «Ок».

[| Сброс | | Отмена |

Рисунок 2.47. Окно изменения информации о группе

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого программа сообщит следующее:



Рисунок 2.48 - Окно терминала изменения информации о группе

7. Проведем попытку удалить пароль пользователя, которого не существует. При нажатии на кнопку «passwd» перейдем в диалоговое окно, в котором задаем имя пользователя, указываем параметр `d` (удалить пароль пользователя) и нажимаем кнопку «Ок».

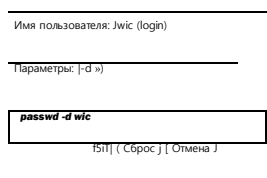


Рисунок 2.49. Окно удаления пароля пользователя

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого программа сообщит следующее:

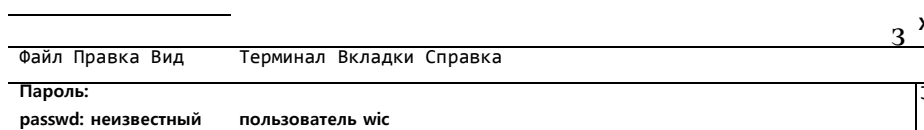


Рисунок 2.50. Окно терминала удаления пароля пользователя

8. Проведем попытку удалить существующего пользователя из

не существующей группы.

При нажатии на кнопку «grasswd» переходим в диалоговое окно, в котором задаем имя группы, имя пользователя и указываем параметр d (удалить пользователя из группы) и нажимаем кнопку «Ок».

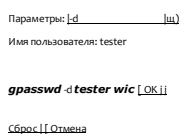


Рисунок 2.51. Окно удаления пользователя из группы

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого программа сообщит следующее:

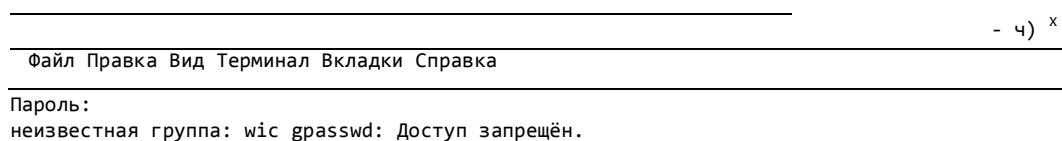


Рисунок 2.52. Окно терминала удаления пользователя из группы

9. Проведем попытку изменить владельца и группу для файла «15.jpg», указав не существующую группу.

При нажатии на кнопку «chown» переходим в диалоговое окно, в котором задаем имя пользователя, имя группы и указываем необходимый

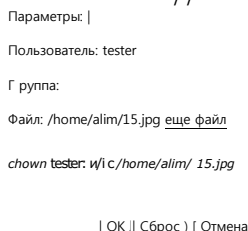


Рисунок 2.53. Окно изменения владельца и группы для файла

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого программа сообщит

файл и нажимаем кнопку «Ок».

следующее:

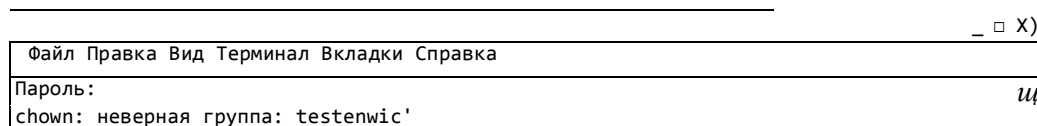


Рисунок 2.54. Окно терминала изменения владельца и группы для файла

10. Проведем попытку изменить права доступа к файлу «15.jpg», не заполнив поле «Права». При нажатии на кнопку «СИтосьпереходим в диалоговое окно, в котором указываем путь к файлу, выбираем для каких пользователей (в данном случае all — для всех) и оставляем пустым поле «Права».

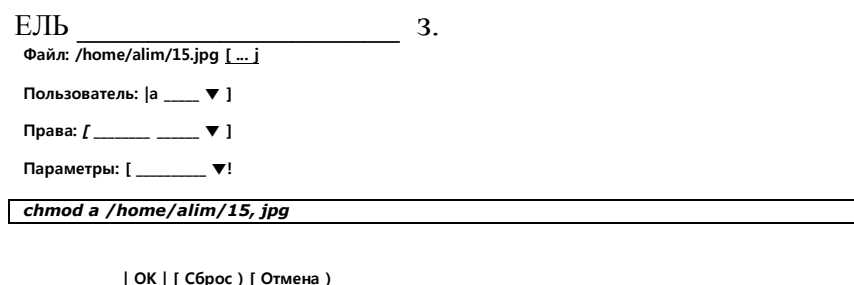


Рисунок 2.55. Окно изменения прав доступа к файлу

78

Затем осуществится переход в системный терминал, в котором потребуется ввести пароль супер пользователя. После этого программа сообщит следующее:

