Установка и настройка ОС «СИНЕРГИЯ»

АННОТАЦИЯ

В данном документе приводится техническая информация о процессе установки и первоначальной настройки операционной системы (ОС) Синергия. Дальнейшая работа с ОС осуществляется в соответствии с эксплуатационной документацией.

Данный документ предназначен для группы администраторов системы.

СОДЕРЖАНИЕ

1. АДМИНИСТРИРОВАНИЕ

Административное управление в ОС отделено от общего доступа пользователей.

ОС позволяет администратору (или суперпользователю root) выполнять над файлом или процессом любую операцию. Кроме того, некоторые системные вызовы (обращения к ядру) может выполнять только суперпользователь. Некоторые системные вызовы доступны всем пользователям, но имеют специальные опции для суперпользователя root. Суперпользователь может назначить определённым пользователям права для выполнения действий по настройке ОС, требующих привилегий суперпользователя root (через механизм `sudo`). Далее по тексту такой пользователь именуется администратором.

Примеры операций, которые может выполнить только суперпользователь root:

- монтирование и размонтирование ФС;
- изменение корневого каталога процесса командой chroot;
- создание файлов устройств;
- установка системных часов;
- изменение принадлежности файлов;
- задание host-имени системы;
- конфигурирование сетевых интерфейсов.

Администратор может выполнять указанные операции через механизм `sudo`. Пароль рекомендуется создавать способом, максимально затрудняющим его подбор. Наиболее безопасный пароль состоит из случайной (псевдослучайной) последовательности букв, знаков препинания, специальных символов и цифр.

ВНИМАНИЕ. Действия по администрированию ОС необходимо выполнять в мандатном контексте безопасности субъекта с нулевым уровнем и пустым набором категорий.

2. УСТАНОВКА И НАСТРОЙКА ОС

2.1. Общие положения

Дистрибутив ОС содержит все необходимые файлы для выполнения процесса ее установки в необходимом варианте установки на жесткий диск целевого компьютера, имеющего устройство чтения данного типа машинного носителя.

2.2. Установка с машинного носителя (запуск программы установки)

Для начала установки необходимо настроить BIOS для загрузки с соответствующего устройства. Обратитесь к документации на ваш компьютер за руководством по настройке порядка загрузки.

После инициализации BIOS и старта загрузки с необходимого машинного носителя, на экране будет отображено окно выбора действия (рис.1):

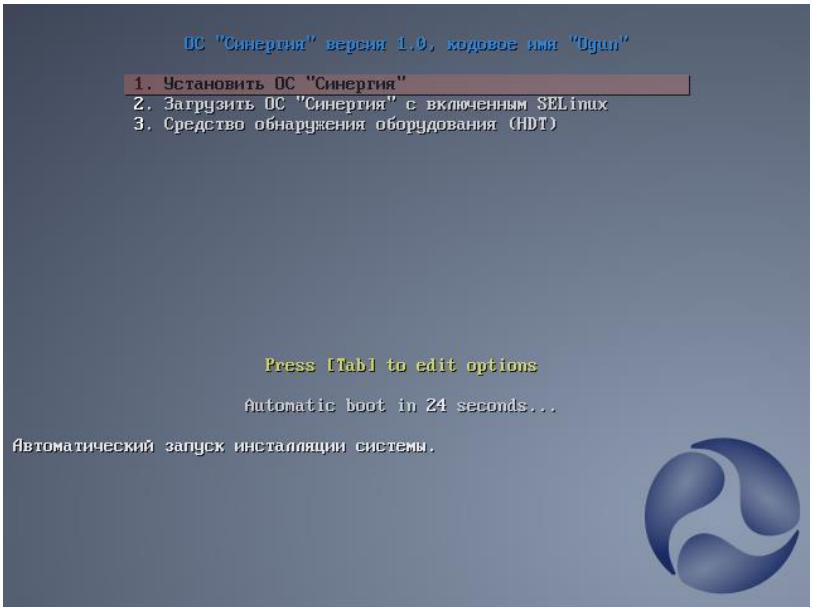

Рисунок 1. Меню выбора действия

Выберите вариант «1» для старта установки ОС «Синергия».

После этого выберите диск для установки ОС и подтвердите выбор нажатием кнопки «ОК» (рис.2, 3):

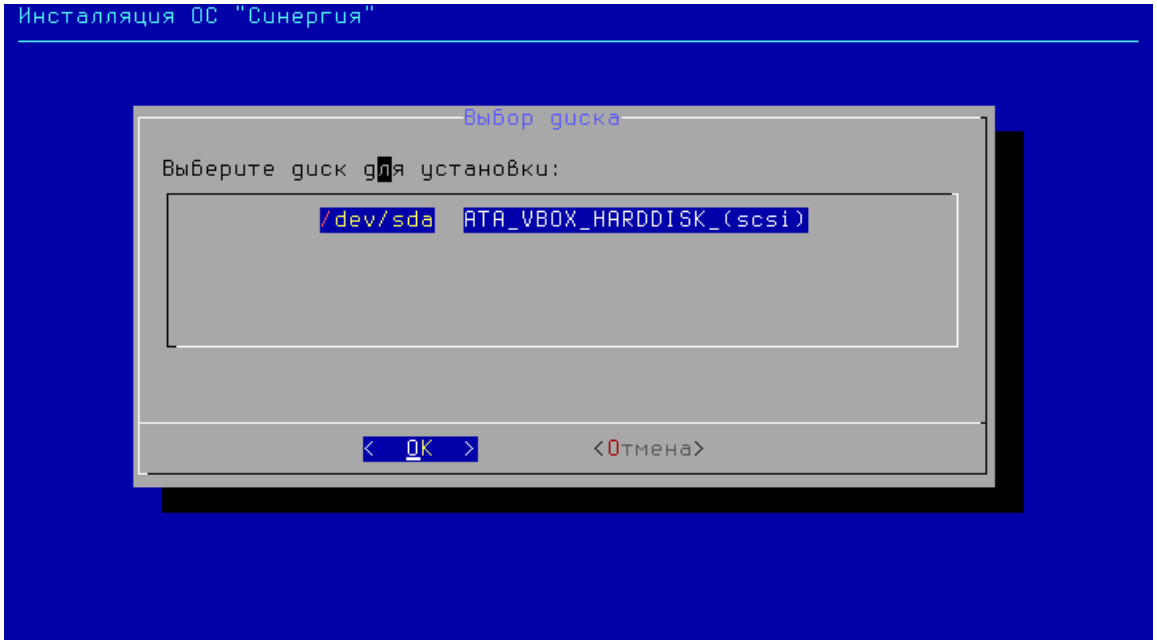

Рисунок 2. Меню выбора диска для установки

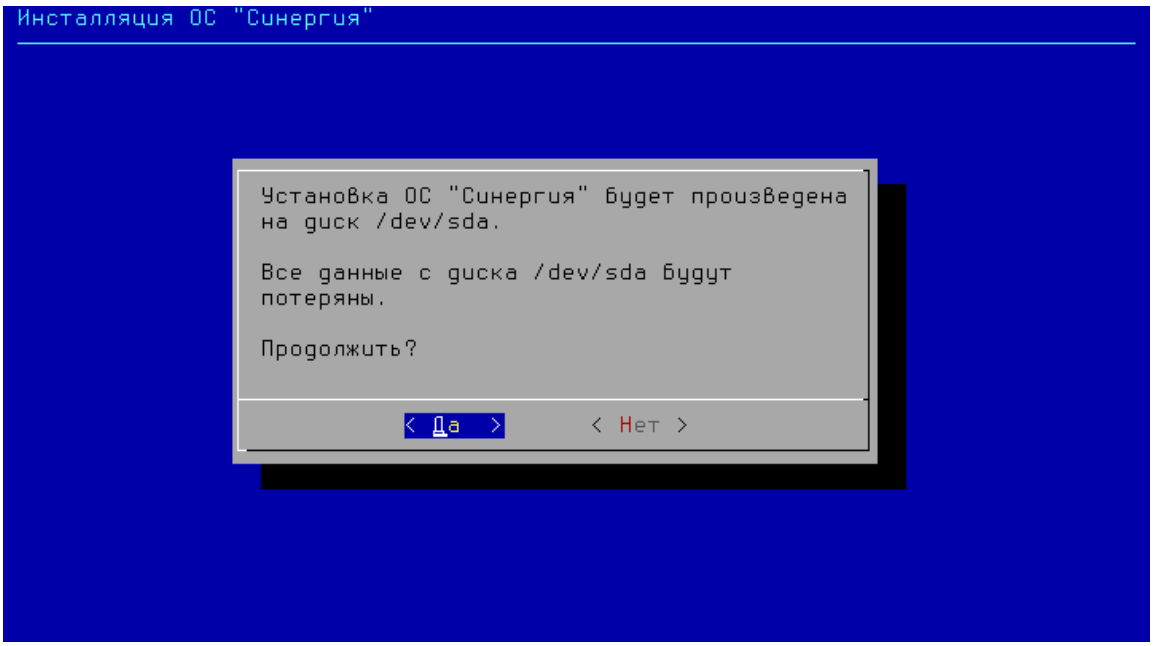

Рисунок 3. Диалог подтверждения выбора диска

Далее, установщик запросит ввести пароль для пользователя admin (рис.4):

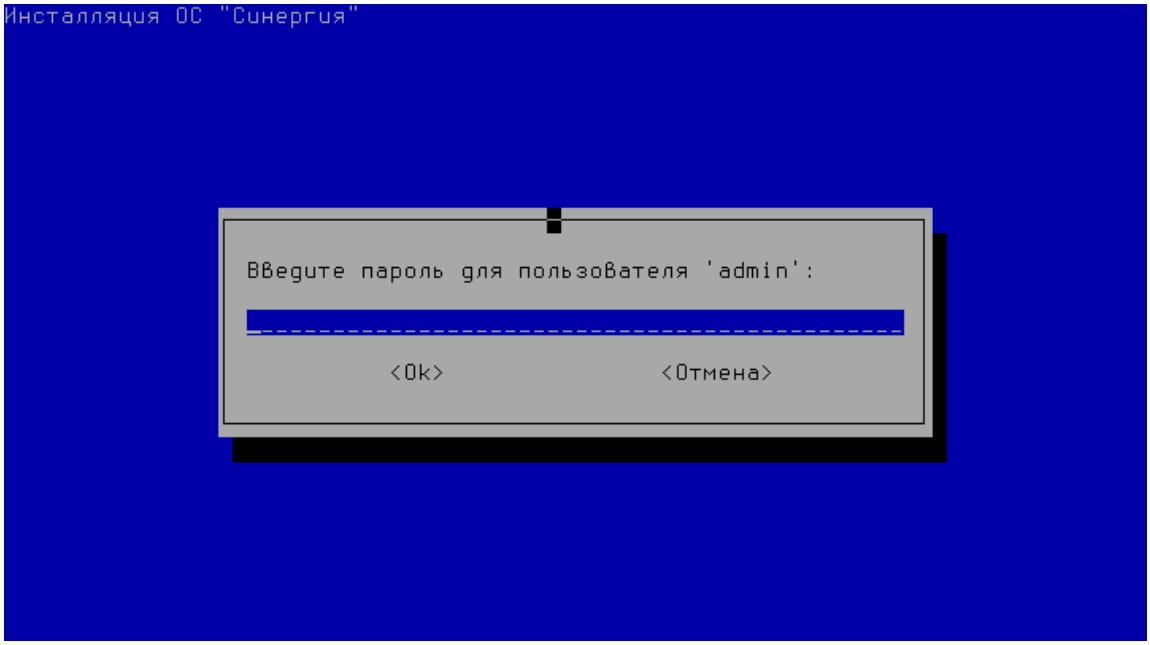

Рисунок 4. Диалог ввода пароля администратора

После ввода пароля, установщик начнёт процедуру установки ОС на ЭВМ. После завершения первого этапа установки инсталлятор покажет окно в соответствии с рисунком 5:

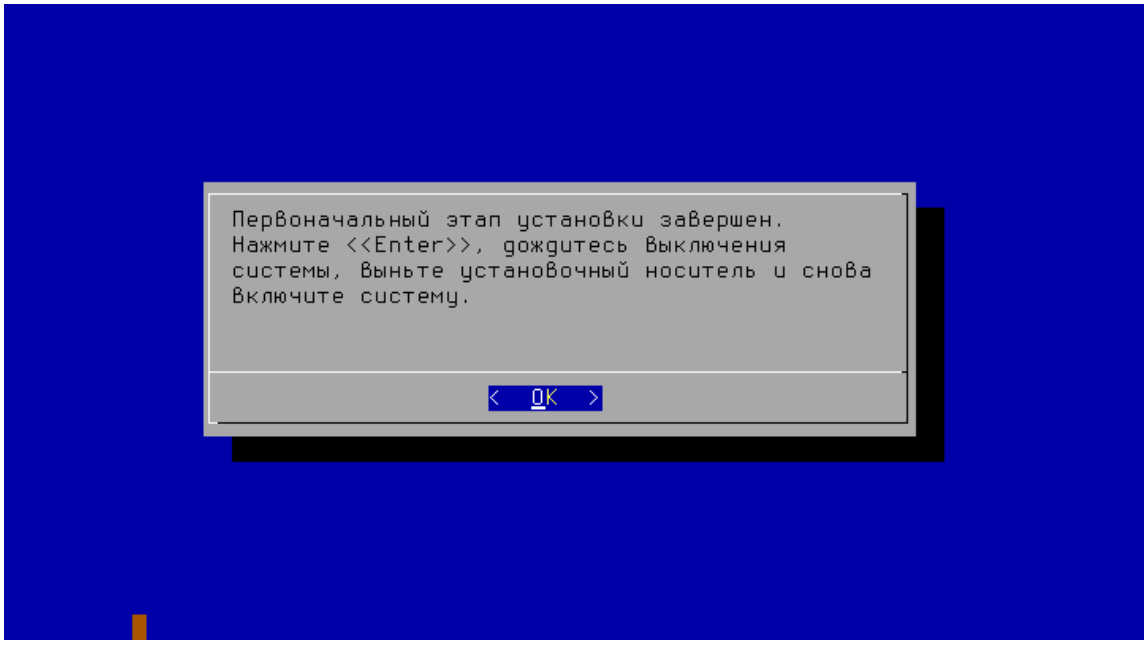

Рисунок 5. Сообщение о завершении первого этапа установки

Далее после нажатия на кнопку «ОК» система выполнит перезагрузку ЭВМ, после которой на экране должно появиться стандартное меню загрузчика (рис. 6):