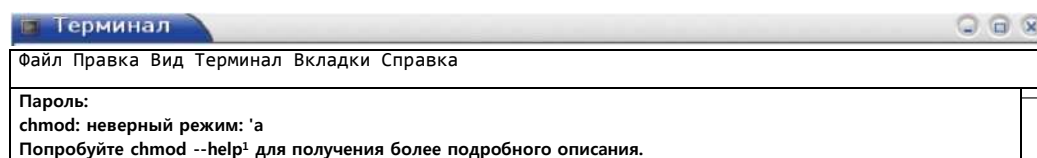


Рисунок 2.56. Окно терминала изменения прав доступа к файлу

11 .Проведем попытку запустить службу удаленного доступа (ssh), введя не корректные настройки в конфигурационном файле sshd config. При нажатии на кнопку «Запустить» осуществится переход в системный терминал, в котором

потребуется ввести пароль супер пользователя. После этого программа сообщит следующее:

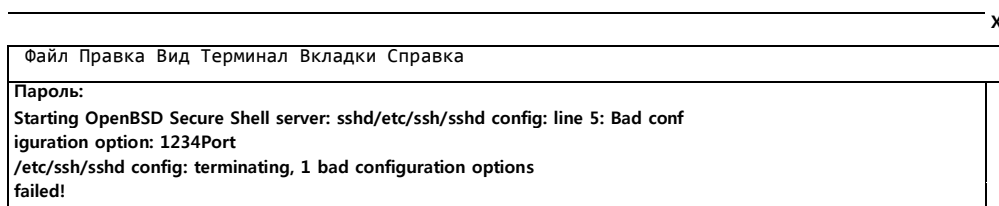


Рисунок 2.57. Окно терминала запуска службы ssh

Анализ результатов тестирования. В результате проведенных испытаний можно сказать о следующем:

- программа корректно работает для всех задач изменения в системе;
- программа выдает сообщения о неверно заданных параметрах;
- автоматический анализ конфигурации проходит корректно.

3. БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ

3.1. Местное освещение на рабочем месте оператора ПК

При освещении производственных помещений используют естественное освещение, создаваемое прямыми солнечными лучами и рассеянным светом небосвода и меняющемся в зависимости от географической широты, времени года и суток, степени облачности и прозрачности атмосферы; искусственное освещение, создаваемое электрическими источниками света, и совмещенное освещение, при котором недостаточное по нормам естественное освещение дополняют искусственным. Конструктивно естественное освещение подразделяют на боковое (одно- и двухстороннее), осуществляемое через световые проемы в наружных стенах; верхнее - через аэрационные и зенитные фонари, проемы в кровле и перекрытиях; комбинированное - сочетание верхнего и бокового освещения. Искусственное освещение по конструктивному исполнению может быть двух видов - общее и комбинированное. Систему общего освещения применяют в помещениях, где по всей площади выполняются однотипные работы (литейные, сварочные, гальванические цехи),

а также в административных, конторских и складских помещениях. Различают общее равномерное освещение (световой поток распределяется равномерно по всей площади без учета расположения рабочих мест) и общее локализованное освещение (с учетом расположения рабочих мест). При выполнении точных зрительных работ (например, слесарных, токарных, контрольных) в местах, где оборудование создает глубокие, резкие тени или рабочие поверхности расположены вертикально (штампы, гильотинные ножницы), наряду с общим освещением применяют местное. Совокупность местного и общего освещения называют комбинированным освещением. Применение одного местного освещения внутри производственных помещений не допускается,

поскольку образуются резкие тени, зрение быстро утомляется и создается опасность производственного травматизма. По функциональному назначению искусственное освещение подразделяют на рабочее, аварийное и специальное, которое может быть охранным, дежурным, эвакуационным, эритемным, бактерицидным и др. Рабочее освещение предназначено для обеспечения нормального выполнения производственного процесса, прохода людей, движения транспорта и является обязательным для всех производственных помещений.

Аварийное освещение устраивают для продолжения работы в тех случаях, когда внезапное отключение рабочего освещения (при авариях) и связанное с этим нарушение нормального обслуживания оборудования могут вызвать взрыв, пожар, отравление людей, нарушение технологического процесса и т.д. Минимальная освещенность рабочих поверхностей при аварийном освещении должна составлять 5% нормируемой освещенности рабочего освещения, но не менее 2 лк.