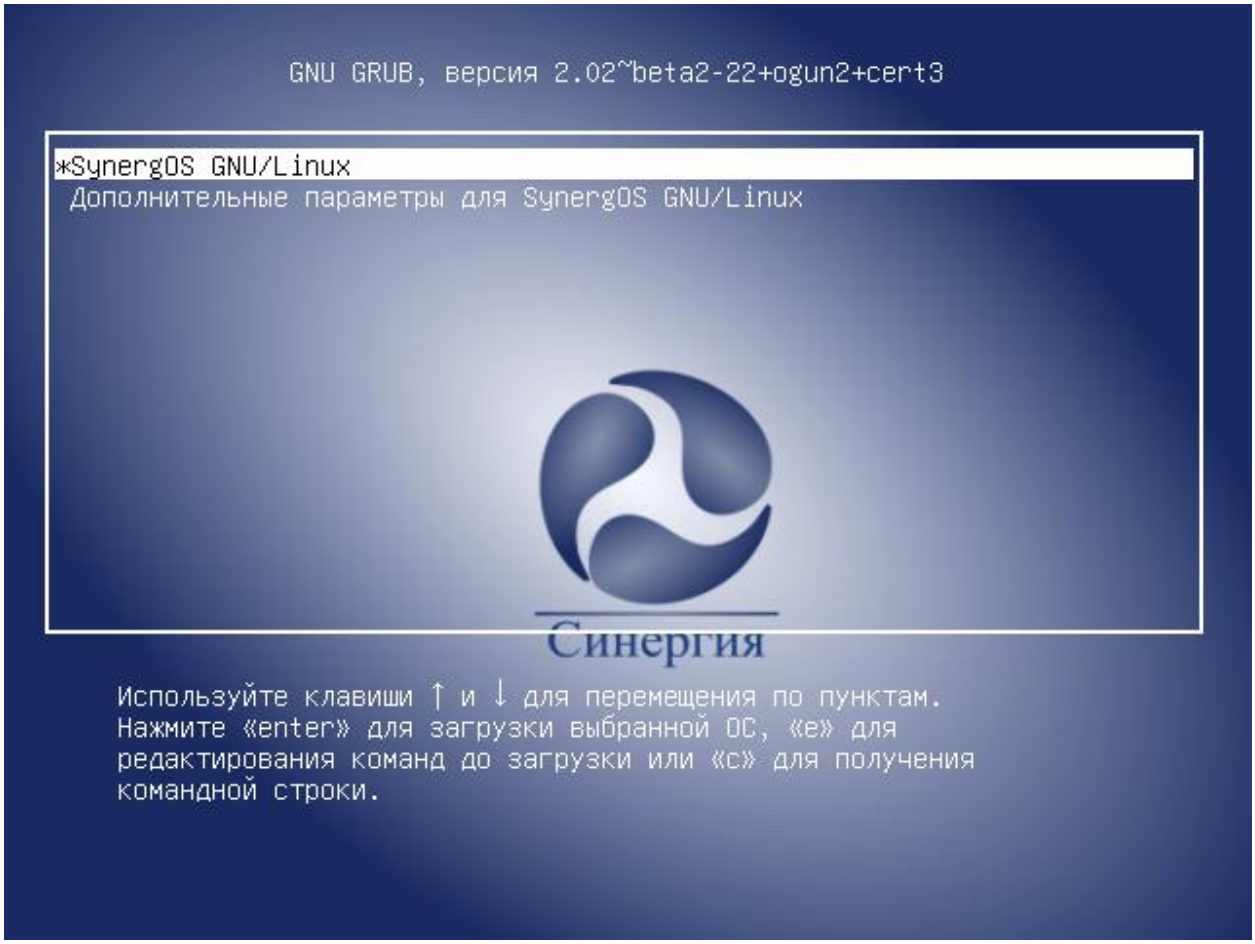

Рисунок 6. Меню загрузчика ОС

Во время первой загрузки операционной системы будут выполнены инициализационные сценарии SELinux (рис. 7), после завершения которых операционная система вновь выполнит перезагрузку.

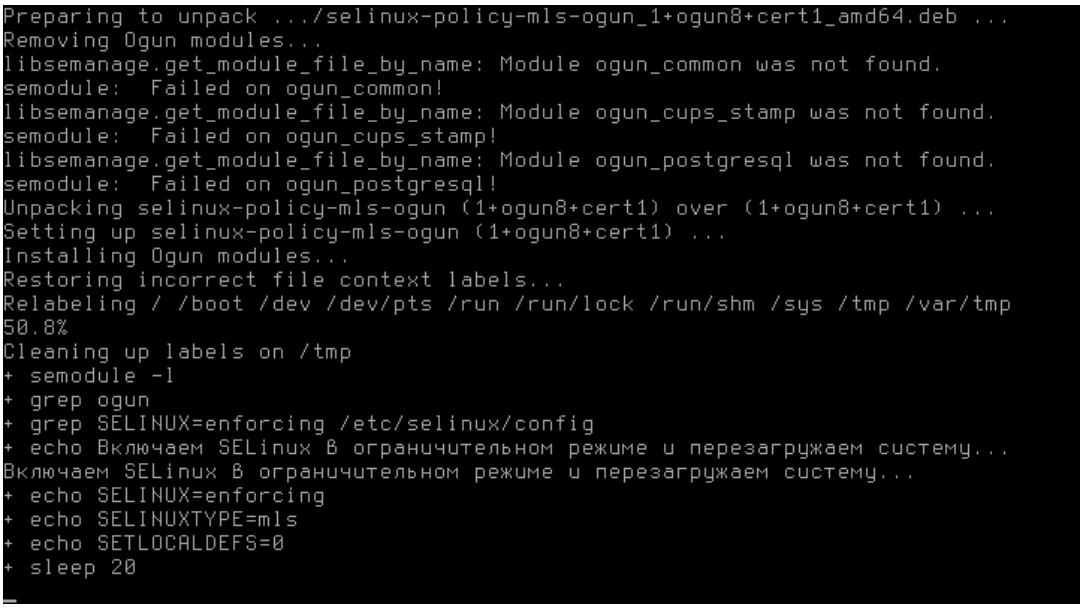

Рисунок 7. Вывод при первой загрузке

После второй перезагрузки, в случае успешной установки, операционная система является установленной на ЭВМ и будет выполнена штатная загрузка ОС, в результате которой будет запущен экран входа в систему (рис.8).

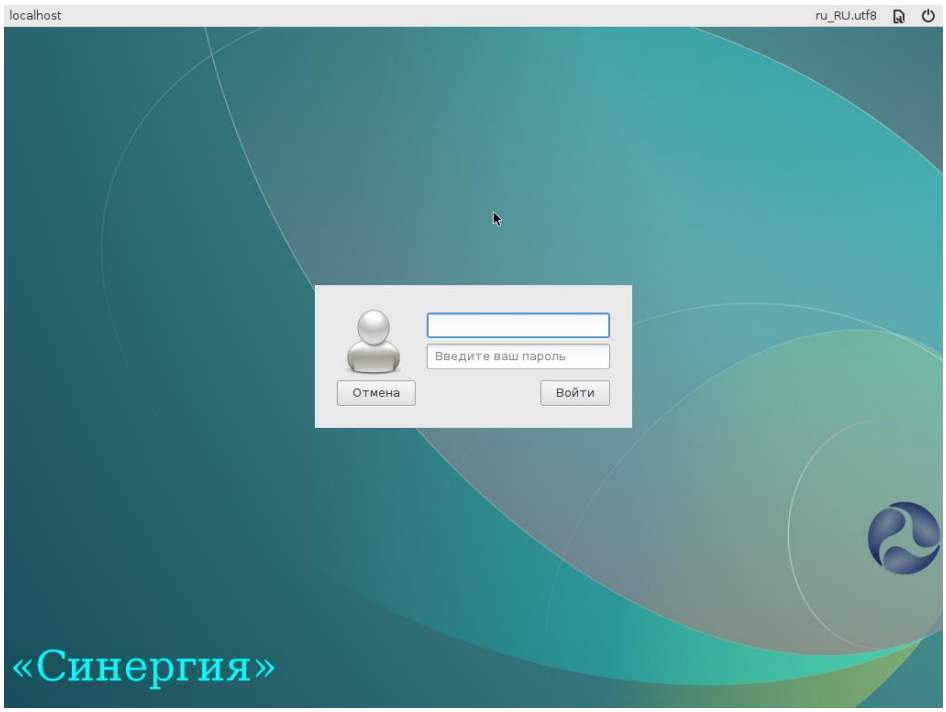

Рисунок 8. Экран входа в ОС

3. УПРАВЛЕНИЕ ПОЛЬЗОВАТЕЛЯМИ

3.1. Работа с пользователями

Управление пользователями означает добавление, удаление пользователей и определение их привилегий.

Управление пользователями предусматривает:

- добавление имен пользователей для возможности их работы в системе;
- определение их привилегий;
- создание и назначение рабочих каталогов;
- определение групп пользователей;
- удаление имен пользователей.

Каждый пользователь должен иметь уникальное регистрационное имя, дающее возможность идентифицировать пользователя и избежать ситуации, когда один пользователь может стереть файлы другого. Кроме того, каждый пользователь должен иметь свой пароль для входа в систему.

3.1.1. Добавление

При добавлении пользователя в файл `/etc/passwd` вносится учетная запись в такой форме:

```
login_name: encrypted_password: user_ ID: group_ ID: user_ information:
login_directory: login_shell
```

В этой записи поля разделены двоеточиями, а значения этих полей приведены в таблице 3.