Эвакуационное освещение предназначено для обеспечения эвакуации людей из производственного помещения при авариях и отключении рабочего освещения; организуется в местах, опасных для прохода людей: на лестничных клетках, вдоль основных проходов производственных помещений, в которых работают более 50 чел. Минимальная освещенность на полу основных проходов и на ступеньках при эвакуационном освещении должна быть не менее 0,5лк, на открытых территориях - не менее 0,2лк. Охранное освещение устраивают вдоль границ территорий, охраняемых специальным персоналом. Наименьшая освещенность в ночное время 0,5лк. Сигнальное освещение применяют для фиксации границ опасных зон; оно указывает на наличие опасности, либо на безопасный путь эвакуации.

Условно к производственному освещению относят бактерицидное и эритемное облучение помещений.

Основные требования к производственному освещению. Основной задачей производственного освещения является поддержание на рабочем

месте освещенности, соответствующей характеру зрительной работы. Увеличение освещенности рабочей поверхности улучшает видимость объектов за счет повышения их яркости, увеличивает скорость различения деталей, что сказывается на росте производительности труда. Так, при выполнении отдельных операций на главном конвейере сборки автомобилей при повышении освещенности с 30 до 75лк производительность труда повысилась на 8%. При дальнейшем повышении до 100 лк - на 28 % (по данным проф. А. Л. Тарханова). Дальнейшее повышение освещенности не дает роста производительности. При организации производственного освещения необходимо обеспечить равномерное распределение яркости на рабочей поверхности и окружающих предметах. Перевод взгляда с ярко освещенной на слабо освещенную поверхность вынуждает глаз переадаптироваться, что ведет к утомлению зрения и соответственно к снижению производительности труда. Для повышения равномерности естественного освещения больших цехов осуществляется комбинированное освещение. Светлая окраска потолка, стен и оборудования способствует равномерному распределению яркостей в поле зрения работающего.

Производственное освещение должно обеспечивать отсутствие в поле зрения работающего резких теней. Наличие резких теней искажает размеры и формы объектов, их различение, и тем самым повышает утомляемость, снижает производительность труда. Особенно вредны движущиеся тени, которые могут привести к травмам. Тени необходимо смягчать, применяя, например, светильники со светорассеивающими молочными стеклами, при естественном освещении, используя солнцезащитные устройства (жалюзи, козырьки и др.). Для улучшения видимости объектов в поле зрения работающего должна отсутствовать прямая и отраженная блескость. Блескость - это повышенная яркость светящихся поверхностей, вызывающая нарушение зрительных функций (ослепленность), т.е. ухудшение видимости объектов. Блескость ограничивают уменьшением яркости источника света, правильным выбором защитного угла светильника, увеличением высоты

подвеса светильников, правильным направлением светового потока на рабочую поверхность, а также изменением угла наклона рабочей поверхности. Там, где это возможно, блестящие поверхности следует заменять матовыми. Колебания освещенности на рабочем месте, вызванные, например, резким изменением напряжения в сети, обуславливают пере адаптацию глаза, приводя к значительному утомлению. Постоянство освещенности во времени достигается стабилизацией плавающего напряжения, жестким креплением светильников, применением специальных схем включения газоразрядных ламп. При организации производственного освещения следует выбирать необходимый спектральный состав светового потока. Это требование особенно существенно для обеспечения правильной цветопередачи, а в отдельных случаях для усиления цветовых контрастов. Оптимальный спектральный состав обеспечивает естественное освещение. Для создания правильной цветопередачи применяют монохроматический свет, усиливающий одни цвета и ослабляющий другие. Осветительные установки должны быть удобны и просты в эксплуатации, долговечны, отвечать требованиям эстетики, электробезопасности, а также не должны быть причиной возникновения взрыва или пожара. Обеспечение указанных требований достигается применением защитного за нуления или заземления, ограничением напряжения питания переносных и местных светильников, защитой элементов осветительных сетей от механических повреждений и т.п.

Источники света, применяемые для искусственного освещения, делят на две группы - газоразрядные лампы и лампы накаливания. Лампы накаливания относятся к источникам света теплового излучения. Видимое излучение в них получается в результате нагрева электрическим током вольфрамовой нити. В газоразрядных лампах излучение оптического диапазона спектра возникает в результате электрического разряда в атмосфере инертных газов и паров металлов, а также за счет явлений люминесценции, которое невидимое ультрафиолетовое излучение преобразует в видимый свет.

Благодаря удобству в эксплуатации, простоте в изготовлении, низкой

инерционности при включении, отсутствии дополнительных пусковых устройств, надежности работы при колебаниях напряжения и при различных метеорологических условиях окружающей среды лампы накаливания находят широкое применение в промышленности. Наряду с отмеченными преимуществами лампы накаливания имеют и существенные недостатки: низкая световая отдача (для ламп общего назначения $\eta = 7 \dots 20$ лм/Вт), сравнительно малый срок службы (до 2,5 тыс. ч), в спектре преобладают желтые и красные лучи, что сильно отличает их спектральный состав от солнечного света. В последние годы все большее распространение получают галогеновые лампы - лампы накаливания с йодным циклом. Наличие в колбе паров йода позволяет повысить температуру накала нити, т.е. световую отдачу лампы (до 40 лм/Вт). Пары вольфрама, испаряющиеся с нити накаливания, соединяются с йодом и вновь оседают на вольфрамовую спираль, препятствуя распылению вольфрамовой нити и увеличивая срок службы лампы до 3 тыс. ч. Спектр излучения галогеновой лампы более близок к естественному. Основным преимуществом газоразрядных ламп перед лампами накаливания является большая световая отдача 40... 110 лм/Вт. Они имеют значительно больший срок службы, который у некоторых типов ламп достигает 8... 12 тыс. ч. От газоразрядных ламп можно получить световой поток любого желаемого спектра, подбирая соответствующим образом инертные газы, пары металлов, люминофоров. По спектральному составу видимого света различают лампы дневного света (ЛД), дневного света с улучшенной цветопередачей (ЛЛД), холодного белого (ЛХБ), теплого белого (ЛТБ) и белого цвета (ЛБ). Основным недостатком газоразрядных ламп является пульсация светового потока, что может привести к появлению стробоскопического эффекта, заключающегося в искажении зрительного восприятия. При кратности или совпадении частоты пульсации источника света и обрабатываемых изделий вместо одного предмета видны изображения нескольких, искажается направление и скорость движения, что

делает невозможным выполнение производственных операций и ведет к

увеличению опасности травматизма.

Электрический светильник - это совокупность источника света и осветительной арматуры, предназначенной для перераспределения излучаемого источником светового потока в требуемом направлении, предохранения глаз рабочего от слепящего действия ярких элементов источника света, защиты источника от механических повреждений, воздействия окружающей среды и эстетического оформления помещения.

3.2. Обеспечение пожаробезопасности

Пожар - неконтролируемое горение вне специального очага, наносящее материальный ущерб. Возможности создания условий для возникновения пожара или его быстрого развития представляют собой пожарную опасность.

Опасными поражающими факторами пожара являются:

- открытый огонь и искры;
- выделяющееся при горении тепло вызывает повышение температуры окружающей среды, и когда она доходит до критической для окружающих очаг пожара предметов и вещей, загораются и они. Очаг пожара разрастается;
- токсичные продукты горения, дым;
- падающие части строительных конструкций и агрегатов.

Основными факторами взрыва являются:

- воздушная взрывная волна, основным параметром которой является избыточное давление в ее фронте;
- осколочные поля, создаваемые летящими обломками взрывающихся объектов, поражающее действие которых определяется количеством летящих осколков, их кинетической энергией и радиусом разлета.

Согласно стандартам производственное здание относится к категории Г (производства, связанные с обработкой негорючих веществ и материалов в горячем, раскаленном или расплавленном состоянии, сопровождающееся выделением лучистого тепла, искр и пламени). Использование в компьютерах

мощных радиоэлементов: понижающие высоковольтные трансформаторы, выпрямляющие диоды, транзисторы, микросхемы, токопроводящие линии работающих при относительно больших токах и напряжениях, предполагает их нагрев до высоких температур. Тепловое действие электрического тока, проходящего по проводим, при неисправностях или перегрузка электроустановок или аппаратуры может быть причиной пожара.

Основными причинами пожара являются:

- повреждение изоляции проводов;
- попадание на неизолированные провода токопроводящих предметов;
- воздействие на провода химически активных веществ, паров;
- неправильный монтаж установки прибора;

Защита электрооборудования осуществляется при помощи плавких предохранителей и специальных автоматов, включаемых последовательно в электрическую цепь. При конструировании аппаратуры, где возможно образование искры или электродуги, предусмотрены кожухи или дугогасящие решетки.

На производстве применяются методы и средства пожаротушения: применения углекислотных огнетушителей, так как СО₂ не портит оборудование и не проводит электрический ток.

В случае пожара имеются эвакуационные пути:

- из помещений первого этажа - наружу непосредственно или через коридор, вестибюль, лестничную клетку;
- из помещений любого этажа - в коридор или проход, ведущий к лестничной клетке, или на лестничную клетку, имеющую выход наружу, отделенной от примыкающих коридоров перегородками с дверьми.