Таблица 3

Поле	Назначение
login_name	Регистрационное имя пользователя
encrypted_password	Указатель на теневой файл паролей (shadow)
user_ID	Уникальный номер, используемый ОС для идентификации пользователя. Для локальных пользователей не должен превышать 2499
group_ID	Уникальный номер или имя, используемые для идентификации первичной группы пользователя. Если пользователь является членом нескольких групп, он может (если это разрешено системным администратором) в процессе работы менять группу
user_information	Описание пользователя, например, его имя и должность
login_directory	Рабочий каталог пользователя (в котором он оказывается после входа в систему)
login_shell	Оболочка, используемая пользователем, после входа в систему (например, /bin/bash

Также описание файла `/etc/passwd` приведено `в man 5 passwd`.

Для добавления пользователя применяется команда `adduser` с параметром - именем добавляемого пользователя, например:

```
adduser User1
```

Команда `adduser` добавляет пользователя, создает домашний каталог, создает почтовый ящик, а также копирует файлы, имена которых начинаются с точки, из каталога `/etc/skel` в рабочий каталог пользователя. Каталог `/etc/skel` должен содержать все файлы-шаблоны, которые имеет каждый пользователь. Обычно это персональные

конфигурационные файлы, такие как .profile, .cshrc и .login для настройки оболочки. Команда adduser представляет собой файл сценария bash, находящийся в каталоге /usr/sbin. Можно добавить запрос дополнительной информации о пользователе. Чтобы это сделать, необходимо воспользоваться командой chfn для изменения стандартных записей о пользователе.

Описание команд приведено в man adduser и man chfn.

3.1.2. Установка пароля

Для установки пароля пользователя предназначена команда passwd. Необходимо определить пароли для каждого пользователя. Для установки пароля пользователя необходимо выполнить, например, следующее:

- ввести команду и регистрационное имя пользователя, например:

```
passwd User1
```

- нажать клавишу <Enter>;
- после появления приглашения: New password:
- ввести пароль (он не будет отображаться на экране монитора);
- после появления сообщения повторить ввод пароля еще раз, ввести его снова.

Пароль будет зашифрован и внесен в файл /etc/shadow. При выборе пароля необходимо учесть правила, установленные системой по умолчанию: пароль должен иметь не менее восьми символов, а также должен включать в себя как прописные, так и строчные буквы, знаки препинания и цифры (настройка сложности пароля описана в п. 3.3.1.2.1).

Необходимо периодически изменять пароль.

После выполнения всех действий запись в файле будет выглядеть примерно так:

```
anna:x:123:121:Anna_M.:/home/anna:/bin/bash
```

Второе поле записи содержит пароль в зашифрованном виде.

Описание команды приведено в `man passwd`.

Примечание. Если пользователь забыл свой пароль, то администратор системы не может напомнить его пользователю, т.к. в явном виде пароль нигде не хранится. Поэтому действия по восстановлению доступа пользователя в систему сводятся к замене администратором пароля пользователя на новый пароль с помощью команды:

```
passwd user_name
```

3.1.2.1. Настройка сложности пароля

Настройка сложности пароля осуществляется за счет конфигурирования модуля `pam_cracklib` из состава PAM. Данный модуль используется только в режиме управления паролями (а именно, для проверки прочности паролей, которые вводятся пользователями) в связке с другими модулями (которые должны выполнять собственно изменение пароля).

Работа данного модуля по умолчанию заключается в следующем: получить от пользователя пароль, проверить его на прочность, и, если он окажется достаточно крепким, спросить подтверждение введенного пароля и сравнить его с изначально введенным значением. Модуль может выполнять следующие проверки:

- проверка на соответствие словам из словаря;
- проверка на достаточную длину;
- проверка на достаточное разнообразие символов.

Возможные опции модуля:

- `retry=N`: задает максимальное количество попыток выбрать надежный пароль, которые могут быть предоставлены пользователю (по умолчанию предоставляется лишь одна такая попытка);

- type=XXX: замещает в стандартных приглашениях на введение нового пароля «New UNIX password» и «Retype UNIX password» слово «UNIX» рядком XXX;

- difok=N: определяет минимально допустимое количество символов, присутствующих в новом и отсутствующих в старом пароле (значение по умолчанию 10) (если хотя бы половина символов в новом пароле не встречается в старом, пароль также будет принят);

- minlen=N: определяет минимально допустимую длину нового пароля (значение по умолчанию 9) (нужно иметь в виду, что, во-первых, опции dcredit, lcredit, ucredit, ocredit могут фактически увеличивать эту величину, во-вторых, эта величина не может быть меньше 6);

- dcredit=N: если N>=0, определяет максимальный кредит минимально допустимой длины нового пароля, выделенный на включение в пароль цифр (N первых цифр не будут учитываться при сравнивании длины пароля с минимально допустимой; по умолчанию это значение 1); если N<0, определяет минимально допустимое количество цифр в новом пароле;

- ucredit=N: если N>=0, определяет максимальный кредит минимально допустимой длины нового пароля, выделенный на включение в пароль букв в верхнем регистре (N первое количество букв в верхнем регистре не будут учитываться при сравнивании длины пароля с минимально допустимой; по умолчанию это значение 1); если N<0, определяет минимально допустимое количество букв в верхнем регистре в новом пароле;

- lcredit=N: если N>=0, определяет максимальный кредит минимально допустимой длины нового пароля, выделенный на включение в пароль букв в нижнем регистре (N первых букв в нижнем регистре не будут учитываться при сравнивании длины пароля с минимально допустимой; по умолчанию это значение 1); если N<0, определяет минимально допустимое

количество букв в нижнем регистре в новом пароле;

- ocredit=N: если N>=0, определяет максимальный кредит минимально допустимой длины нового паролю, выделенный на включение в пароль других символов (N первых таких символов не будут учитываться при сравнивании длины пароля с минимально допустимой; по умолчанию это значение 1); если N<0, определяет минимально допустимое количество других символов в новом пароле (по умолчанию это ограничение не накладывается).

Рассмотрим файл /etc/pam.d/common-password (может иметь другое имя, взависимости от дистрибутива) на который ссылается содержимое /etc/pam.d/passwd

Листинг файла /etc/pam.d/system-auth-ac

```
password        requisite           pam_cracklib.so retry=3 minlen=8 lcredit=1
ucredit=1 dcredit=1 ocredit=1 difok=3
password          [success=1  default=ignore]             pam_unix.so  obscure
use_authtok try_first_pass sha512
password       requisite         pam_deny.so
password       required         pam_permit.so
Изучим строку
password        requisite           pam_cracklib.so retry=3 minlen=8 lcredit=1
ucredit=1 dcredit=2 ocredit=1 difok=3
```

В данной строке параметр retry=3 это количество попыток для смены пароля, длина пароля должна составлять минимум 8 символов (minlen=8), из которых два должны быть числами (dcredit=2), один - символом верхнего регистра (ucredit=1), один - символом нижнего регистра (lcredit=1), еще один — не алфавитным знаком (ocredit=1). Значеное опции difok=3 означает, что допустимое количество символов, присутствующих в новом и отсутствующих в старом пароле, должно быть не менее 3-х.

3.1.3. Удаление

Есть несколько степеней удаления пользователя:

- лишение пользователя возможности входа в систему;
- удаление записи;
- удаление пользователя и всех его файлов.

Лишение пользователя возможности входа в систему полезно в случае его длительного перерыва в работе.

На время отсутствия пользователя можно заблокировать его запись с помощью команды:

```
usermod -L user_name
```

Для разблокировки записи выполнить команду:

```
usermod -U user_name
```

При этом все пользовательские файлы и каталоги остаются нетронутыми, но войти в систему под его именем становится невозможно.

Для смены имени пользователя выполнить команду:

```
usermod -l new_user_name old_user_name
```

Удаление пользователя таким образом производится либо путем непосредственного редактирования файла `/etc/passwd`, либо с помощью команды:

```
userdel user_name
```

Удаление пользователя и всех его файлов - это окончательное и полное удаление пользователя из системы с помощью команды:

```
find / -user user_name -exec rm -r {} \;
```

Затем следует удалить рабочий каталог пользователя с помощью команды:

```
rmdir user_home_dir
```

и запись о пользователе из файла `/etc/passwd`.

Также удалить пользователя и его домашний каталог можно с помощью команды:

```
userdel -r user_name
```

Для удаления файлов, не принадлежащих ни одному пользователю в системе, выполнить команду:

```
find / -nouser -exec rm -r {} \;
```

Описание команд приведено в `man usermod`, `man userdel` и `man find`.

3.1.4. Неудачный вход в систему

Команда показывает содержимое журнала неудачных попыток (файл `/var/log/faillog`) входа в систему. Также она может быть использована для управления счетчиком неудачных попыток и их ограничением. При запуске `faillog` без параметров выводятся записи `faillog` только тех пользователей, у которых имеется хотя бы одна неудачная попытка входа.

Для сброса неудачных попыток входа необходимо пользоваться опцией `-r`

Описание команды, а также файла `/var/log/faillog` приведено в `man faillog` и `man 5 faillog`.

3.2. Работа с группами

Каждый пользователь является членом группы. Различным группам можно назначить различные возможности и привилегии.

Информация о группах содержится в файле `/etc/group`. Пример записи из этого файла:

```
Admin :: 21: user1, user2, user3
```

Здесь имя группы - `admin`, идентификатор - 21, членами группы являются `user1`, `user2`, `user3`. Пользователь может быть членом

нескольких групп и переходить из одной в другую в процессе работы.

Описание файла `/etc/group` приведено в `man 5 group`.

3.2.1. Добавление

Добавление группы производится с помощью команды:

```
groupadd users
```

Данная команда добавляет группу `users`.

Также новая группа создается путем непосредственного редактирования файла `/etc/group`, ввода необходимой информации о группе. Каждой группе присваивается свой уникальный идентификационный номер (ОС при работе учитывает номер, а не имя группы), поэтому, если присвоить двум группам один номер, для ОС получится одна и та же группа.

Описание команды приведено в `man groupadd`.

3.2.2. Удаление

Удаление группы производится с помощью команды:

groupdel users

Данная команда удаляет группу users.

Также удаление группы производится путем удаления записи о ней в файле /etc/group.

Описание команды приведено в man groupdel.

3.3. Рабочие каталоги пользователей

Рабочие каталоги пользователей на одном компьютере следует разместить в отдельном каталоге верхнего уровня (по умолчанию - `/home`). Если пользователей достаточно много, то следует оптимально разделить их

домашние каталоги по группам (подразделениям), например `/home/hr` (отдел персонала) `/home/admins, /home/buhg` и т.д.).

Таким образом, они будут достаточно логично сгруппированы, что в дальнейшем облегчит администрирование системы.

4. СИСТЕМНЫЕ СЕРВИСЫ И КОМАНДЫ

4.1. Сервисы

Сервисы - это программы, которые запускаются и останавливаются через инициализированные скрипты, расположенные в каталоге /etc/init.d. Многие из этих сервисов запускаются на этапе старта ОС. /usr/sbin/service обеспечивает интерфейс (взаимодействие) пользователя с инициализированными скриптами. А сами эти скрипты обеспечивают интерфейс для управления сервисами, предоставляя пользователю опции для запуска, остановки, перезапуска, запроса состояния сервиса и выполнения других воздействий на сервис. К примеру, инициализированный скрипт сервиса syslog имеет следующие опции:

```
/usr/sbin/service cron
Usage: /etc/init.d/cron
{start|stop|status|restart|reload|force-reload}
```

В ОС можно просмотреть текущее состояние всех системных служб с помощью опции --status-all команды service:

```
- /usr/sbin/service --status-all
- acpid (pid 2481) is running...
- anacron (pid 2647) is running...
- atd (pid 2657) is running...
- auditd (pid 2189) is running...
```

В ОС существует шесть системных уровней выполнения, каждый из которых определяет список служб (сервисов), запускаемых на данном уровне. Уровни 0 и 6 соответствуют выключению и перезагрузке системы.

При загрузке системы процесс `init` запускает все сервисы, указанные в каталоге `/etc/rc(0-6).d/` для уровня по умолчанию. Поменять его можно в конфигурационном файле `/etc/inittab`. Строка:

```
id:2:initdefault:
```

соответствует второму уровню выполнения.

Команда `telinit` наиболее эффективна для тестирования изменений, внесенных в файл `inittab`. При указании аргумента `-q` процесс `init` повторно читает `inittab`.

Для перехода системы на нужный уровень выполнения можно воспользоваться командой `init`, например:

```
init 3
```

Данная команда переведет систему на третий уровень выполнения, запустив все сервисы, указанные в каталоге `/etc/rc3.d/`.

Описание данных команд и сервисов приведено на страницах руководства `man`.

4.2. Команды

Основные системные команды ОС приведены в таблице 5.

Таблица 5

Команда	Назначение
awk	Язык обработки строковых шаблонов
bc	Строковый калькулятор
chfn	Управление информацией учетной записи (имя, описание)
chsh	Управление выбором командного интерпретатора (по умолчанию - для учетной записи)
cut	Разбивка файла на секции, задаваемые контекстными разделителями

Команда	Назначение
df	Вывод отчета об использовании дискового пространства
dmesg	Вывод содержимого системного буфера сообщений
du	Вычисление количества использованного пространства элементов ФС
echo	Вывод содержимого аргументов на стандартный вывод
egrep	Поиск в файлах содержимого согласно регулярных выражений
fgrep	Поиск в файлах содержимого согласно фиксированных шаблонов
file	Определение типа файла
find	Поиск файла по различным признакам в иерархии каталогов
gettext	Получение строки интернационализации из каталогов перевода
grep	Вывод строки, содержащей шаблон поиска
groupadd	Создание новой учетной записи группы
groupdel	Удаление учетной записи группы
groupmod	Изменение учетной записи группы
groups	Вывод списка групп
gunzip	Распаковка файла
gzip	Упаковка файла
hostname	Вывод и задание имени хоста
install	Копирование файла с установкой атрибутов
ipcrm	Удаление ресурса IPC
ipcs	Вывод характеристик ресурса IPC
kill	Прекращение выполнения процесса
killall	Удаление процессов по имени
lp	Отправка задания печати на принтер
ls	Вывод содержимого каталога

Команда	Назначение
lsb_release	Вывод информации о дистрибутиве
m4	Макропроцессор
mknod	Создание файла специального типа
mktemp	Генерация уникального имени файла
more	Постраничный вывод содержимого файла
mount	Монтирование ФС
newgrp	Смена идентификатора группы
nice	Изменение приоритета процесса перед его запуском
nohup	Работа процесса после выхода из системы
od	Вывод содержимого файла в восьмеричном и других видах
passwd	Смена пароля учетной записи
patch	Применение файла описания изменений к оригинальному файлу
pidof	Вывод идентификатора процесса по его имени
ps	Вывод информации о процессах
renice	Изменение уровня приоритета процесса
sed	Строковый редактор
sh	Командный интерпретатор
shutdown	Команда останова системы
su	Изменение идентификатора запускаемого процесса
sync	Сброс системных буферов на носители
tar	Файловый архиватор
umount	Размонтирование ФС
useradd	Создание новой учетной записи или обновление существующей
userdel	Удаление учетной записи и соответствующих файлов окружения

Команда	Назначение
usermod	Модификация информации об учетной записи
w	Список пользователей, кто в настоящий момент работает в системе и с чем
who	Вывод списка пользователей системы

Описание команд приведено на страницах руководства `man`.

4.3. Создание сети TCP/IP

Процесс создания сети TCP/IP состоит из следующих этапов:

- планирование сети;
- назначение IP-адресов;
- настройка сетевых интерфейсов;
- настройка статических маршрутов.

4.3.1. Планирование сети

Планирование сети включает: определение сегментов сети, определение технических и программных средств, с помощью которых сегменты объединяются в сеть, определение серверов и рабочих станций, которые будут установлены в каждом сегменте, и определение типа среды (витая пара и др.).

4.3.2. Назначение lP-адресов

Адреса назначают сетевым интерфейсам, а не компьютерам. Если у компьютера есть несколько интерфейсов, у него будет несколько сетевых адресов.

Назначая компьютеру IP-адрес, следует указать соответствие между этим адресом и именем компьютера в файле `/etc/hosts`. Это соответствие позволит обращаться к компьютерам по их именам.