86

Заключение

В выпускной квалификационной работе была разработана программная система оптимизации работы сети, модуль протоколов управления, ориентированная на использование в небольших локальных сетях и в

университетских лабораториях компьютерных сетей для проведения практических занятий. В процессе выполнения работы были получены следующие результаты:

1. Исследована область применения систем такого класса и актуальность их внедрения.
2. Проанализированы литературные источники по теме.
3. Проведен системотехнический анализ задачи, в результате которого были выявлены принципы оптимального разбиения на подзадачи для оптимизации процесса разработки и улучшения ее качества.
4. Проведен вариантный анализ и выбора протокола, предоставляющий удаленный доступ по сети.
5. Формально описана разработанная подсистема - модуль протоколов управления.
6. Разработана и воплощена методика тестирования подсистемы.

Программная система рекомендуется к применению в учебных заведениях, в качестве лабораторного практикума для изучения основных особенностей системного администрирования, а также в малых и средних предприятиях, использующих системное администрирование компьютерной сети в различных отраслях и сферах деятельности.

Список литературы

1. Каримов И.А.: Постановление Президента Республики Узбекистан «О мерах по дальнейшему внедрению развитию современных информационно-коммуникационных технологий» / Собрание законодательства Республики Узбекистан, 2012 г., № 13, стр. 139. Ташкент, 2012.
2. Методические указания «Процедура системного анализа при проектировании программных систем» для студентов-дипломников дневной и заочной формы обучения специальности 7.091501 / Сост. Сергеев Г.Г., Скатков А.В., Мащенко Е.Н. - Севастополь каф. КВТ СевНТУ, 2005.-32с.
3. Стахнов А.А. Сеть для офиса и Linux-сервер своими руками. - СПб.: БХВ-Петербург, 2006. - 320с.:ил.

4. Эви Немет. Руководство администратора Linux.: Пер.с англ. - М.: Издательский дом «Вильямс», 2005. - 880с.: ил. - Парал. тит.англ.
5. Браун П. Введение в операционную систему UNIX. -М.: Мир, 1987. -287 с.
6. Тихомиров В.П., Давидов М.И. Операционная система UNIX: Инструментальные средства программирования. -М.: Финансы и статистика, 1988. -206 с.
7. Керниган Б.В., Пайк Р. UNIX - Универсальная среда программирования. - М.: Финансы и статистика, 1992 -304 с.
8. Т. Адельштайн, Б. Любанович. Системное администрирование в Linux.: Пер.с англ. - М.: Издательский дом «Питер», 2010. - 288с.: ил. - Парал. тит.англ.
9. Алексей Береснев. Администрирование GNU/Linux с нуля. - СПб.: БХВ-Петербург, 2010. - 576с.:ил.
10. В.М.Вишне夫斯基. Теоретические основы проектирования компьютерных сетей Москва: Техносфера, 2003. — 512с.
88
11. Ното П. JAVA: Справ, руководство: Пер. с англ./ Под ред. А.Тихонова. М.: БИНОМ, 1996. -447с.
12. Патрик Нотон, Герберт Шилдт Полный справочник по Java.- McGraw- Hill, 1997, Издательство "Диалектика", 1997.
13. Чен М.С. и др.Программирование на JAVA: 1001 совет:Наиболее полное руководство по Java и Visual J++:Пер.с англ./Чен М.С.,Грифис С.В.,Изи Э.Ф.-Минск: Попурри, 1997.-640с. ил.
14. И.Ю.Баженова Язык программирования Java.- АО "Диалог-МИФИ", 1997.
Кен Арнольд, Джеймс Гослинг Язык программирования Java.- Addison-Wesley Longman,U.S.A.,1996, Издательство "Питер-Пресс", 1997.
15. Методические указания к выполнению курсовой работы по дисциплинам «Основы менеджмента и маркетинга», «Менеджмент» для студентов специальностей 7.091501, 7.091401, 7.080401 всех форм обучения / Сост. Г.А. Раздобреева, Е.В. Коваль, Т.В. Кулешова, С.В. Ключко - Севастополь: Изд-во СевНТУ, 2009. - 24с.

16. Экология и безопасность жизнедеятельности: Учебное пособие для студентов ВУЗов/ ред. Л. А. Муравий, 2002.
17. Белов С.В. Безопасность жизнедеятельности М.: Высшая школа. 2003.
18. Ёрматов Ф.Е., Исамухамедов Ё.У. Мех,натни мухофаза килиттт. Дарслик. Узбекистан нашриёти. Тошкент 2002.
19. <http://www.pcweek.ru/foss/article/detail.php?ID=136556>.