4.3.3. Настройка сетевых интерфейсов

Команда `ifconfig` используется для включения и выключения сетевого интерфейса, задания IP-адреса, широковещательного адреса и связанной с ним маски подсети, а также для установки других опций и параметров. Она обычно выполняется во время первоначальной настройки, но может применяться и для внесения изменений в дальнейшем.

В большинстве случаев команда `ifconfig` имеет следующий формат:

```
ifconfig интерфейс [семейство] адрес up опция ...
```

Пример:

```
ifconfig eth0 128.138.240.1 up netmask 255.255.255.0 broadcast 128.138.240.255
```

Здесь интерфейс обозначает аппаратный интерфейс, к которому применяется команда. Как правило, это двух-трехсимвольное имя устройства, за которым следует число.

Примеры распространенных имен `eth1`, `lo0`, `ppp0` образуются из имени драйвера устройства, используемого для управления им. Для того чтобы выяснить, какие интерфейсы имеются в системе, можно воспользоваться командой:

```
netstat-i
```

Ключевое слово `up` включает интерфейс, а ключевое слово `down` выключает его.

Описание команды приведено в `man ifconfig`.

4.3.4. Настройка статических маршрутов

Команда `route` определяет статические маршруты - явно заданные элементы таблицы маршрутизации, которые обычно не меняются даже в тех случаях, когда запускается серверный процесс маршрутизации.

Маршрутизация выполняется на уровне IP. Когда поступает пакет, предназначенный для другого компьютера, IP-адрес пункта назначения пакета сравнивается с маршрутами, указанными в таблице маршрутизации ядра. Если номер сети пункта назначения совпадает с номером сети какого-либо маршрута, то пакет направляется по IP-адресу следующего шлюза, связанного с данным маршрутом.

Существующие маршруты можно вывести на экран командой `route`.

Описание команды приведено в `man route`.

4.4. Настройка SSH

`SSH` - это клиент-серверная система для организации защищенных туннелей между двумя и более компьютерами. В таких туннелях защищаются все передаваемые данные, в т.ч. пароли.

4.4.1. Служба sshd

Служба `sshd` запускается на этапе начальной загрузки из сценария `/etc/rc.d/init.d/sshd`. Этот сценарий, а также ссылки на него в виде сценариев запуска и останова службы создаются в процессе установки программы. По умолчанию служба прослушивает порт 22. Когда поступает запрос на подключение, он порождает дочерний процесс, который управляет передачей данных в рамках конкретного соединения.

Служба берет свои конфигурации сначала из командной строки, затем из файла`/etc/ssh/sshd_config`.

Синтаксис:

`sshd [-deiqtD46] [-b bits] [-f config_file] [-g login_grace_time] [-h host_key_file] [-k key_gen_time] [-o option] [-p port] [-u len]`

Параметры, которые могут присутствовать в файле `/etc/ssh/sshd_config`, описаны в таблице 21. Пустые строки, а также строки, начинающиеся с #, игнорируются. Названия параметров не чувствительны к регистру символов.

Таблица 21

Ключ	**Описание**
AllowGroups	Задает разделенный пробелами список групп. Эти группы будут допущены в систему
DenyGroups	То же, что AllowGroups, только смысл проверки обратный. Записанные в этот параметр группы не будут допущены в систему
AllowUsers	Задает разделенный пробелами список пользователей. Только перечисленные пользователи получают доступ в систему. По умолчанию доступ разрешен всем пользователям
DenyUsers	То же, с противоположным смыслом проверки
AFSTokenPassing	Указывает на то, может ли маркер afs пересылаться на сервер. По умолчанию - yes
AllowTCPForwarding	Указывает на то, разрешены ли запросы на переадресацию портов (по умолчанию - yes)
Banner	Отображает полный путь к файлу сообщения, выводимого перед аутентификацией пользователя
ChallengeResponseAuthentication	Указывает на то, разрешена ли аутентификация по

Ключ	Описание
	методу «оклик - ответ». По умолчанию - yes
Ciphers	Задает разделенный запятыми список методов защиты соединения, разрешенных для использования
CheckMail	Указывает на то, должна ли служба sshd проверять почту в интерактивных сеансах регистрации (по умолчанию - no)
ClientAliveInterval	Задает интервал ожидания в секундах, по истечении которого клиенту посылается запрос на ввод данных
ClientAliveCountMax	Задает число напоминающих запросов, посылаемых клиенту. Если по достижении указанного предела от клиента не поступит данных, сеанс завершается и сервер прекращает работу. Значение по умолчанию - 3
HostKey	Полный путь к файлу, содержащему личный ключ компьютера. (По умолчанию - /etc/ssh/ssh_host_key)
GatewayPorts	Указывает на то, могут ли удаленные компьютеры подключаться к портам, для которых клиент запросил переадресацию (по умолчанию - no)
HostbasedAuthentication	Указывает на то, разрешена ли аутентификация пользователей с проверкой файлов .rhosts и /etc/hosts.equiv и открытого ключа компьютера. Значение по умолчанию - no
IgnoreRhosts	Указывает на то, игнорируются ли файлы $HOME/.rhosts и $HOME/.shosts. По умолчанию - yes
IgnoreUserKnownHosts	Указывает на то, игнорируется ли файл $HOME/.ssh/known_hosts в режимах аутентификации RhostsRSAAuthentication и HostbasedAuthentication (по умолчанию - no)
KeepAlive	Если равен yes (по умолчанию), демон sshd будет периодически проверять наличие связи с клиентом. В

Ключ	Описание
	случае неуспешного завершения проверки соединение разрывается. Чтобы отключить этот механизм, надо задать параметр, равным no, в файле конфигурации и сервера, и клиента
KerberosAuthentication	Указывает на то, разрешена ли аутентификация с использованием Kerberos. По умолчанию - no
KerberosOrLocalPasswd	Указывает на то, должна ли использоваться локальная парольная аутентификация в случае неуспешной аутентификации на основе Kerberos
KerberosTgtPassing	Указывает на то, может ли структура tgt системы Kerberos пересылаться на сервер (по умолчанию - no)
KerberosTicketCleanup	Указывает на то, должен ли при выходе пользователя удаляться кэш-файл его пропуска Kerberos
ListenAddress	Задает интерфейс, к которому подключается служба sshd. Значение по умолчанию - 0.0.0.0, т.е. любой интерфейс
LoginGraceTime	Задает интервал времени в секундах, в течение которого должна произойти аутентификация пользователя. Если процесс аутентификации не успевает завершиться вовремя, сервер разрывает соединение и завершает работу. Значение по умолчанию - 600 с
LogLevel	Задает степень подробности журнальных сообщений. Возможные значения: quiet, fatal, error, info (по умолчанию), verbose, debag (не рекомендуется)
MACs	Задает разделенный запятыми список доступных алгоритмов MAC (код аутентификации сообщений), используемых для обеспечения целостности данных
MaxStartups	Задает максимальное число одновременных неаутентифицированных соединений с демоном sshd

Ключ	Описание
PAMAuthenticationViaKbdInt	Указывает на то, разрешена ли парольная аутентификация с использованием PAM (по умолчанию - no)
PasswordAuthentication	Если равен yes (по умолчанию), и ни один механизм беспарольной аутентификации не приносит положительного результата, тогда пользователю выдается приглашение на ввод пароля, который проверяется самим демоном sshd. Если параметр равен no, парольная аутентификация запрещена
PermitEmptyPasswords	Если равен yes, пользователи, не имеющие пароля, могут быть аутентифицированы службой sshd. Если параметр равен no (по умолчанию), пустые пароли запрещены
PermitRootLogin	Указывает на то, может ли пользователь root войти в систему с помощью команды ssh. Возможные значения: yes (по умолчанию), without-password, forced-command-only и no
PidFile	Задает путь к файлу, содержащему идентификатор главного процесса (по умолчанию - /var/run/sshd.pid)
Port	Задает номер порта, к которому подключается sshd. По умолчанию - 22
PrintLastLog	Указывает на то, должна ли служба sshd отображать сообщение о времени последнего доступа. По умолчанию - yes
PrintMotd	Указывает на то, следует ли после регистрации в системе отображать содержимое файла /etc/motd. По умолчанию - yes
Protocol	Задает разделенный запятыми список версий протокола, поддерживаемых службой sshd
PubKeyAuthentication	Указывает на то, разрешена ли аутентификация с

Ключ	Описание
	использованием открытого ключа (по умолчанию - yes)
ReverseMappingCheck	Указывает на то, должен ли выполняться обратный поиск имен. По умолчанию - no
StrictModes	Если равен yes (по умолчанию), sshd будет запрещать доступ любому пользователю, чей начальный каталог и/или файл .rhosts принадлежат другому пользователю, либо открыт для записи
Subsystem	Предназначается для конфигурирования внешней подсистемы. Аргументами является имя подсистемы и команда, выполняемая при поступлении запроса к подсистеме
SyslogFacility	Задает название средства, от имени которого регистрируются события в системе Syslog. Возможны значения: DAEMON, USER, AUTH (по умолчанию), L0CAL0-7
UseLogin	Указывает на то, должна ли применяться команда login для организации интерактивных сеансов регистрации (по умолчанию - no)
X11Forwarding	Указывает на то, разрешена ли переадресация запросов к системе X Window (по умолчанию -no)
X11DisplayOffset	Задает номер первого дисплея (сервера) системы XWindow, доступного демону sshd для переадресации запросов (по умолчанию - 10)
XAuthLocation	Задает путь к команде xauth (по умолчанию - /usr/X11R6/bin/xauth)

4.4.2. Клиент ssh

Клиентом является команда `ssh`. Синтаксис командной строки:

```
ssh [-afgknqstvxACNTX1246] [-b bind_address] [-c cipher_spec] [-e escape_char] [-i identity_file] [-login_name] [-m mac_spec] [-o option] [-p port] [-F configfile] [-L port:host:hostport] [-R port:host:hostport] [-D port] hostname | user@hostname [command]
```

Подробно со значениями флагов можно ознакомиться в руководстве `man`.

В простом варианте инициировать соединение с сервером `sshd` можно командой:

```
ssh 10.1.1.170
```

где `10.1.1.170` - IP-адрес компьютера с запущенной службой `sshd`. При этом `sshd` будет считать, что пользователь, запрашивающий соединение, имеет такое же имя, под каким он аутентифицирован на компьютере-клиенте. Теоретически клиент `ssh` может заходить на сервер `sshd` под любым именем, используя флаг:

```
-l <имя_клиента>
```

Однако сервер будет согласовывать ключ сеанса (например, при беспарольной аутентификации по открытому ключу пользователя), проверяя открытые ключи в домашнем каталоге пользователя именно с этим именем на компьютере-клиенте. Если же используется парольная аутентификация, на компьютере-сервере должна существовать учетная запись с таким именем. Использовать беспарольную аутентификацию по открытым ключам компьютера настоятельно не рекомендуется, т.к. при этом способе в системе должны существовать потенциально опасные файлы: `/etc/hosts.equiv`, `/etc/shosts.equiv`, `$HOME/.rhosts`, `$HOME/.shosts`.

Команда `ssh` берет свои конфигурационные установки сначала из командной строки, затем из пользовательского файла `$HOME/.ssh/config`

и из общесистемного файла `/etc/ssh/ssh_config`. Если идентичные параметры заданы по-разному, выбирается самое первое значение.

В таблице 22 описаны параметры, которые могут присутствовать в файле $HOME/.ssh/config или `/etc/ssh/ssh_config`. Пустые строки и комментарии игнорируются.

Таблица 22

Параметр	**Описание**
CheckHostIP	Указывает на то, должна ли команда ssh проверять IP-адреса в файле known_hosts (по умолчанию - yes)
Ciphers	Задает разделенный запятыми список методов защиты сеанса, разрешенных для использования. По умолчанию - aes122b-cbc, 3des-cbc, blowfish-cbc, cast128-cbc, arcfour, aes192-cbc, aes256-cbc
Compression	Указывает на то, должны ли данные сжиматься с помощью команды gzip (по умолчанию - no). Эта установка может быть переопределена с помощью опции командной строки -C
ConnectionAttempts	Задает число неудачных попыток подключения (одна в секунду), после чего произойдет завершение работы. Значение по умолчанию - 4
EscapeChar	Задает escape-символ, используемый для отмены специального назначения следующего символа в сеансах с псевдотерминалом. По умолчанию -. Значение none запрещает использование escape-символа
ForwardAgent	Указывает на то, будет ли запрос к команде ssh-agent переадресован на удаленный сервер (по умолчанию - no)
ForwardX11	Указывает на то, будут ли запросы к системе X Window автоматически переадресовываться через

Параметр	Описание
	SSH-туннель с одновременной установкой переменной среды display (по умолчанию - no)
GatewayPorts	Указывает на то, могут ли удаленные компьютеры подключаться к локальным портам, для которых включен режим переадресации (по умолчанию - no)
GlobalKnownHostsFile	Задает файл, в котором хранится глобальная база ключей компьютера (по умолчанию - /etc/ssh/ssh_known_hosts)
HostbasedAuthentication	Указывает на то, разрешена ли аутентификация пользователей с проверкой файлов .rhosts, /etc/hosts.equiv и открытого ключа компьютера. Этот параметр рекомендуется установить в значение no
HostKeyAlgorithm	Задает алгоритмы получения ключей компьютеров в порядке приоритета. Выбор по умолчанию - ssh-rsa, ssh-dss
HostKeyAlias	Задает псевдоним, который должен использоваться при поиске и сохранении ключей компьютера
HostName	Задает имя или IP-адрес компьютера, на котором следует регистрироваться. По умолчанию выбирается имя, указанное в командной строке
IdentityFile	Задает файл, содержащей личный ключ пользователя (по умолчанию - $HOME/.ssh/identity). Вместо имени начального каталога пользователя может стоять символ ~. Разрешается иметь несколько таких файлов. Все они будут проверены в указанном порядке
KeepAlive	Если равен yes (по умолчанию), команда ssh будет периодически проверять наличие связи с сервером. В случае неуспешного завершения проверки (в т.ч. из-за временных проблем с маршрутизацией) соединение разрывается. Чтобы отключить этот механизм, следует

Параметр	Описание
	задать данный параметр, равным no, в файлах /etc/ssh/sshd_config и /etc/ssh/ssh_config (либо $HOME/.ssh/config)
KerberosAuthentication	Указывает на то, разрешена ли аутентификация с применением Kerberos
KerberosTgtPassing	Указывает на то, будет ли структура tgt системы Kerberos пересылаться на сервер
LocalForward	Требует значения в формате порт:узел:удаленный_порт. Указывает на то, что запросы к соответствующему локальному порту перенаправляются на заданный порт удаленного узла
LogLevel	Задает степень подробности журнальных сообщений команды ssh. Возможные значения: quiet, fatal, error, info (по умолчанию), VERBOSE, DEBAG
MACs	Задает разделенный запятыми список доступных алгоритмов аутентификации сообщений для обеспечения целостности данных. Стандартный выбор: hmac-md5, hmac-sha1, hmac-ripemd160@openssh.com, hmac-sha1-96, hmac-md5-96
NumberOfPasswordPrompts	Задает число допустимых попыток ввести пароль (по умолчанию - 3)
PasswordAuthentication	Если равен yes (по умолчанию), то в случае необходимости команда ssh пытается провести парольную аутентификацию
Port	Задает номер порта сервера (по умолчанию - 22)
PreferredAuthentications	Задает порядок применения методов аутентификации (по умолчанию - publickey, password, keyboard-interactive)
Protocol	Задает в порядке приоритета версии протокола SSH
ProxyCommand	Задает команду, которую следует использовать вместо

Параметр	Описание
	ssh для подключения к серверу. Эта команда выполняется интерпретатором /bin/sh. Спецификация %p соответствует номеру порта, а %h - имени удаленного узла
PubkeyAuthentication	Указывает на то, разрешена ли аутентификация с использованием открытого ключа (по умолчанию - yes)
RemoteForward	Требует значения в формате удаленный_порт:узел:порт. Указывает на то, что запросы к соответствующему удаленному порту перенаправляются на заданный порт заданного узла. Переадресация запросов к привилегированным портам разрешена только после получения прав суперпользователя на удаленной системе. Эта установка может быть переопределена с помощью опции командной строки -R
StrictHostKeyCheking	Если равен yes, команда не будет автоматически добавлять ключи компьютера в файл $HOME/.ssh/known_hosts и откажется устанавливать соединение с компьютерами, ключи которых изменились. Если равен no, команда будет добавлять непроверенные ключи сервера в указанные файлы. Если равен ask (по умолчанию), команда будет спрашивать пользователя о том, следует ли добавлять открытый ключ сервера в указанные файлы
UsePrivilegedPort	Указывает на то, можно ли использовать привилегированный порт для установления исходящих соединений. Значение по умолчанию - no
User	Задает пользователя, от имени которого следует регистрироваться в удаленной системе. Эта установка

Параметр	Описание
	может быть переопределена с помощью опции командной строки -l
UserKnownHostsFile	Задает файл, который используется для автоматического обновления открытых ключей
XAuthLocation	Задает путь к команде xauth (по умолчанию - /usr/X11R6/bin/xauth)

Клиентские конфигурационные файлы бывают глобальными, на уровне системы (`/etc/ssh/ssh_config`), и локальными, на уровне пользователя (`$HOME/.ssh/config`). Следовательно, пользователь может полностью контролировать конфигурацию клиентской части `SSH`.

Конфигурационные файлы разбиты на разделы, установки которых относятся к отдельному компьютеру, группе компьютеров или ко всем компьютерам. Установки разных разделов могут перекрывать друг друга.

5. УПРАВЛЕНИЕ ПРОГРАММНЫМИ ПАКЕТАМИ

В ОС используются программные пакеты (далее по тексту - пакеты) в формате «.deb». Для управления пакетами в режиме командной строки (или в эмуляторе терминала в графическом режиме) предназначены набор команд нижнего уровня `dpkg` и комплекс программ высокого уровня `apt-get`, `apt-cache` и `aptitude`.

По умолчанию обычный пользователь не имеет права использовать эти инструменты. Для всех операций с пакетами (за исключением некоторых случаев получения информации о пакетах) необходимы права суперпользователя, которые администратор может получить через механизм `sudo`.

Средства управления пакетами обеспечивают возможность автоматизированной установки обновлений ОС.

5.1. Набор команд dpkg

Набор команд `dpkg` предназначен, в основном, для операций с пакетами на локальном уровне. С помощью команды `dpkg` и других команд этого набора можно устанавливать и удалять пакеты, собирать их из исходных текстов, получать информацию о конкретном пакете и об установленных в системе пакетах:

```
dpkg -i <полный_путь>/<полное_имя_пакета>
```

Если пакет (например, `iptables_1.4.21-2_amd64.deb`), который необходимо установить, помещен в рабочий каталог (например, `/home/user1`) или находится на смонтированном внешнем носителе, следует выполнить следующую команду:

```
dpkg -i /home/user1/iptables_1.4.21-2_amd64.deb
```

В случае если неудовлетворенные зависимости пакета отсутствуют, он будет установлен. В случае нарушения зависимостей `dpkg` выдаст сообщение об ошибке, в котором будут перечислены все необходимые пакеты, которые следует установить, чтобы разрешить обязательные зависимости.

Для удаления ненужного пакета, но сохранения всех его файлов настройки, следует выполнить команду:

```
dpkg -r <значимая_часть_имени_пакета>
```

Для приведенного выше примера команда будет выглядеть следующим образом:

```
dpkg -r iptables
```

Для удаления пакета и очистки системы от всех его компонентов (в случае, если данный пакет не связан зависимостями с другими установленными пакетами) следует выполнить команду:

```
dpkg -P <значимая_часть_имени_пакета>
```

Если же удаляемый пакет зависит от других пакетов, последует сообщение об ошибке с перечнем зависимостей. Следует отметить, что использование полного имени пакета регулируется для всех команд семейства `dpkg` простым правилом: для любых действий с уже установленным пакетом в командной строке применяется значимая часть имени, а во всех остальных случаях - полное имя.

Подробное описание команды приведено в `man dpkg`.

5.2. Комплекс программ apt

Комплекс программ apt предназначен, в основном, для управления всеми операциями с пакетами (в т. ч. автоматическим разрешением зависимостей) при наличии доступа к сетевым или локальным архивам (источникам) пакетов.

5.2.1. Настройка доступа к архивам пакетов

Информация о сетевых и локальных архивах пакетов для комплекса программ `apt` содержится в файле `/etc/apt/sources.list`. В этом файле находится список источников пакетов, который используется программами для определения местоположения архивов. Список источников разрабатывается для поддержки любого количества активных источников и различных видов этих источников. В данном файле перечисляется по одному источнику на строку, где источники следуют в порядке убывания их приоритета.

Описание файла `/etc/apt/sources.list` приведено в `man sources.list`.

Пример файла `sources.list`:

```
deb [trusted=1] file://home/repo/ogun-fstek ogun main
```

При установке ОС репозиторий со всеми доступными пакетами создается в `/home/repo/ogun-fstek`. Нет необходимости монтировать машинный носитель с репозиторием, т.к. весь репозиторий уже располагается в файловой система после установки ОС (для любого варианта установки).

5.2.2. Установка и удаление пакетов

После установки ОС вместе с репозиторием создается локальная БД с информацией обо всех доступных пакетах. Эта информация может выводиться в различной форме при помощи команды `apt-cache`. Например, команда:

```
apt-cache show iptables
```

выведет всю информацию, содержащуюся в описании пакета iptables.

Обновить содержимое локальной БД можно при помощи команды:

```
apt-get update
```

Эту операцию необходимо выполнять при каждом изменении как списка источников пакетов, так и содержимого этих источников (например, при переходе к использованию обновленной версии ОС).

Полное обновление всех установленных в системе пакетов производится при помощи команды:

```
apt-get upgrade
```

Установка отдельного пакета (если он отсутствовал в системе) производится при помощи команды:

```
apt-get install <значимая_часть_имени_пакета>
```

При этом будут исследованы и разрешены все обязательные зависимости и, при необходимости, установлены необходимые дополнительные пакеты.

Удаление пакета (с сохранением его файлов настройки) производится при помощи команды:

```
apt-get remove <значимая_часть_имени_пакета>
```

Если при этом необходимо полностью очистить систему от всех компонент удаляемого пакета, то применяется команда:

```
apt-get remove --purge <значимая_часть_имени_пакета>
```

Описание команд приведено в `man apt-cache` и `man apt-get`.

5.3. Пересмотр прав доступа к файлам

Во время установки пакета права доступа к файлам назначаются автоматически, и установочный сценарий корректно определяет права доступа к каждому файлу пакета. Однако следует пересмотреть их и решить, разрешено ли работать с пакетом тем пользователям, для которых он предназначен, и не может ли злоумышленник воспользоваться им для проникновения в систему.

Права доступа к исполняемым файлам позволяют всем пользователям запускать их на выполнение, но удалять или модифицировать такие файлы может только суперпользователь. Обычно приложения устанавливаются в каталог с правами чтения всеми пользователями, но без права записи в него.

Примечание. По умолчанию в ОС для всех пользователей заблокирована возможность изменения атрибутов любого неисполняемого файла на исполняемый. Это обеспечивается передачей ядру значения «1» для соответствующих параметров:

```
/proc/sys/fs/user_noacl
/proc/sys/fs/user_nox
/proc/sys/kernel/user_nosetuid
```

Администратор может снять это ограничение, откорректировав файл `/etc/sysctl.conf`, в котором следует записать следующие строки:

```
fs.user_noacl=0
fs.user_nox=0
kernel.user_nosetuid=0
```

После этого необходимо выполнить команду:

```
/sbin/sysctl -p /etc/sysctl.conf
```

При следующей загрузке системы будут использоваться новые значения параметров, и ограничение на изменение атрибутов файла для пользователей будет отменено.

Описание файла `/etc/sysctl.conf` приведено в `man sysctl.conf`.

5.4. Удаление приложения

Не всегда достаточно просто стереть с диска файлы приложения и удалить его каталог. Драйверы и другие приложения должны быть корректно отключены во избежание проблем в дальнейшем. Мониторинг

системы в процессе установки и ведение журнала установки позволяют корректно удалить приложение, ставшее ненужным.